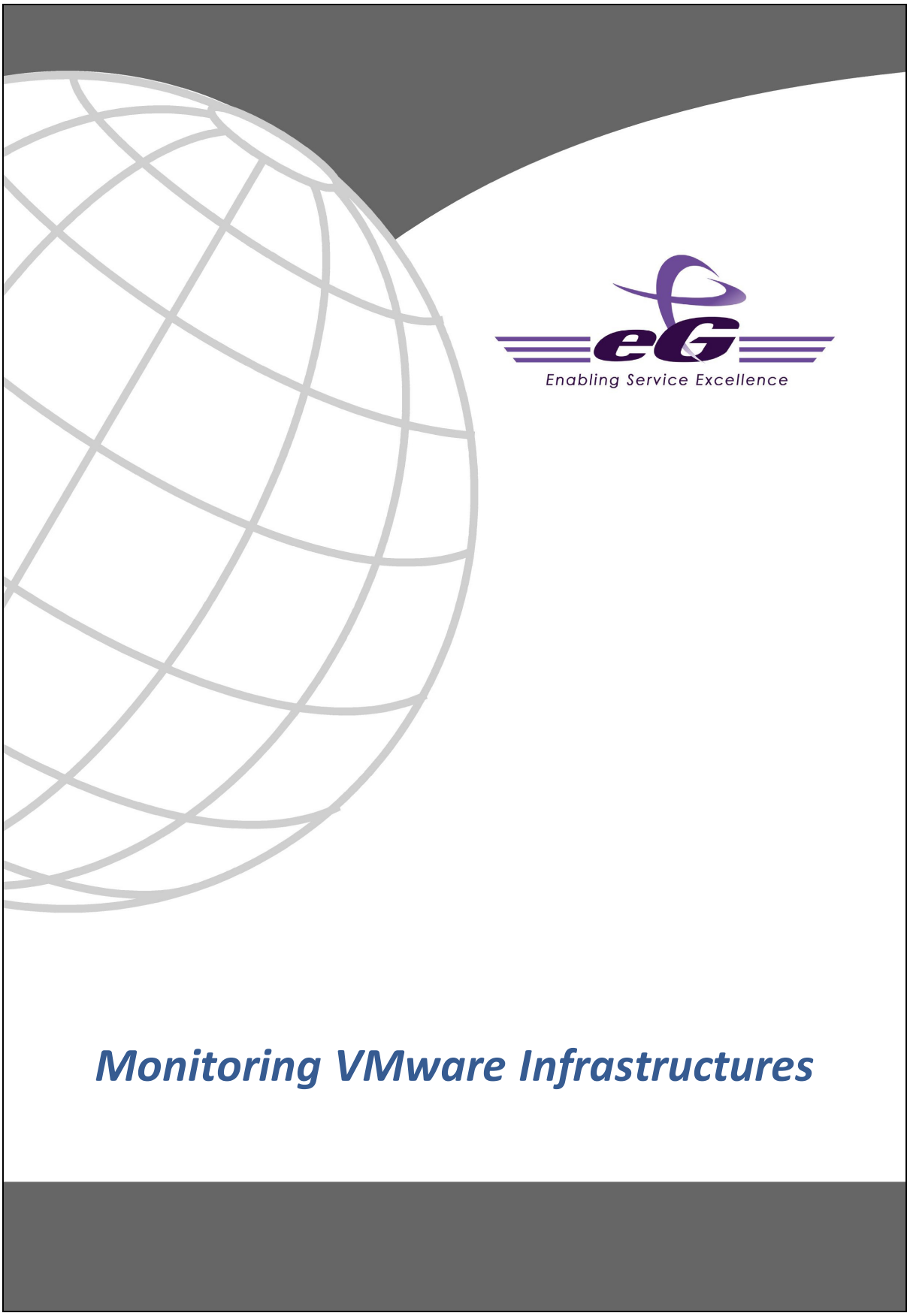


Monitoring VMware Infrastructures

Restricted Rights Legend

The information contained in this document is confidential and subject to change without notice. No part of this document may be reproduced or disclosed to others without the prior permission of eG Innovations Inc. eG Innovations Inc. makes no warranty of any kind with regard to the software and documentation, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

Trademarks

Microsoft Windows, Windows NT, Windows 2008, Windows 7, Windows 8 and Windows 2010 are either registered trademarks or trademarks of Microsoft Corporation in United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Copyright

©2016 eG Innovations Inc. All rights reserved.

Table of contents

INTRODUCTION 1

1.1 Virtualization Deployment Models 2

1.2 Challenges in Monitoring Virtual Environments 3

1.3 Limitations of Existing VMware Monitoring Solutions 5

HOW DOES EG ENTERPRISE MONITOR VMWARE ENVIRONMENTS? 6

2.1 Pre-requisites for Monitoring VMware Infrastructures 8

2.1.1 General Pre-requisites 8

2.1.2 Pre-requisites for Auto-Discovering the VMs on a vSphere/ESX Server 9

2.1.3 Pre-requisites for Obtaining the “Outside View” of VMs by connecting to a vCenter Server 9

2.1.4 Pre-requisites for Obtaining the “Outside View” of VMs by directly connecting to the target vSphere/ESX Server 10

2.1.5 Pre-requisites for Collecting Hardware Metrics 11

2.1.6 Pre-requisites for Obtaining the “Inside View” of Windows VMs, using the eG VM Agent 11

2.1.7 Pre-requisites for Obtaining the “Inside View” of VMs, without using the eG VM Agent 11

2.2 Assigning the ‘Read-Only’ Role to a Local/Domain User to vCenter 12

2.3 Creating a Special Role on vCenter and Assigning the Role to a Local/Domain User 18

2.4 Increasing the Memory Settings of the eG Agent that Monitors ESX Servers 27

2.4.1 Increasing the Memory Settings of the eG Agent on Windows 27

2.4.2 Increasing the Memory Settings of an eG Agent on Solaris 29

2.4.3 Increasing the Memory Settings of an eG Agent on Linux 29

2.5 Creating a New User with Read-Only Privileges to the ESX Server 30

2.6 Creating a Special Role on an ESX Server and Assigning the Role to a New User to the Server 36

2.7 Configuring the eG Agent to Collect Current Hardware Status Metrics 47

2.8 Configuring the Remote Agent to Obtain the Inside View of Windows VMs, using the eG VM Agent 57

2.8.1 Installing the eG VM Agent 58

2.8.2 Silent Installation of the eG VM Agent 61

2.8.3 Communication between the eG Agent and the eG VM Agent 62

2.8.4 Licensing of the eG VM Agent 63

2.8.5 Benefits of the eG VM Agent 63

2.9 Configuring Windows Virtual Machines to Support the eG Agent’s Inside View without the eG VM Agent 63

2.9.1 Enabling ADMIN$ Share Access on Windows Virtual Guests 63

2.9.2 Configuring Windows Firewalls to Allow File and Print Sharing 74

2.10 Configuring the eG Agent to Monitor NVIDIA Graphics Processing Units (GPUs) 77

2.10.1 Installing NVWMI as Part of the NVIDIA GPU Driver 79

2.10.2 Installing NVWMI from the Standalone Installer 81

2.10.3 Installing NVWMI via the Command Line 83

2.11 Monitoring Models offered by eG Enterprise for VMware Infrastructures 84

MONITORING VSPHERE/ESX SERVERS WITH VMS HOSTING SERVER APPLICATIONS 86
3.1 The Hardware Layer 86
3.1.1 Fans - ESX Test 87
3.1.2 Memory Sensors – ESX Test 93
3.1.3 Power – ESX Test 96
3.1.4 Processor Sensor - ESX Test 100
3.1.5 Temperature - ESX Test 104
3.1.6 Voltage – ESX Test 108
3.1.7 Software Components - ESX Test 113
3.1.8 Slots/Connectors - ESX Test 116
3.1.9 System – ESX Test 120
3.1.10 Battery - ESX Test 124
3.1.11 Chassis - ESX Test 128
3.1.12 Watchdog – ESX Test 132
3.1.13 Power Usage - ESX Test 135
3.1.14 Storage Hardware – ESX Test 140
3.2 The Operating System Layer 143
3.2.1 Datastores - ESX Test 144
3.2.2 Subsystems – ESX Test 154
3.2.3 Processors - ESX Test 161
3.2.4 CPU - ESX Test 164
3.2.5 Storage LUNs – ESX Test 168
3.2.6 Storage Activity – ESX Test 178
3.2.7 Memory – ESX Test 182
3.2.8 System - Console Test 191
3.2.9 Disk Space - Console Test 194
3.2.10 Disk Activity - Console Test 195
3.2.11 ManagementAgent - ESX Test 198
3.2.12 Storage Paths - ESX Test 201
3.2.13 Storage Adapters - ESX Test 207
3.2.14 Uptime - ESX Test 213
3.2.15 Grid GPUs – ESX Test 217
3.3 The Network Layer 232
3.3.1 Network - ESX Test 233
3.4 The Virtual Network Layer 239
3.4.1 Virtual Network Traffic Test 240
3.4.2 Port Groups – ESX Test 246
3.4.3 Virtual Switches Test 250

3.4.4 Distributed Port Groups Test 256
3.4.5 Distributed Switches Test 261
3.5 The VM Processes Layer 265
3.5.1 TCP Port Status Test 266
3.5.2 VmHostd Log Monitor Test 267
3.5.3 VmKernel Log Monitor Test 288
3.5.4 VmKWarning Log Monitor Test 291
3.5.5 VmMessages Log Monitor Test 294
3.5.6 ESX Server Status Test 297
3.6 The Outside View of VMs Layer 301
3.6.1 VM Details - ESX Test 302
3.6.2 Virtual Machines – ESX Test 327
3.6.3 Resource Pools – ESX Test 343
3.6.4 VM Connectivity Test 355
3.6.5 VM Datastores Test 359
3.6.6 VM Snapshots Test 368
3.6.7 VM Device Status Test 373
3.6.8 VM Virtual Disks Test 378
3.7 The Inside View of VMs Layer 382
3.7.1 Disk Activity - VM Test 385
3.7.2 Disk Space - VM Test 393
3.7.3 System Details - VM Test 398
3.7.4 Uptime - VM Test 406
3.7.5 Windows Memory - VM Test 412
3.7.6 Windows Network Traffic - VM Test 419
3.7.7 Network Traffic - VM Test 425
3.7.8 Tcp - VM Test 430
3.7.9 Tcp Traffic - VM Test 436
3.7.10 Handles Usage - VM Test 442
3.7.11 Windows Services - VM Test 448
3.7.12 Memory Usage - VM Test 455
3.7.13 Disk Alignment - VM Test 464
3.7.14 GPU - VM Test 471
3.7.15 Crash Details - VM Test 495
3.7.16 502
3.8 The Inside View Dashboard 502
3.8.1 The System Dashboard 502
3.8.2 The Network Dashboard 517

MONITORING VSPHERE/ESX SERVERS WITH VMS HOSTING DESKTOP APPLICATIONS 523
4.1 The Outside View of VMs layer 524
4.1.1 VM Details – ESX Test 525
4.1.2 Virtual Machines - ESX Test 540
4.1.3 VDI Logins – ESX Test 548
4.1.4 VDI Applications Test 554
4.2 The Inside View of Desktops Layer 561
4.2.1 Desktop's HDX Channel Test 562
4.2.2 Virtual Desktop Client's Network Connection Test 571
4.2.3 PCoIP Session - VM Test 578
4.2.4 Blast Session - VM Test 585
4.2.5 User Profile Management - VM 591
4.2.6 Domain Time Sync – VM Test 601
4.2.7 Browser Activity – VM Test 606
4.2.8 Personal vDisk – VM Test 614
4.2.9 Virtual Desktop Session Start-up Details Test 624
4.2.10 Virtual Desktop Sessions Details Test 645
TROUBLESHOOTING 652
5.1 Troubleshooting the Failure of the eG Remote Agent to Obtain the 'Inside View' of a Windows VM 652
5.1.1 Troubleshooting the Failure of the eG Remote Agent to Connect to or Report Measures for Linux Guests 655
5.2 Troubleshooting the Failure of the eG Remote Agent to Connect to or Report Measures for Windows 2008 or Windows Vista VMs 658
HOW TO STOP THE 'AGENTLESS MONITORING' OF VSPHERE/ESX SERVERS AND VMS? 659
6.1 How to Stop the "Agentless Monitoring" of a Single VM on an ESX Server? 659
6.2 How to Stop the "Agentless Monitoring" of all the VMs on an ESX server? 661
6.3 How to Stop the "Agentless Monitoring" of an ESX server and its VMs? 662
THE VM DASHBOARD 664
7.1 The Summary Tab Page 667
7.3 The VC Tab Page 689
7.4 The Hosts Tab Page 692
7.5 The Datastores Tab Page 700
7.6 The Current Events Tab Page 703
7.7 The Resource Pools Tab Page 709
FREQUENTLY ASKED QUESTIONS (FAQ) 711

Table of Figures

Figure 1.1: The architecture of an ESX server 2

Figure 1.2: A sample virtual desktop infrastructure 3

Figure 2.1: Agentless monitoring of vSphere/ESX servers 7

Figure 2.2: The Remote agent communicating with vCenter servers to extract metrics related to vSphere/ESX servers they are managing 8

Figure 2.3: The login screen 13

Figure 2.4: The VI Client console 13

Figure 2.5: List of vCenter users and roles assigned to them 14

Figure 2.6: Selecting the Add Permission option 14

Figure 2.7: Clicking on the Add button 15

Figure 2.8: Selecting the user for whom a role is to be assigned. 16

Figure 2.9: Assigning the Read-Only role to the user 17

Figure 2.10: The new user and the role assigned to the user being displayed 17

Figure 2.11: Connecting to vCenter 18

Figure 2.12: Clicking on the Administration icon 19

Figure 2.13: Viewing the roles available on the vCenter 20

Figure 2.14: Selecting Add from the shortcut menu 21

Figure 2.15: Creating a new role on a vCenter server (prior to v 4.1) with the 'View and Terminate Sessions' privilege 22

Figure 2.16: Creating a new role on vCenter 4.1 (and above) with the 'View and stop sessions' privilege 22

Figure 2.17: The Roles tab page listing the newly created role 23

Figure 2.18: Details of users to vCenter 24

Figure 2.19: Selecting the Add Permission option 25

Figure 2.20: Selecting a local user 26

Figure 2.21: Adding the chosen user 26

Figure 2.22: Assigning the Read-Only role to a chosen user 27

Figure 2.23: Increasing the memory setting of an eG agent on Windows by editing the debugon.bat file 28

Figure 2.24: Increasing the memory settings of an eG agent on Windows by editing the debugoff.bat file 28

Figure 2.25: Increasing the memory setting of an eG agent on Solaris by editing the starta file 29

Figure 2.26: Increasing the memory setting of an eG agent on Linux by editing the start_agent file 30

Figure 2.27: Connecting to an ESX host 31

Figure 2.28: The VI Client console 31

Figure 2.29: Selecting the Add option 32

Figure 2.30: Adding a new user 32

Figure 2.31: The newly added user name displayed in the list of users to the ESX host 33

Figure 2.32: Figure 2.32: Selecting the Add Permission option 33

Figure 2.33: Clicking on the Add button 34

Figure 2.34: Selecting the user for whom a role is to be assigned. 34

Figure 2.35: The user name displayed in the Users box 35

Figure 2.36: Assigning the Read-Only role to the user 35
Figure 2.37: The new user and the role assigned to the user being displayed 36
Figure 2.38: The VI Client console - Users & Groups tab page 37
Figure 2.39: Selecting the Add option 38
Figure 2.40: Adding a new user 39
Figure 2.41: The newly added user name displayed in the list of users to the ESX host 40
Figure 2.42: The list of roles 40
Figure 2.43: Selecting the Add option to add a new role 41
Figure 2.44: Creating a new role on an ESX server 3/3.5 with the 'View and Terminate Sessions' privilege 42
Figure 2.45: Creating a new role on vSphere 4.1 (and above) with the 'View and stop sessions' privilege 43
Figure 2.46: The newly created role appended to the list of roles 43
Figure 2.47: Figure 2.47: Selecting the Add Permission option 44
Figure 2.48: Clicking on the Add button 45
Figure 2.49: Selecting the user for whom a role is to be assigned. 45
Figure 2.50: ssigning the new role to the user 46
Figure 2.51: The new user and the role assigned to the user being displayed 46
Figure 2.52: Connecting to vCenter 47
Figure 2.53: Clicking on the Administration icon 48
Figure 2.54: Viewing the roles available on the vCenter 49
Figure 2.55: Selecting Add from the shortcut menu 50
Figure 2.56: Creating a new role on a vCenter server with the 'Change Settings' privilege 51
Figure 2.57: The VI Client console - Users & Groups tab page 52
Figure 2.58: Selecting the Add option 53
Figure 2.59: Adding a new user 54
Figure 2.60: The newly added user name displayed in the list of users to the ESX host 55
Figure 2.61: The list of roles 55
Figure 2.62: Selecting the Add option to add a new role 56
Figure 2.63: Creating a new role on a vSphere server with the 'Change Settings' privilege 57
Figure 2.64: Welcome screen of the eG VM Agent installation wizard 58
Figure 2.65: Accepting the license agreement 59
Figure 2.66: Specifying the install directory of the eG VM Agent 59
Figure 2.67: Specifying the VM agent port 60
Figure 2.68: A summary of your specifications 60
Figure 2.69: Finishing the installation 61
Figure 2.70: The ADMIN$ share does not exist 64
Figure 2.71: Admin$ share pre-exists 65
Figure 2.72: Creating the ADMIN$ share 65
Figure 2.73: Clicking the Add button 66

Figure 2.74: Selecting the administrative user to whom access rights are to be granted 66
Figure 2.75: The administrator account granted access permissions 67
Figure 2.76: Defining the Security settings for the ADMIN$ share 68
Figure 2.77: Adding the administrator account 68
Figure 2.78: The Administrator account in the Security list 69
Figure 2.79: Selecting the Share option from the shortcut menu 70
Figure 2.80: Clicking on Advanced Sharing 71
Figure 2.81: Enabling the ADMIN$ share 71
Figure 2.82: Clicking on the Add button 72
Figure 2.83: Allowing a domain administrator to access the folder 72
Figure 2.84: Allowing full access to the local/domain administrator 73
Figure 2.85: Applying the changes 73
Figure 2.86: Selecting the guest OS 74
Figure 2.87: Opening the Windows Firewall 75
Figure 2.88: The General tab of the Windows Firewall dialog box 75
Figure 2.89: Deselecting the 'Don't allow exceptions' check box 76
Figure 2.90: Enabling 'File and Printer Sharing' 76
Figure 2.91: Opening ports 77
Figure 2.92: Custom check box 80
Figure 2.93: Selecting the NVIDIA WMI check box 81
Figure 2.94: NVIDIA Enterprise Management Toolkit (NVWMI) 82
Figure 2.95: NVIDIA WMI minimum driver requirements 83
Figure 2.96: Checking the Control Panel for NVIDIA WMI 84
Figure 3.1: The VMware vSphere ESX server layer model 86
Figure 3.2: The tests mapped to the Hardware layer 87
Figure 3.3: Adding the details of a vCenter server 92
Figure 3.4: The tests mapped to the Operating System layer of the VMware vSphere ESX model 144
Figure 3.5: The detailed diagnosis of the Datastore availability measure 154
Figure 3.6: The detailed diagnosis of the Number of LUNs for physical storage measure 154
Figure 3.7: The detailed diagnosis of the VMs descriptor of the Subsystems - Esx Test 161
Figure 3.8: The detailed diagnosis of the Virtual CPU utilization of console measure 193
Figure 3.9: The detailed diagnosis of the Free memory of console OS measure 194
Figure 3.10: Dedicated GPU Technology 218
Figure 3.11: Shared vGPU Technology 219
Figure 3.12: The tests mapped to the Network layer 232
Figure 3.13: The Virtual Network Layer 240
Figure 3.14: The detailed diagnosis of the Data received measure 245
Figure 3.15: The detailed diagnosis of the Packets received measure 245

Figure 3.16: The detailed diagnosis of the Data transmitted measure 246
Figure 3.17: The detailed diagnosis of the Packets transmitted measure 246
Figure 3.18: The detailed diagnosis of the Total VMs in port group measure 250
Figure 3.19: The detailed diagnosis of the PoweredOff VMs in PortGroup measure 250
Figure 3.20: The detailed diagnosis of the Portgroups in vSwitch measure 255
Figure 3.21: The detailed diagnosis of the Virtual Machine in vSwitch measure 256
Figure 3.22: The detailed diagnosis of the Physical NICs in vSwitch measure 256
Figure 3.23: The detailed diagnosis of the Physicals NICs up in vSwitch measure 256
Figure 3.24: The detailed diagnosis of the Physicals NICs down in vSwitch measure 256
Figure 3.25: The tests mapped to the VM Processes layer 266
Figure 3.26: Connecting to vCenter 271
Figure 3.27: Clicking on the Administration icon 271
Figure 3.28: Viewing the roles available on the ESX server 272
Figure 3.29: Creating a new role and assigning Diagnostics privilege to it 273
Figure 3.30: The Roles tab page listing the newly created role 274
Figure 3.31: Details of users to the ESX server 275
Figure 3.32: Creating a new user 276
Figure 3.33: The Permissions tab page 277
Figure 3.34: Adding a permission 278
Figure 3.35: Selecting the role to be assigned to the new user 278
Figure 3.36: Selecting the newly created user 279
Figure 3.37: Adding the chosen user 279
Figure 3.38: Assigning the Diagnostics role to a chosen user 280
Figure 3.39: Connecting to vCenter 281
Figure 3.40: The VMware Infrastructure Client console 281
Figure 3.41: Adding a new role 282
Figure 3.42: Providing the details of the new role 283
Figure 3.43: The Permissions tab page 284
Figure 3.44: Selecting the Add Permission option 285
Figure 3.45: Adding a user 285
Figure 3.46: Selecting a local/domain user 286
Figure 3.47: Assign the newly created role to the user 287
Figure 3.48: The Permissions tab page displaying the chosen user and the role assigned to him/her 288
Figure 3.49: The tests associated with the Outside View of VMs layer 302
Figure 3.50: Configuring a VM test 325
Figure 3.51: The VM user configuration page 325
Figure 3.52: Adding another user 326
Figure 3.53: Associating a single domain with different admin users 326

Figure 3.54: The test configuration page displaying multiple domain names, user names, and passwords 327
Figure 3.55: How VMotion works? 327
Figure 3.56: The detailed diagnosis of the Guests powered on measure of a VMware vSphere ESX server 335
Figure 3.57: The “outside view” measures for the nj_printserver VM 336
Figure 3.58: The topology of the ESX server, ESX35-KS 337
Figure 3.59: The topology of the ESX3i server 337
Figure 3.60: Selecting the Migrate option from the shortcut menu 338
Figure 3.61: Selecting the resource pool to which the VM is to be moved. 339
Figure 3.62: Setting the migration priority 339
Figure 3.63: Finishing the migration 340
Figure 3.64: Progress of the migration 340
Figure 3.65: Investigating the minor problem with the ESX server, 192.168.10.179 341
Figure 3.66: Detailed diagnosis of the Added guests measure revealing the addition of a guest to the ESX server, ESX3i 342
Figure 3.67: The topology of ESX3i server with the Print server application 342
Figure 3.68: The topology of the ESX35_KS server without the Print server application 343
Figure 3.69: The detailed diagnosis of the VM Snapshots test 373
Figure 3.70: A list of guest operating systems on an ESX server host and their current state 383
Figure 3.71: The tests associated with the Inside View of Vms layer of a VMware vSphere ESX server 385
Figure 3.72: The detailed diagnosis of the Percent virtual busy measure 392
Figure 3.73: The top 10 CPU consuming processes 405
Figure 3.74: The detailed diagnosis of the Free memory measure listing the top 10 memory consuming processes 406
Figure 3.75: The detailed diagnosis of the Handles used by processes measure 448
Figure 3.76: The detailed diagnosis of the Processes using handles above limit in VM measure 448
Figure 3.77: The SAN, VMFS, and NTFS blocks 464
Figure 3.78: Unaligned partitions 465
Figure 3.79: Aligned partitions 465
Figure 3.80: The detailed diagnosis of the Disk partition alignment status measure 471
Figure 3.81: Dedicated GPU Technology 472
Figure 3.82: Shared vGPU Technology 473
Figure 3.83: The tests mapped to the Virtual Servers layer of a VMware vSphere ESX server 491
Figure 3.84: Measures pertaining to a chosen guest on a VMware vSphere ESX server 492
Figure 3.85: Live graph of the VMware vSphere ESX server 493
Figure 3.86: Adding a server application to a virtual environment 494
Figure 3.87: Depicts the applications that have been deployed on the guest OS of an ESX server 495
Figure 3.88: The System Dashboard 503
Figure 3.89: The System Overview Dashboard 504
Figure 3.90: Viewing the current system alerts in the System Overview dashboard 505
Figure 3.91: History of events of a particular priority 505

Figure 3.92: The layer model page that appears upon clicking on a key performance indicator 506
Figure 3.93: An enlarged measure graph in the System Overview Dashboard 507
Figure 3.94: An enlarged top-n bar chart 507
Figure 3.95: Detailed diagnosis displaying the top memory-consuming processes executing within a VM 508
Figure 3.96: The Dashboard of the CPU Subsystem 509
Figure 3.97: The enlarged CPU usage by top processes graph 510
Figure 3.98: Figure 3.98: The detailed diagnosis of the top processes by CPU usage 510
Figure 3.99: The Dashboard of the Memory Subsystem 511
Figure 3.100: The enlarged memory usage graph 512
Figure 3.101: The detailed diagnosis displaying the top-10 memory-intensive processes on the VM 513
Figure 3.102: The Dashboard of the Disk Subsystem 514
Figure 3.103: The enlarged Disk busy by top processes graph 515
Figure 3.104: The detailed diagnosis displaying the top I/O intensive processes on the VM 516
Figure 3.105: The Uptime Dashboard 516
Figure 3.106: The Network Overview Dashboard 518
Figure 3.107: The History of network events 519
Figure 3.108: Clicking on a Key Performance Indicator 519
Figure 3.109: An enlarged network traffic graph 520
Figure 3.110: Dashboard of the Network Subsystem 521
Figure 3.111: Dashboard of the Tcp Subsystem 522
Figure 4.1: The layer model of a VMware vSphere VDI server 523
Figure 4.2: The tests associated with the Outside View of VMs layer of a VMware vSphere VDI server 525
Figure 4.3: The detailed diagnosis of the Guests powered on measure 548
Figure 4.4: The detailed diagnosis of the Guests with users measure 548
Figure 4.5: The current state of the desktops configured on the ESX VDI (i) server host that is monitored 561
Figure 4.6: The tests associated with the Virtual Desktop layer of a VMware vSphere VDI 562
Figure 4.7: The detailed diagnosis of the Running browser instances measure 614
Figure 4.8: The detailed diagnosis of the Recent web sites measure 614
Figure 4.9: Citrix user logon process 624
Figure 4.10: The measures pertaining to a particular desktop 644
Figure 4.11: Live graph for VMware vSphere VDI server 645
Figure 6.1: Selecting the Remove option 660
Figure 6.2: Confirming the removal of the eG VM agent and its components 660
Figure 6.3: Disabling VM-related tests 661
Figure 6.4: Selecting the ESX server to be unmanaged 662
Figure 6.5: Unmanaging the ESX server 663
Figure 7.1: The VM Dashboard 665
Figure 7.2: The Virtual Infrastructure tree 666

Figure 7.3: Display settings for the tree-view 666
Figure 7.4: The Configuration Settings page 667
Figure 7.5: The Summary tab page if the global Zones node is clicked 668
Figure 7.6: The names of physical vSphere/ESXi servers displayed in a pop-up 669
Figure 7.7: The Hosts tab page appears when the 'Physical servers' label in the Summary tab page is clicked 670
Figure 7.8: The VMs tab page that appears when the 'VMs' label in the Summary tab page is clicked 670
Figure 7.9: The VMs tab page listing only powered-on VMs 671
Figure 7.10: Moving the mouse pointer over an alarm priority that corresponds to a physical server 672
Figure 7.11: The layer model of a physical server indicating the layer where the root-cause of a problem lies 672
Figure 7.12: The alarms of a particular priority corresponding to a virtual machine 673
Figure 7.13: The 'inside view' of a VM clicked on in the Virtual Machines section of the Summary tab page 673
Figure 7.14: Adding a new measure to the Top Servers by list 674
Figure 7.15: Adding a new measure to the Top VMs by list 675
Figure 7.16: The Dashboard configuration page 676
Figure 7.17: The layer model of a CPU-intensive physical server revealing the processor that is excessively consuming physical CPU 677
Figure 7.18: The expanded resource usage graph of a physical server 678
Figure 7.19: The layer model of a CPU-intensive VM revealing the resource usage of the VM 678
Figure 7.20: The expanded resource usage graph of a VM 679
Figure 7.21: The Summary tab page if a particular zone is chosen from the tree-structure in the left panel 680
Figure 7.22: The Summary tab page of the node representing a virtual server-type 680
Figure 7.23: The Summary tab page if the 'VMware vCenter' node is clicked 682
Figure 7.24: The Summary tab page if the node representing a particular vCenter server is clicked in the tree 683
Figure 7.25: Expanding a graph in the Graphs section of the Summary tab page 684
Figure 7.26: The Summary tab page if a folder in the tree is clicked 684
Figure 7.27: The Summary tab page of a datacenter chosen from the tree-structure 685
Figure 7.28: The Summary tab page of the cluster sub-node 686
Figure 7.29: The Summary tab page for an ESX host 687
Figure 7.30: The Summary tab page of a resource pool 688
Figure 7.50: The VC tab page if the global Zones node is clicked 689
Figure 7.51: Adding a new measure to the VC tab page 690
Figure 7.52: The layer model, tests, and measures pertaining to the vCenter server 691
Figure 7.53: The VC tab page if the 'VMware vCenter' node is chosen from the tree 692
Figure 7.54: The VC tab page for a particular vCenter server chosen from the tree 692
Figure 7.55: Zooming into the performance of a VM from the VM dashboard 693
Figure 7.56: The Timeline window 694
Figure 7.57: he Hosts tab page for a virtual component type in the tree 695
Figure 7.58: The Hosts tab page if the 'VMware vCenter' node is chosen from the tree 695
Figure 7.59: The Hosts tab page if a particular vCenter server is chosen from the tree 696

Figure 7.60: The Hosts tab page for a folder selected from the tree 697
Figure 7.61: The Hosts tab page for a datacenter chosen from the tree 697
Figure 7.62: The Hosts tab page if the cluster sub-node is clicked on the tree 698
Figure 7.63: The VMs tab page for a particular virtual host selected from the tree 699
Figure 7.64: The layer model of the virtual host 699
Figure 7.65: The Datastore tab page of a folder node 700
Figure 7.66: The Datastores tab if a particular datacenter is chosen from the tree 701
Figure 7.67: The layer model of the vCenter server that appears when a datastore is clicked on in the Datastores tab page 702
Figure 7.68: The Datastores tab if a particular ESX host is selected from the tree 702
Figure 7.69: The Current Events tab for the global Zones node 703
Figure 7.70: The Events tab page for a virtual component type in the tree 704
Figure 7.71: The Events tab page for a vCenter server chosen from the tree 705
Figure 7.72: The Current Events tab page of a folder 706
Figure 7.73: The Events tab page for a datacenter chosen from the tree 706
Figure 7.74: The Current Events tab page for a cluster 707
Figure 7.75: The Events tab page for a particular virtual host selected from the tree 708
Figure 7.76: The problem layer, test, and measure of the problem component 709
Figure 7.77: The Resource Pools tab page 710
Figure 7.78: The outside view of a VM under the resource pool 710
Figure 7.79: The metrics related to the child resource pool 710

1

Introduction

In an era where the 24x7 availability and rapid delivery of business-critical services is crucial to the success of an enterprise, the IT infrastructure, which underlies such service offerings, is often "fine-tuned" to meet these high service-quality expectations in an efficient and cost-effective manner. As a result, we have resource-slim thin clients replacing the resource-intensive fat clients; user-friendly web front-ends are fast substituting primitive interfaces, and the most recent addition to this list is "virtualization". Virtualization allows multiple virtual machines, with heterogeneous operating systems to run in isolation, side-by-side on the same physical machine. Each virtual machine has its own set of virtual hardware (e.g., RAM, CPU, NIC, etc.) upon which an operating system and applications are loaded. The main benefits that enterprises can obtain from a virtualization solution include:

- realizing production server consolidation and containment by running software applications on fewer servers;
- achieving better utilization of existing resources by dynamically allocating resources across operating systems and applications, based on current demands;
- providing advanced business continuity protection and delivering high availability for critical applications at a lower cost;
- re-hosting legacy applications on new hardware for improved reliability;
- streamlining disparate software test and development environments with multiple operating systems loaded on the same hardware;

Among the many virtualization solutions available, the VMware® Virtual Infrastructure solutions are one of the most popular. At the heart of this virtualization solution is the VMware ESX server. The ESX server installs on the "bare metal" and allows multiple unmodified operating systems and their applications to run in virtual machines that share physical resources. The operating system running on the bare metal is referred to as the physical server or virtual host, and the individual operating systems hosted by the ESX server as guests/virtual machines. From simple desktop tools to complex servers, the virtual guests can host a wide variety of applications.

The following illustration shows the main components of an ESX Server host.

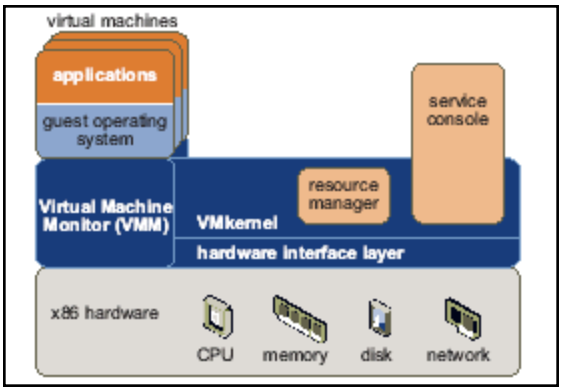


Figure 1.1: The architecture of an ESX server

The VMkernel is an operating system that runs directly on the ESX Server host; it controls and manages most of the physical resources on the hardware, such as the memory, processors, CPU memory disk network, and storage and networking controllers. The resource manager partitions the physical resources of the underlying server. It uses a proportional share mechanism to allocate CPU, memory, and disk resources to virtual machines that are powered on. Users can specify shares, reservations, and limits for each virtual machine. The resource manager takes that information into account when it allocates CPU and memory to each virtual machine. The VMkernel hardware interface hides hardware differences from ESX Server (and virtual machine) users. It enables hardware-specific service delivery and includes device drivers, and Virtual Machine File System® (VMFS®). The virtual machine monitor (VMM) is responsible for virtualizing the CPUs. When a virtual machine starts running, control transfers to the VMM, which begins executing instructions from the virtual machine. The transfer of control to the VMM involves setting the system state so that the VMM runs directly on the hardware.

1.1 Virtualization Deployment Models

Virtualization is being deployed in a couple of different ways. Early adoption of virtualization was for deploying multiple server applications on the same physical system. For instance, a Citrix server, a Web server, and an Oracle database server can all be operated on three virtual servers all of which are hosted on the same physical server. Such server virtualization was also mandated by applications that could not make use of higher memory availability on the servers, and as a result, it became necessary to host multiple application instances, one per virtual server.

A more recent trend is the use of virtualization as an alternative to traditional server-based computing technologies. In these virtual desktop infrastructures (VDI), virtual desktops (rather than servers) are published on the virtual hosts, and users connect to virtual desktops using the remote desktop protocol (RDP) or equivalent mechanism, and run their client applications. In this scenario, each user is assigned to a virtual desktop and can run his/her applications within that desktop. Figure 1.2 depicts how a virtual desktop infrastructure functions. Typically, simple brokering techniques are used to enable end-users to connect to a virtual machine. Here, end-users are provided a web address on the corporate network for the connection broker. After authenticating, the connection broker provides a list of available resources to the end user. The end-user establishes a connection directly to the virtual machine using RDP.

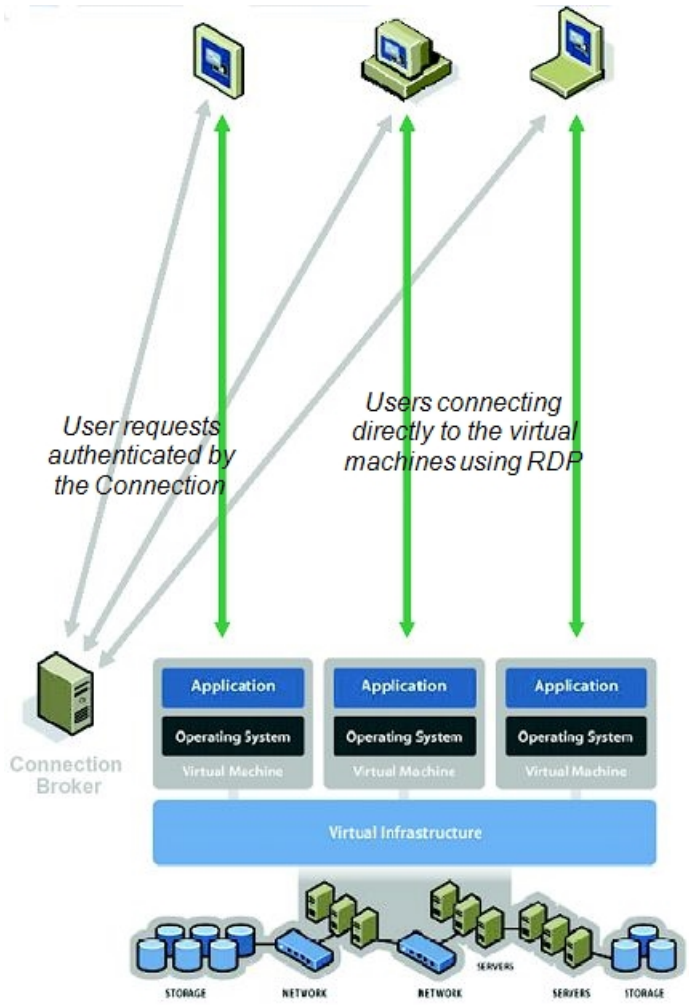


Figure 1.2: A sample virtual desktop infrastructure

1.2 Challenges in Monitoring Virtual Environments

While virtualization technologies provide businesses with the ability to do more with less, using virtualization to consolidate the number of physical servers does not mean your management duties shrink proportionally. For example, if a company that runs 300 operating systems on 300 servers reduces that to running 10 virtual servers on each of 30 physical servers, it still has 300 OSs and related software applications, to manage!

In reality, for a technology that is supposed to make computing easier, virtualization is becoming quite complicated to monitor and manage. First, besides the physical server, each of the guests running on the server needs to be monitored. Conventional monitoring approaches involve deploying agents (or agentless monitors) for each guest operating system. The additional cost of deploying an agent for each guest operating system is likely to make this approach impractical for virtual desktop environments where a single physical server is used to host tens of desktops. The resource usage (one agent per operating system) is another disadvantage of this approach.

Furthermore, an agent deployed on each guest virtual machine (VM) can provide an indication of the comparative resource usage levels of all the applications executing within that guest VM only. That is, this approach cannot provide an indication of how the guest VM itself is performing relative to other guest VMs on the same physical server - e.g., how much of the physical CPU of a server is the guest VM consuming? Is there memory contention among the guest VMs? etc. Since multiple guest VMs share the physical resources of the server, even one malfunctioning guest VM (or one malfunctioning application executing on a guest VM) can impact the performance seen by all the other guest VMs hosted on the same physical server.

Resource contention may also occur because of the way an administrator has chosen to allocate resources to the guest VMs. For example, the disk space allocated to a guest VM or the CPU allocated to a guest VM may not be sufficient for the workload that the guest is supporting.

At the same time, it is important to also identify situations when the physical server itself is overloaded. In such situations, administrators should be able to identify whether the excessive loading of the server is because of a malfunctioning VM, or because of genuine load on the physical server. In the latter case, by comparing the relative loading of the guest VMs, administrators can determine how to load balance the guest VMs across different physical servers to ensure optimal usage of the infrastructure.

The different virtualization deployment models have different requirements. For instance, while the server application virtualization approach typically involves a smaller number of VMs running on a physical server, in the virtual desktop approach, tens of desktop VMs run on a physical server. The scale of deployment of desktops means that it is not practically feasible to deploy an agent for monitoring each desktop. Since the number of servers hosting applications on a physical server is likely to be smaller, an agent-based approach for monitoring is more feasible in this scenario.

While in-depth monitoring of each of the applications is important in the server application virtualization approach, in the virtual desktop approach, since only client applications are executed on the desktop, monitoring of the desktop need not be as in-depth as in the server application virtualization context. Furthermore, in a virtual desktop environment, it is essential to identify which guest a user is logging on to, for how long the user was logged in, and what applications he/she used. This information is critical for planning the capacity of the virtual desktop environment. Notice should also be taken of the fact that in a virtual desktop environment, virtual desktops may come and go off dynamically (e.g., as a user logs on and logs off, respectively), whereas in a server application virtualization approach, the guest operating systems are likely to be more static (i.e., come on and off less frequently).

To effectively monitor and plan the capacity of virtual desktop environments, it is also important that the activity of individual users be monitored. At any point, the administrator should be able to determine which of many servers a specific user is logged on to, when did the user logon, and what application(s) is the user accessing. This information may have to be available on a historical basis to allow for auditing of virtual desktop accesses. The duration and frequency of user accesses must also be monitored to determine who the most frequent or the heaviest users are.

Since virtualization is being deployed in different ways - e.g., using EMC's VMware infrastructure, Microsoft's Virtual Server, AIX LPAR, Solaris Zones, etc., it is important to design the monitoring solution to be easily extensible to accommodate virtual environments besides those based on VMware.

From the above discussion, it is clear that for effectively monitoring a virtual infrastructure, it is important to be able to:

- assess the resource usage on the host and identify resource bottlenecks;
- compare the resource usage of the different guest VMs to identify malfunctioning guests;
- identify resource constraints in each guest VM;
- pin-point malfunctioning applications running on each guest VM;
- handle the diverse monitoring and management requirements of virtual desktops and hosted server applications;
- correlate the performance of the physical server, the guest VMs, and the applications executing on these VMs;
- be extensible to support not just VMware among the virtualization platforms.

1.3 Limitations of Existing VMware Monitoring Solutions

Many solutions for VMware monitoring rely on the vCenter component of the VMware infrastructure to report on the performance of the host and the virtual guests. Since not all infrastructures may have the VMware vCenter available, the reliance on vCenter limits the utility of this approach. Likewise, the granularity and frequency of the monitoring are limited by the capabilities of VMware vCenter. VMware APIs or SNMP are some of the other ways of collecting performance information on the host and the guest operating systems.

These solutions provide an idea of the degree of loading on the host operating system as well as a comparison of the degree of loading that is caused by each of the guest operating systems. However, this only addresses part of the requirements we had outlined for a monitoring solution for virtual solutions. VMware APIs or SNMP interfaces do not provide any information about the resource constraints within each operating system. Hence, a monitoring solution that uses the VMware APIs alone will not be able to determine bottlenecks within a VM guest operating system. For example, consider a case where the space allocated to a VM guest is partitioned across two logical disks. If one of the disks were to become full, monitoring from the ESX server will not be able to report this problem situation, which could be adversely impacting the performance seen by the guest applications.

Many of the existing virtualization monitoring solutions are also not tailored to address the unique requirements of virtual desktop environments. Finally, most of these solutions view the VMware server infrastructure as a silo that needs to be monitored, but they do not consider the inter-dependencies between the physical server and VM guest operating systems and between the applications hosted on these operating systems on the same server. Correlation of performance and usage information across the applications involved in supporting a business service, and identifying the root-cause of a problem by analyzing the virtual machine dependencies of each application involved in supporting the business service is a key value-addition for any monitoring solution for virtual environments.

2

How does eG Enterprise Monitor VMware Environments?

To monitor virtual infrastructures, eG Enterprise uses a patented In-N-Out monitoring approach that is designed to address the requirements outlined in Chapter 1.1. The eG Single Agent not only monitors the VM kernel, but also uses the VMware ESX APIs and commands to report real-time resource usage of each of the guest VMs. The metrics collected report on the percentage of the ESX server's resources that each the VMs on the server are using - i.e., the relative loading of the guest VMs. This represents the view of how a guest VM and its applications are doing - from the "outside" - i.e., from outside the guest VM.

In addition, the eG agent also connects to each guest VM that is currently powered on and determines the guest OS version, the name(s) of the users who are logged on (in the virtual desktop scenario), and the resource usage of the guest and the applications running inside the guest (as seen from within the guest operating system). This represents the view from within the guest operating system - i.e., the "inside" view.

With the In-N-Out monitoring approach, eG Enterprise allows administrators of VMware environments to answer several key questions. Some of these include:

- How many guest VMs are running on each VMware server, what is the IP address of each of guests, what operating system is each guest running, and when was the guest powered on?
- Which user is logged in to the guest VM, and when did he/she login?
- How much memory is allocated to each guest and does each guest VM have sufficient free memory?
- Does the ESX server have sufficient memory available to support the guest VMs that it is hosting?
- What is the CPU utilization of the ESX server and which of the guest VMs is taking up excessive CPU?
- Which application(s) running on each of the guest VMs is taking CPU, memory, and disk resources?
- Is there sufficient disk space in each of the disk partitions of the guest operating system?
- Which of the guests is seeing the highest and lowest network traffic?
- Is there excessive queuing for disk access on any of the guest VMs?
- What are peak usage times of the virtual desktops?
- Who are the most resource intensive users of a virtual desktop environment?

The eG agent that collects the 'inside' and 'outside' views of performance can be deployed on a remote system running a Windows/Linux/Solaris operating system. In other words, eG Enterprise supports only *agentless* monitoring of vSphere/ESX servers.

The **agentless approach** to monitoring vSphere/ESX servers involves the following:

- Deploying the eG agent on a remote system running Microsoft Windows or Linux or Solaris;
- Configuring the remote eG agent to communicate with the Web Services interface of the ESX server using HTTP/HTTPS;
- Configuring the remote eG agent to collect performance metrics from each of the guest VMs configured on the ESX server using SSH/WMI; by default, the eG agent uses SSH/WMI (depending upon the virtual OS to be monitored) to communicate remotely with the virtual machines on the ESX server (see Figure 2.1) and collect metrics. To establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM, which allows the eG agent on the service console to collect "inside view" metrics from the VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**.

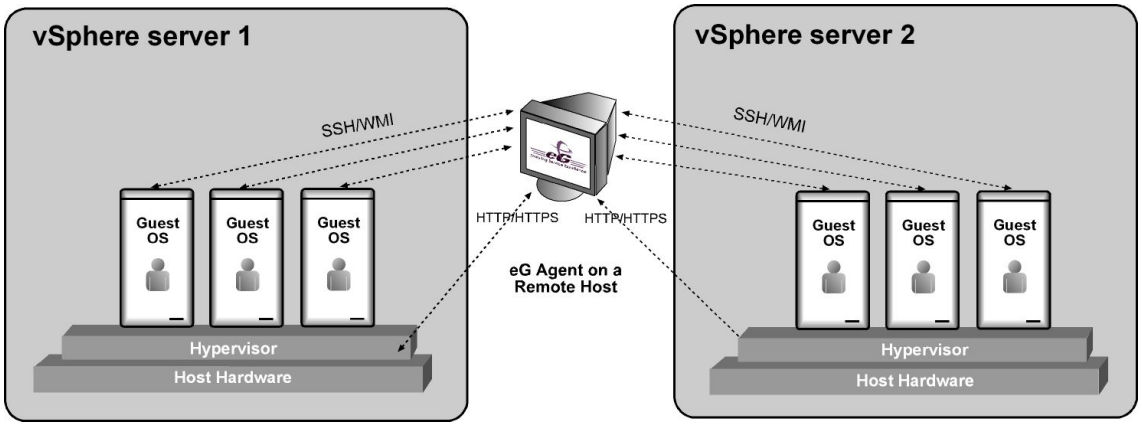


Figure 2.1: Agentless monitoring of vSphere/ESX servers

ESX 3.0, 3.5, ESXi, and vSphere servers can all be monitored using this approach.

The key advantages of agentless monitoring for ESX servers are:

- There is no need to install an eG agent on the ESX server or on any of the guests; related processing overheads, though minimal, can thus be saved.
- A single remote agent can monitor multiple ESX servers in the infrastructure.

Alternatively, instead of configuring the remote agent to communicate with each ESX server for guest discovery and data collection, you can simply configure the remote agent to talk to one/more vCenters in the environment.

Note:

If the vSphere/ESX servers in the environment are in the lockdown mode, then, the remote agent will not be able to monitor them by connecting to them directly. In such a case, vSphere/ESX servers can be monitored via VMware vCenter only.

For this purpose, it would suffice to configure the tests with the credentials of a user with **Administrator** or **Virtual Machine Administrator** privileges. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges, and assign this role to a local/domain user.

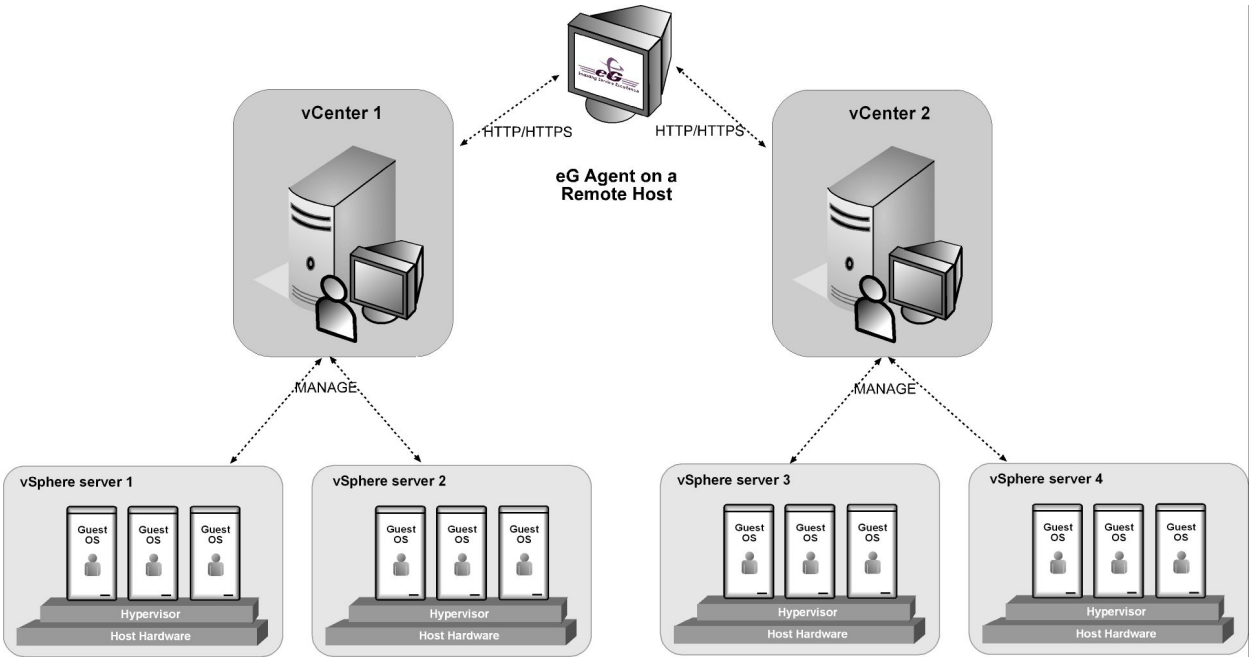


Figure 2.2: The Remote agent communicating with vCenter servers to extract metrics related to vSphere/ESX servers they are managing

Likewise, a few more pre-requisites need to be fulfilled to enable the remote agent to collect metrics remotely from the target host and its VMs. These requirements have been detailed in Section **2.1** of this document.

Note:

If an ESX Server 3.0 is monitored in an 'agentless manner', then the eG remote agent assigned to such an ESX server cannot monitor the server or its VMs directly - instead, you will have to configure the agent to monitor the server via vCenter. Therefore, while configuring the tests for such a server, ensure that the tests are configured to perform monitoring via VMware vCenter.

2.1 Pre-requisites for Monitoring VMware Infrastructures

2.1.1 General Pre-requisites

- Enable the remote agent to communicate with the eG manager port (default: 7077).
- Ensure that the remote agent, if installed on a Windows host, runs using *domain administrator* privileges.

- If VMs running on multi-byte operating systems are to be monitored (eg., Windows Japanese), then the remote agent monitoring such VMs should also run on a multi-byte operating system.
- 32-bit VMs that are to be monitored in an agentless manner should be configured with at least 2 GB RAM, and 64-bit VMs require at least 4 GB RAM. If more than four ESX servers are being monitored in an agentless manner, then the RAM capacity of the VMs should be increased proportionately.

2.1.2 Pre-requisites for Auto-Discovering the VMs on a vSphere/ESX Server

The **VMware tools** (vmtools) must be installed on all guest operating systems hosted on an vSphere/ESX server. Using vmtools, the remote agent determines the IP addresses of the guest VMs and the operating systems that they are configured with. If the eG Enterprise monitor shows "N/A" against the IP address field or the operating system type of any VM, this is usually a good indicator that vmtools has not been installed on that VM.

2.1.3 Pre-requisites for Obtaining the "Outside View" of VMs by connecting to a vCenter Server

- Ensure that the remote agent has IP connectivity to the vCenter server.
- Ensure that the remote agent has web access to the **WEBPORT** (port 80/443) configured for the vCenter server.
- All the tests that the remote agent executes should be configured with the IP address of vCenter, and the name and password of a user with **Administrator** or **Virtual Machine Administrator** privileges to vCenter. However, if owing to security constraints, you cannot expose the credentials of the administrator, then, you can configure the tests with the credentials of a user with **'Read-only'** privileges to vCenter. For this purpose, you can assign the **Read-only** role to a local/domain user to vCenter. To know how to do this, refer to Section 2.2 of this document.

  vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop a remote agent, sessions currently in use by the remote agent will remain open for this timeout period until vCenter times out the session. If the remote agent is restarted within the timeout period, it will open a new set of sessions. If you want the remote agent to close already existing sessions on vCenter before it opens new sessions, then, when monitoring a vSphere/ESX server via vCenter, you can optionally configure the tests with the credentials of a user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission) on vCenter. To know how to grant this permission to a user to vCenter, refer to Section Section **2.3** of this document.

  When the remote agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the remote agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the remote agent starts/restarts.

- VI APIs are memory-intensive, and therefore, while monitoring ESX servers using vCenter, the default memory setting of 128 MB will not be sufficient for the eG agent to pull out the required metrics. The maximum memory setting of the eG agent should hence be increased to any value between 256 MB and 1024 MB, but the exact value will depend upon the number of ESX servers being managed by the vCenter in your environment. Refer to Section **2.4** to know how to alter the memory settings of the eG agent.

2.1.4 Pre-requisites for Obtaining the "Outside View" of VMs by directly connecting to the target vSphere/ESX Server

- Ensure that the remote agent has IP connectivity to the vSphere/ESX server.
- Ensure that the remote agent has web access to the web services interface (port 80/443) of the vSphere/ESX server.
- Configure all the tests that the remote agent executes with the name and password of the **root** user to the vSphere/ESX server. However, if, owing to security constraints, you cannot use root user permissions, you can alternatively configure the tests with the credentials of a user who has been assigned the 'Read-only' role. To see how you can create such a user on the ESX server, refer to Section Section **2.5** of this document.

  vSphere/ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop a remote agent, sessions currently in use by the remote agent will remain open for this timeout period until vSphere/ESX times out the session. If the remote agent is restarted within the timeout period, it will open a new set of sessions. If you want the remote agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an vSphere/ESX user with permissions to **View and Terminate Sessions** (from vSphere 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a vSphere/ESX user, refer to Section Section **2.6** of this document.

  Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the reset and refresh SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a user with Change Settings privilege.
- VI APIs are memory-intensive, and therefore, while monitoring ESX servers in an agentless manner, the default memory setting of 128 MB will not be sufficient for the remote agent to pull out the required metrics. The maximum memory setting of the remote agent should hence be increased to any value between 256 MB and 1024 MB, but the exact value will depend upon the number of ESX servers being monitored in your environment. Refer to Section **2.4** to know how to alter the memory settings of the eG agent.

2.1.5 Pre-requisites for Collecting Hardware Metrics

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters of all hardware tests must be configured with the credentials of a user on the vSphere server / vCenter server (depending upon whether monitoring is performed by connecting to the target vSphere server directly or to the vCenter server managing the target vSphere server) with the **Change Settings** privilege. For that you need to create a special role on vSphere/vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new/existing user on vSphere/vCenter. The procedure for this is detailed in Section Section **2.7** of this document.

2.1.6 Pre-requisites for Obtaining the "Inside View" of Windows VMs, using the eG VM Agent

- Install the eG VM Agent on each Windows VM. For details on how to install the eG VM Agent, refer to Section **2.7**.
- Enable the remote agent to communicate with the port at which the eG VM Agent listens (default port: 60001).
- Set the **INSIDE VIEW USING** flag for all the "inside view" tests to **eG VM Agent (Windows)**.

2.1.7 Pre-requisites for Obtaining the "Inside View" of VMs, without using the eG VM Agent

- Ensure that the remote agent has IP connectivity to at least one of the network interfaces of the VMs.
- The **ADMIN$** share should be enabled for all Windows-based virtual guests being monitored and the administrative account must have permissions to this share drive. Refer to Section **2.9.1** for a step-by-step procedure to achieve this.
- To enable the remote agent to communicate with the Windows VMs, an administrative account login and password (either a local account or a domain account) must be provided when configuring the eG monitoring capabilities.
- In case of VMs with the Windows XP/Windows 2003/Windows 2008/Windows Vista/Windows 7 operating systems, the firewall on the guest should be explicitly configured to allow Windows File and Print Sharing services which are required for the remote agent on the vSphere/ESX host to communicate with the guest operating system. Refer to 2.9.2 of this document for a detailed procedure.
- For monitoring a Windows VM, TCP port 139 must be accessible from the remote agent to the VM.
- For monitoring a Linux VM, the SSH port (TCP port 22) must be enabled for communication between the remote agent and the VM being monitored.

Note:

If the Linux VMs in your environment listen on a different SSH port, then, you can override the default SSH port of 22 using the steps provided below:

a. Login to the eG manager.

b. Edit the **eg_tests.ini** file (in the **<EG_INSTALL_DIR>\manager\config** directory) on the eG manager host.

c. In the **[AGENT_SETTINGS]** section of the file, set the **JavaSshPortForVm** parameter to an SSH port of your choice. By default, this parameter is set to *22*.

d. If your environment consists of multiple Linux VMs, each listening on a different SSH port, then, you can specify a comma-separated list of SSH ports against the **JavaSshPortForVm** parameter. For example: *7711,7271,8102*

e. Finally, save the file.

- For obtaining the "inside view" of Windows VMs, the *eGurkhaAgent* service of the eG remote agent (on Windows) should be configured to run using *domain administrator* privileges. Refer to the eG User Manual for the procedure.
- Set the **INSIDE VIEW USING** flag for all the "inside view" tests to **Remote connection to VM (Windows)**.

2.2 Assigning the 'Read-Only' Role to a Local/Domain User to vCenter

In order to ensure that the eG agent uses one/more vCenters in the target environment to auto-discover VMs and to report statistics revealing the relative performance of the VMs (i.e., the outside view), all the tests that the agent executes should be configured with the IP address of the vCenter server to be used; also, the name and password of a user with 'Read-only' privileges to vCenter should be provided.

Given below are the steps to be followed for assigning the 'Read-Only' role to a local/domain user to vCenter:

1. Login to a system on which the **VMware Infrastructure Client** is installed.
2. Double-click on the **VMware Infrastructure Client** icon on your desktop.
3. A login screen then appears (see Figure 2.3). To connect to the vCenter, select the **IP address / Name** of the vCenter, and then provide the login information. To grant access permissions to a user, you will have to login to vCenter as a user with rights to grant permissions to other users. Therefore, provide the **User name** and **Password** of such a user in the login screen.



Figure 2.3: The login screen

4. The **VMware Infrastructure Client** console then opens (see Figure 2.4). This console has a tree-structured listing of datacenters, clusters, resource pools and ESX hosts on the left hand side and a series of tab pages on the right. Click on the top-most node of the tree in the left panel.

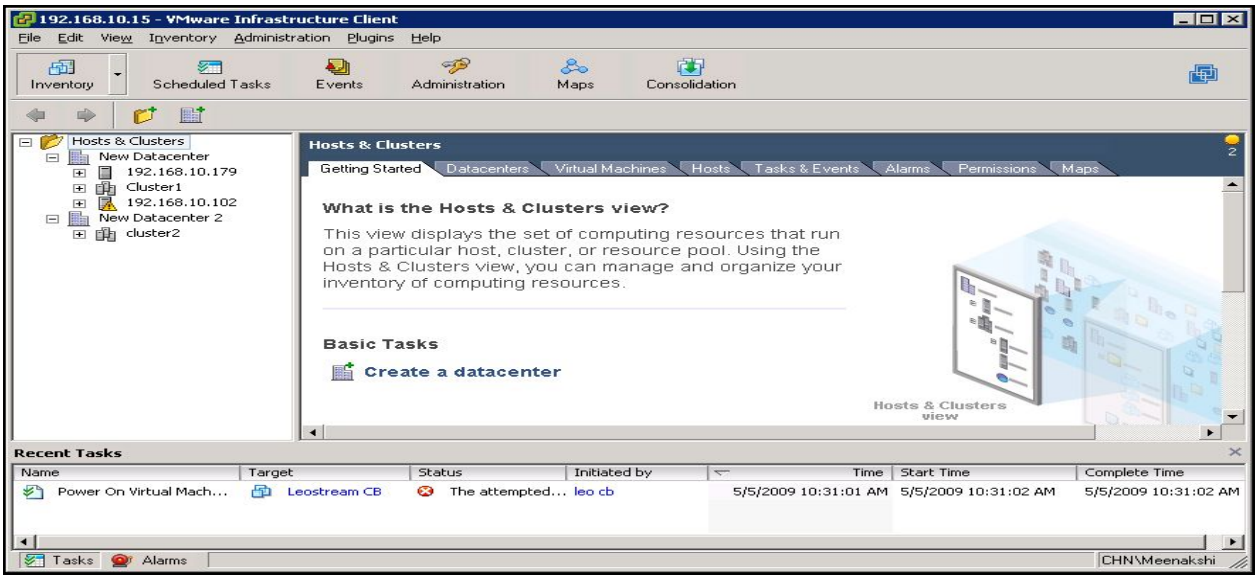


Figure 2.4: The VI Client console

5. To assign 'Read-only' permissions to a vCenter user, first click on the **Permissions** tab page in the right panel (see Figure 2.5). The list of existing users on vCenter and the roles assigned to each of them is then displayed.

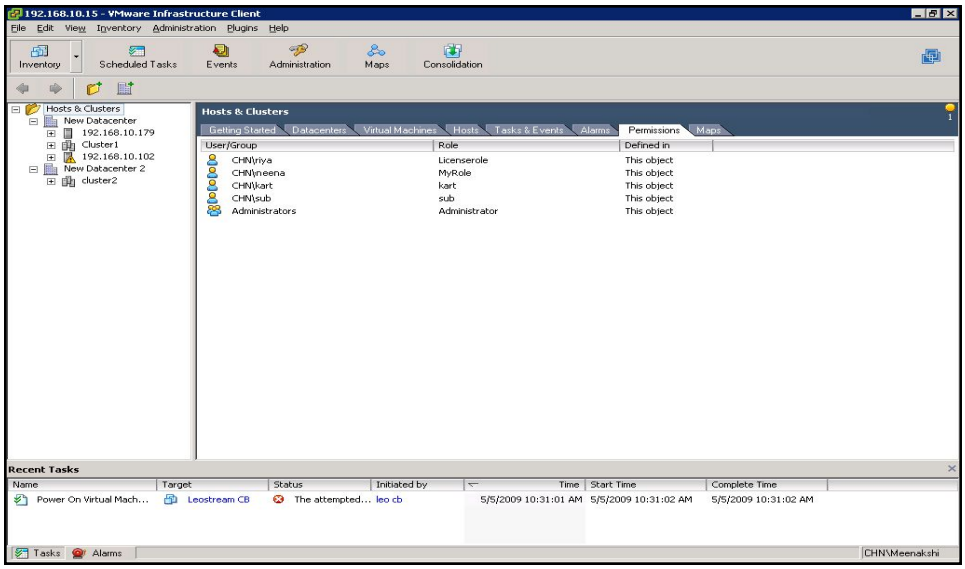


Figure 2.5: List of vCenter users and roles assigned to them

6. Then, right-click anywhere within the tab page, and select the **Add Permission** option (see Figure 2.6).

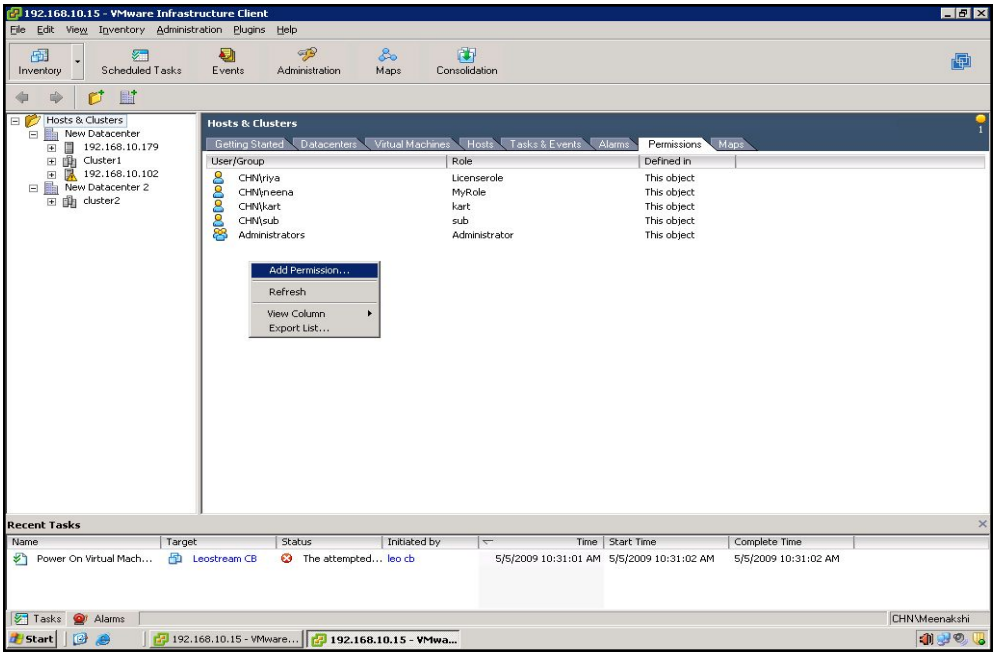


Figure 2.6: Selecting the Add Permission option

7. Figure 2.7 then appears. Click on the **Add** button in Figure 2.7.

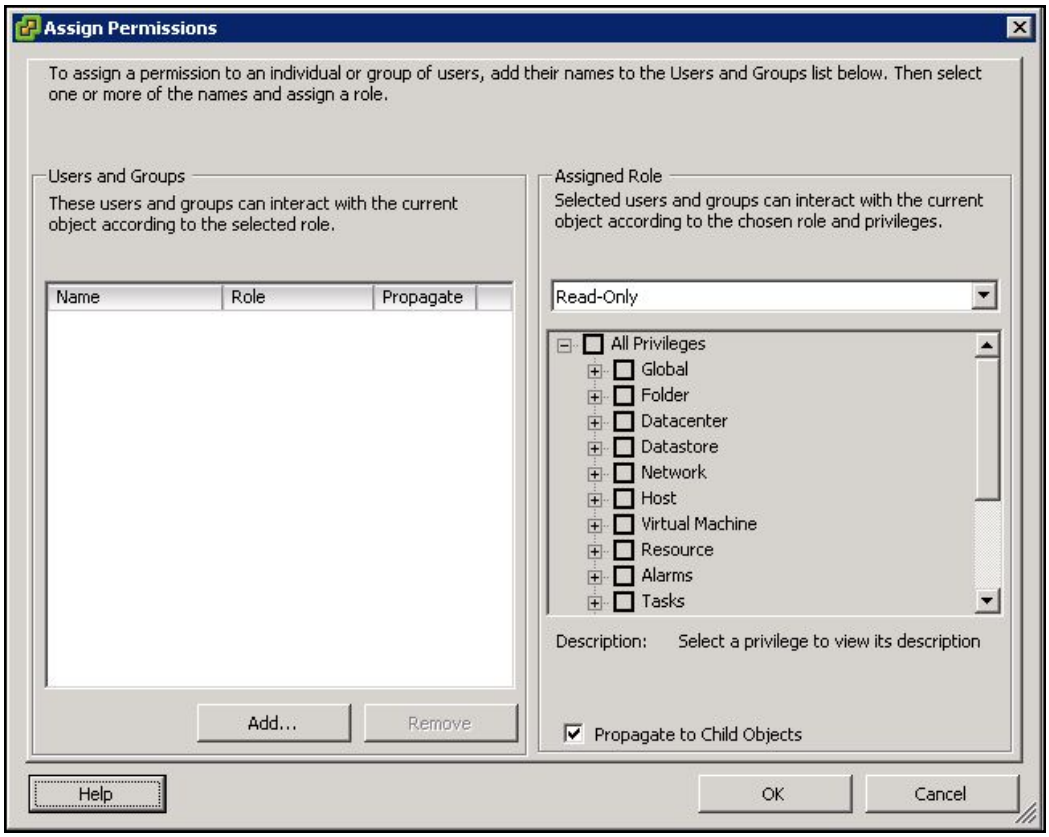


Figure 2.7: Clicking on the Add button

8. Figure 2.8 that then appears allows you to select a local/domain user. If you want to grant a local user the right to View and stop sessions, just select **<servers>** from the **Domain** list. All valid users to the Windows system hosting the vCenter will then be listed in the **Users and Groups** section in Figure 2.8. Select one from this list (see Figure 2.8). To grant the same privilege to a domain user, select the domain from the **Domain** list, and then select a domain user from the **Users and Groups** section (not shown in Figure 2.8).

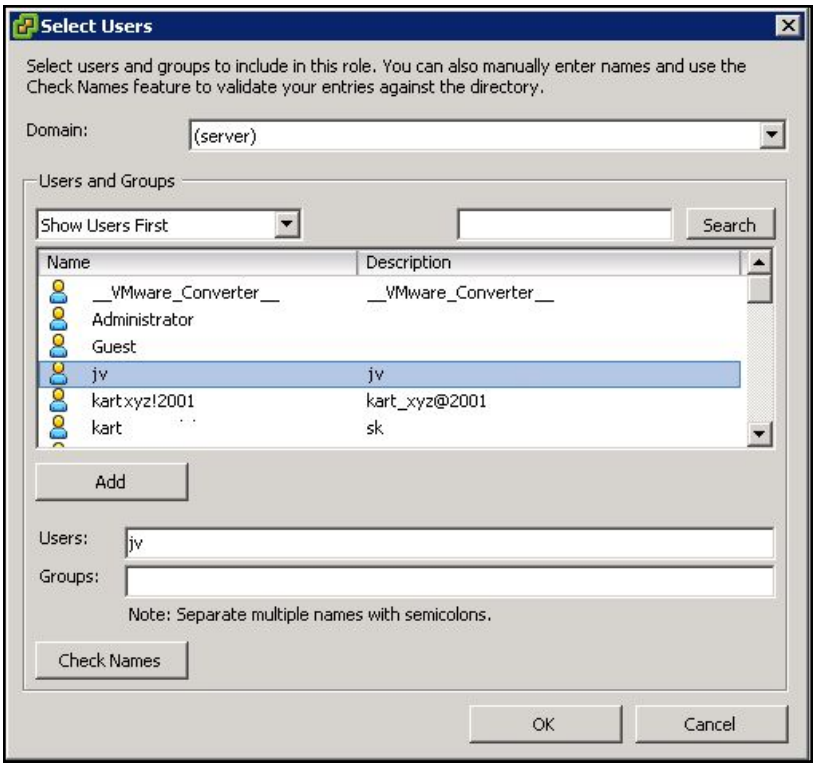


Figure 2.8: Selecting the user for whom a role is to be assigned.

9. Upon double-clicking a user name in the **Users and Groups** list, the selected user name will appear against the **Users** box, as depicted by Figure 2.8.
10. Then, click the **OK** button in Figure 2.8. Figure 2.9 then appears, where you will find the selected user name displayed in the **Users and Groups** section. To assign the read-only role to this user, select the **Read-Only** option from the **Assigned Role** list in Figure 2.9, and click the **OK** button therein.

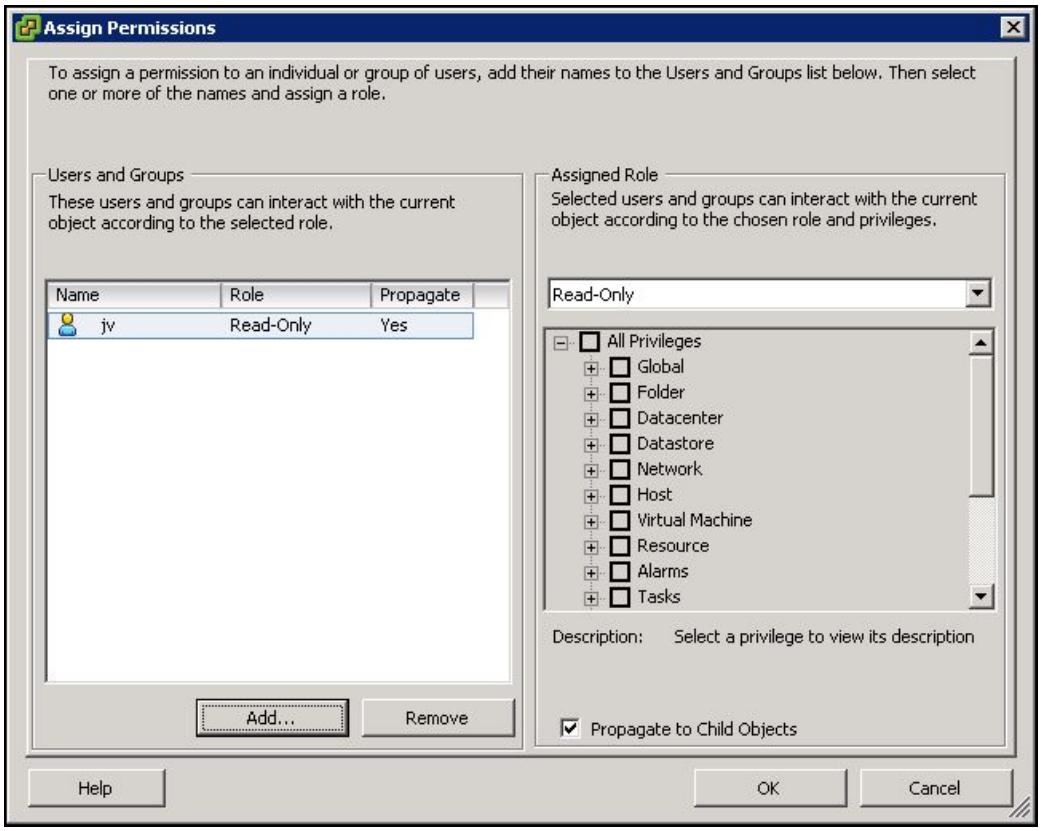


Figure 2.9: Assigning the Read-Only role to the user

11. Figure 2.10 then appears indicating the role assigned to the newly created user.

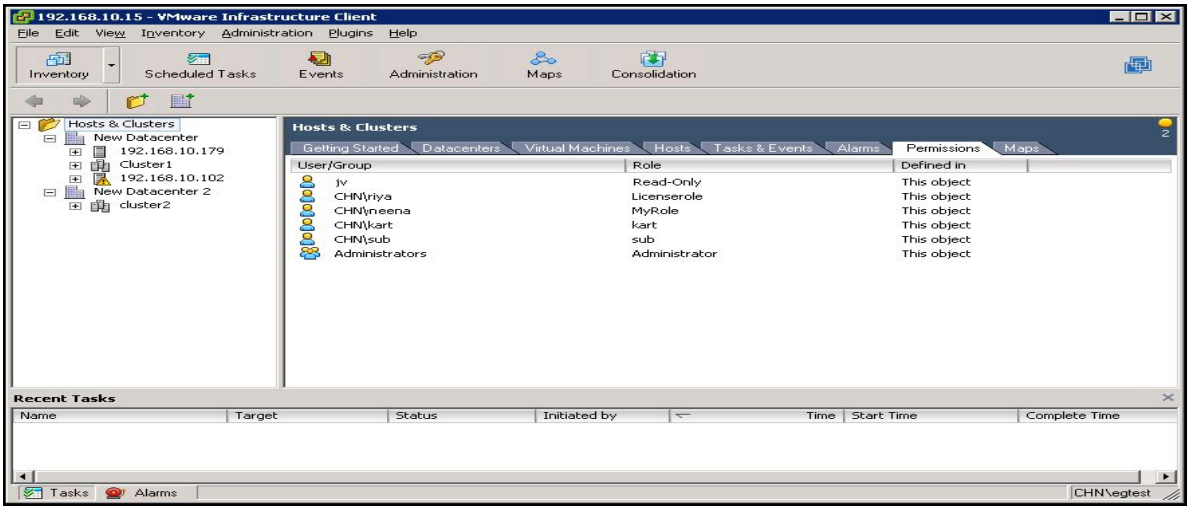


Figure 2.10: The new user and the role assigned to the user being displayed

2.3 Creating a Special Role on vCenter and Assigning the Role to a Local/Domain User

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions on vCenter before it opens new sessions, then, when monitoring ESX servers via vCenter, you would have to configure the tests with the credentials of a user with permissions to **View and stop sessions** on vCenter (prior to vCenter 4.1, this was called the **View and stop sessions** permission).

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Given below are the steps to be followed for creating a separate user account with the **View and stop sessions** privilege:

1. Login to a system on which the **VMware Infrastructure Client** is installed.
2. Double-click on the **VMware Infrastructure Client** icon on your desktop.
3. Figure 2.11 then appears. To connect to the vCenter, select the **IP address / Name** of the vCenter, and then provide the login information. To grant access permissions to a user, you will have to login to vCenter as a user with rights to grant permissions to other users. Therefore, provide the **User name** and **Password** of such a user in Figure 2.11.



Figure 2.11: Connecting to vCenter

4. When the **Virtual Infrastructure Client** opens, click on the **Administration** icon indicated by 2.3.

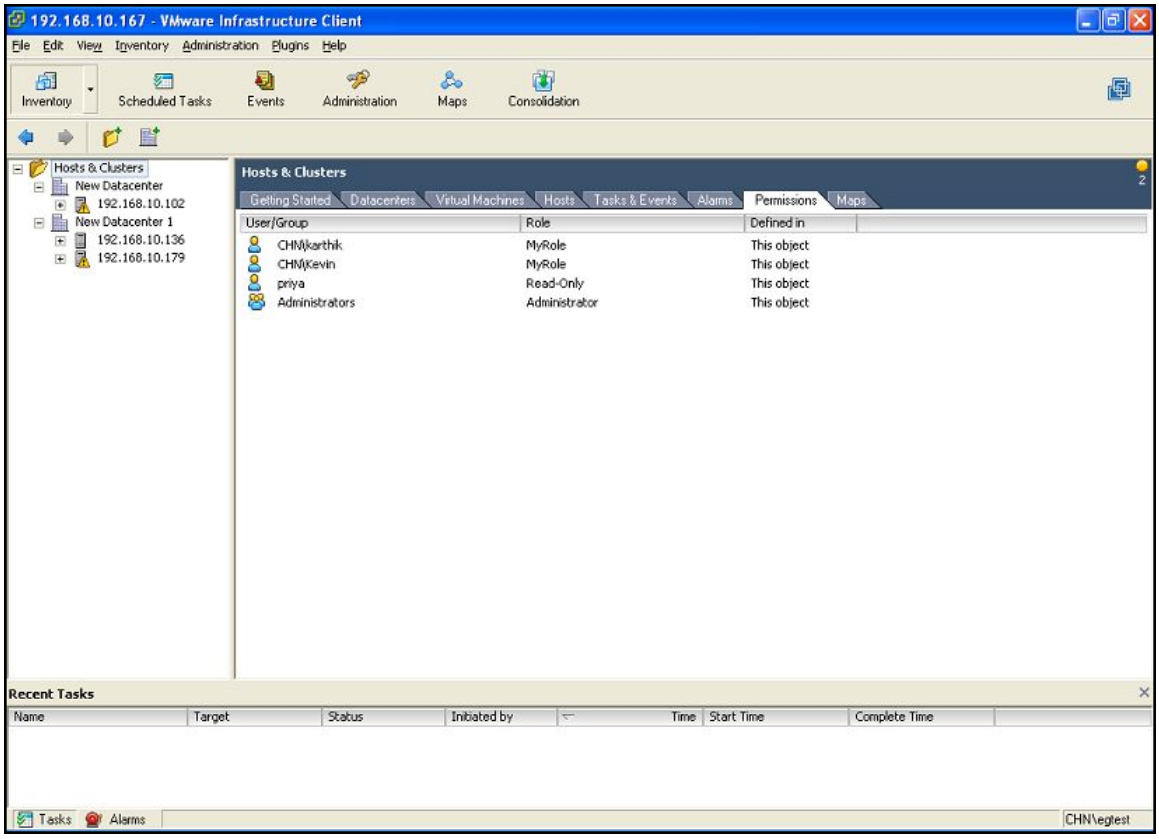


Figure 2.12: Clicking on the Administration icon

5. The **Roles** tab page opens by default, revealing the roles that pre-exist on vCenter (see Figure 2.13).

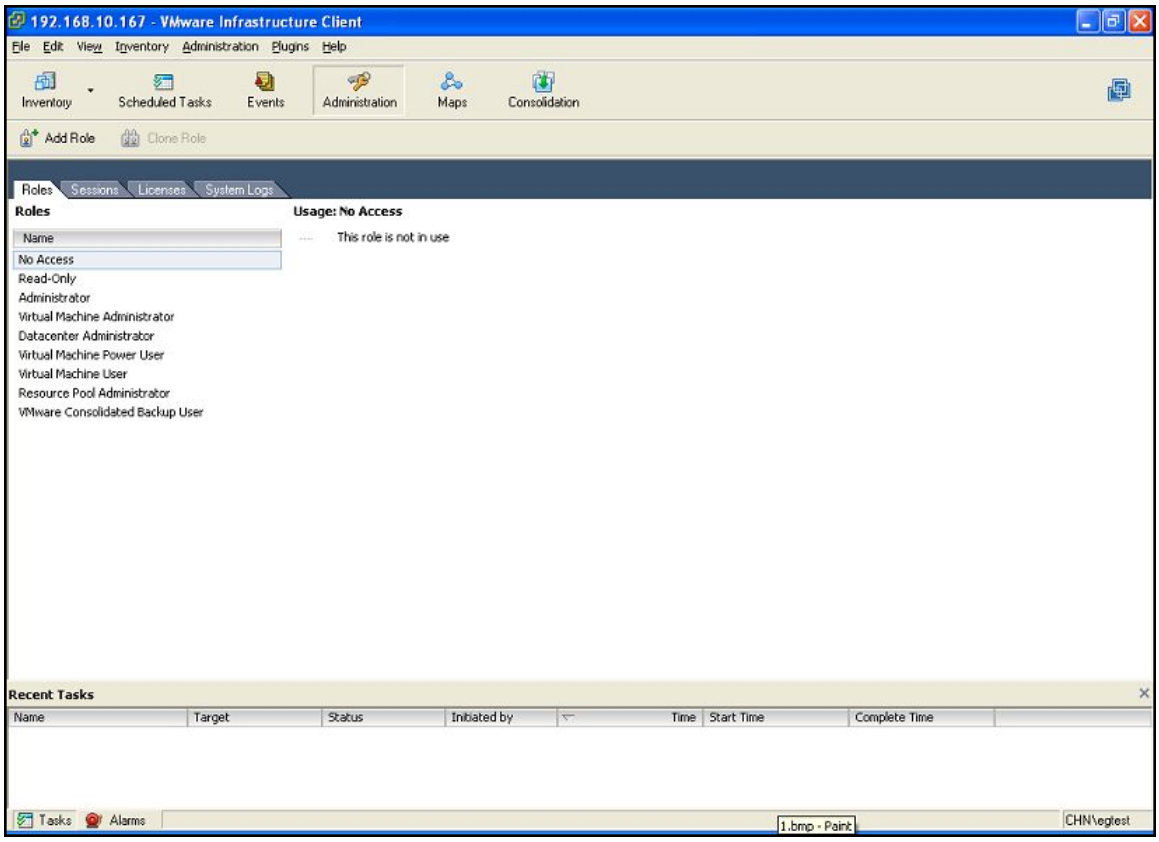


Figure 2.13: Viewing the roles available on the vCenter

6. To create a special role, right-click anywhere within Figure 2.13, and select **Add** from the shortcut menu that appears (see 2.3).

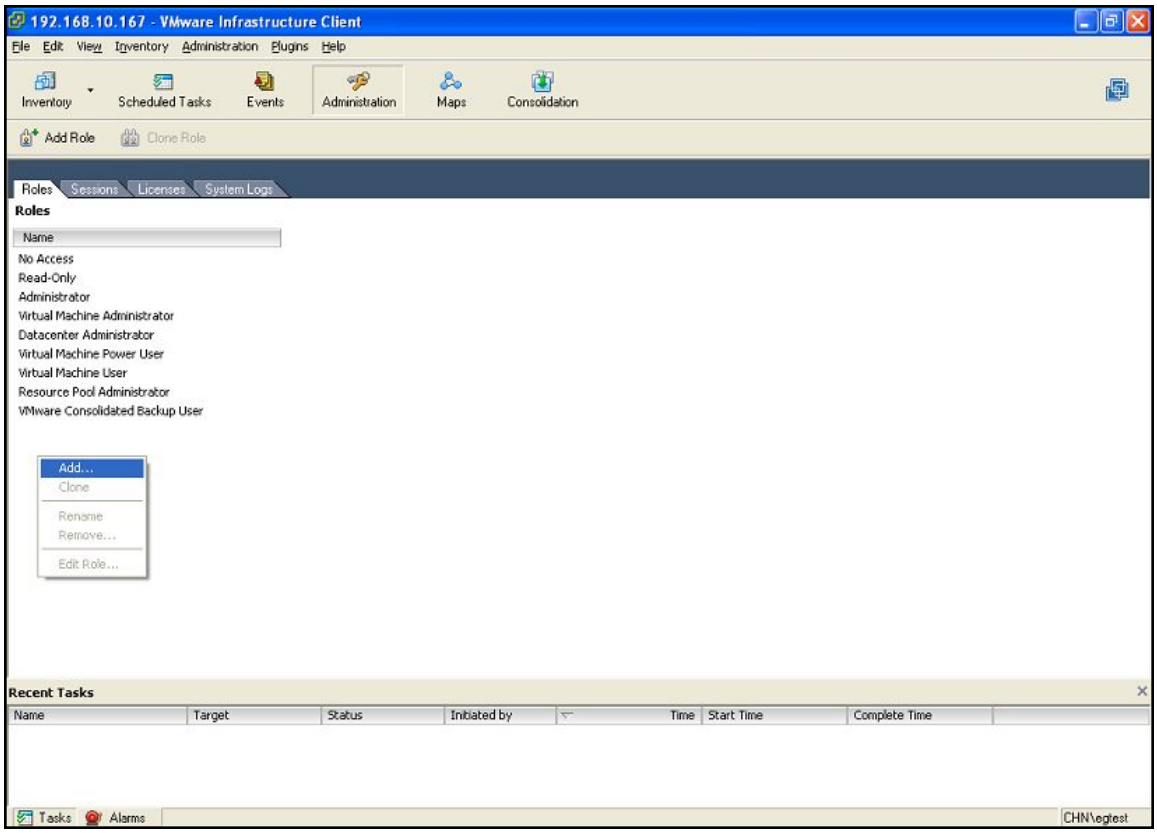


Figure 2.14: Selecting Add from the shortcut menu

7. Figure 2.15 then appears wherein you need to **Enter Name** of the new role. Since the role requires the privilege to view and terminate user sessions on vCenter, expand the **Sessions** node in Figure 2.15, and select the **View and Terminate Sessions** check box under it.

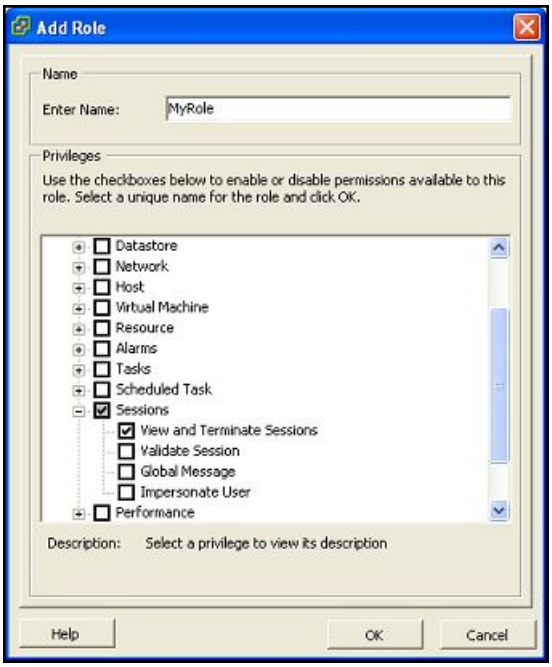


Figure 2.15: Creating a new role on a vCenter server (prior to v 4.1) with the 'View and Terminate Sessions' privilege

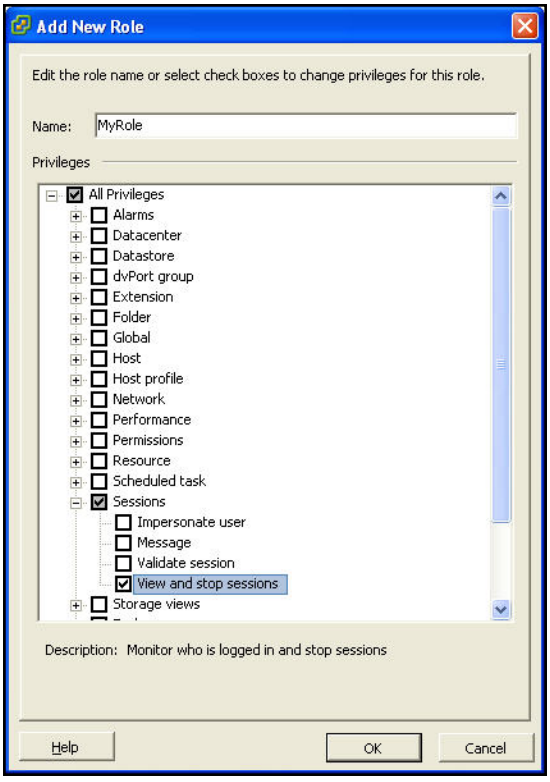


Figure 2.16: Creating a new role on vCenter 4.1 (and above) with the 'View and stop sessions' privilege

8. Then, click the **OK** button in Figure 2.16 to save the changes.

9. Figure 2.17 then appears listing the newly created role. Once the role is created, proceed to assign the role to a local/domain user to vCenter. The first step towards achieving this is to click on the **Inventory** icon indicated by Figure 2.17.

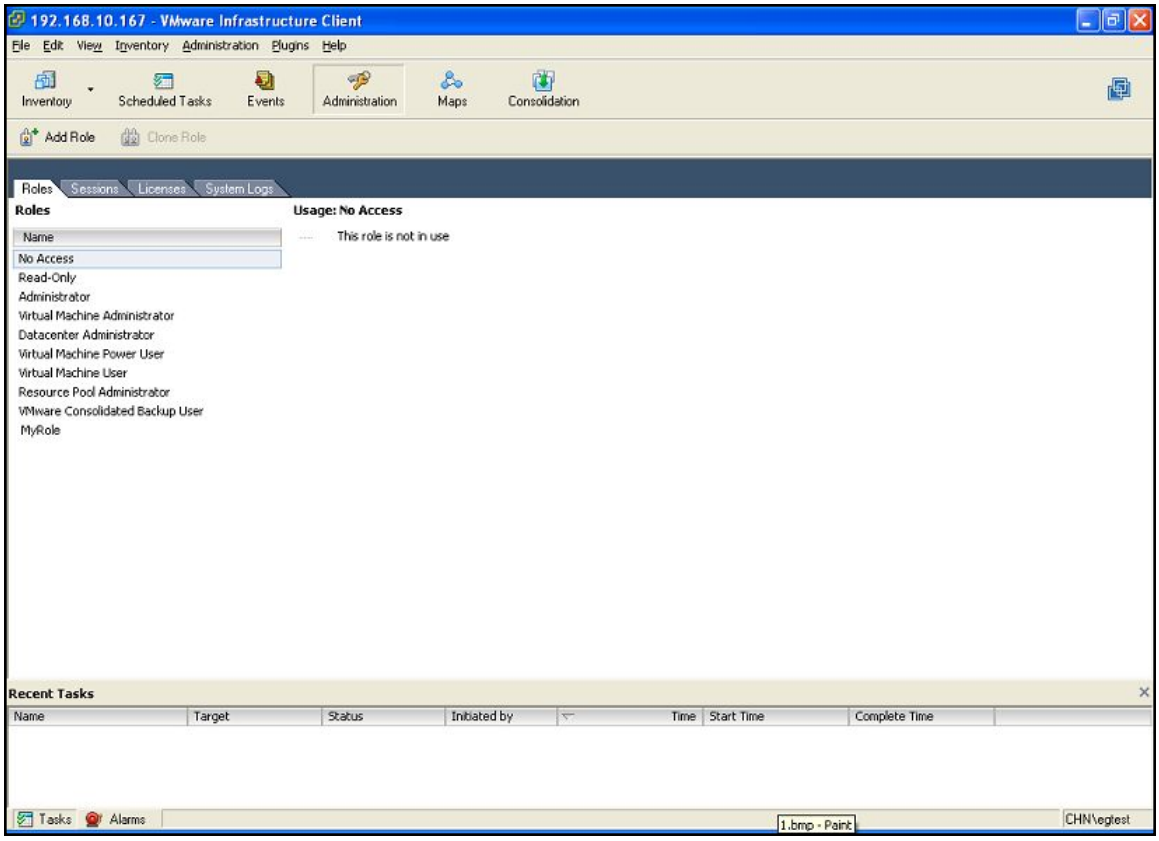


Figure 2.17: The Roles tab page listing the newly created role

10. Figure 2.18 then appears. In the left panel of Figure 2.18, you will find a tree-structure consisting of a wide range of nodes and sub-nodes. Click on the **Hosts & Clusters** node. Clicking on this node reveals a series of tab pages in the right pane. To assign permissions to a user for accessing vCenter, click on the **Permissions** tab in the right pane.

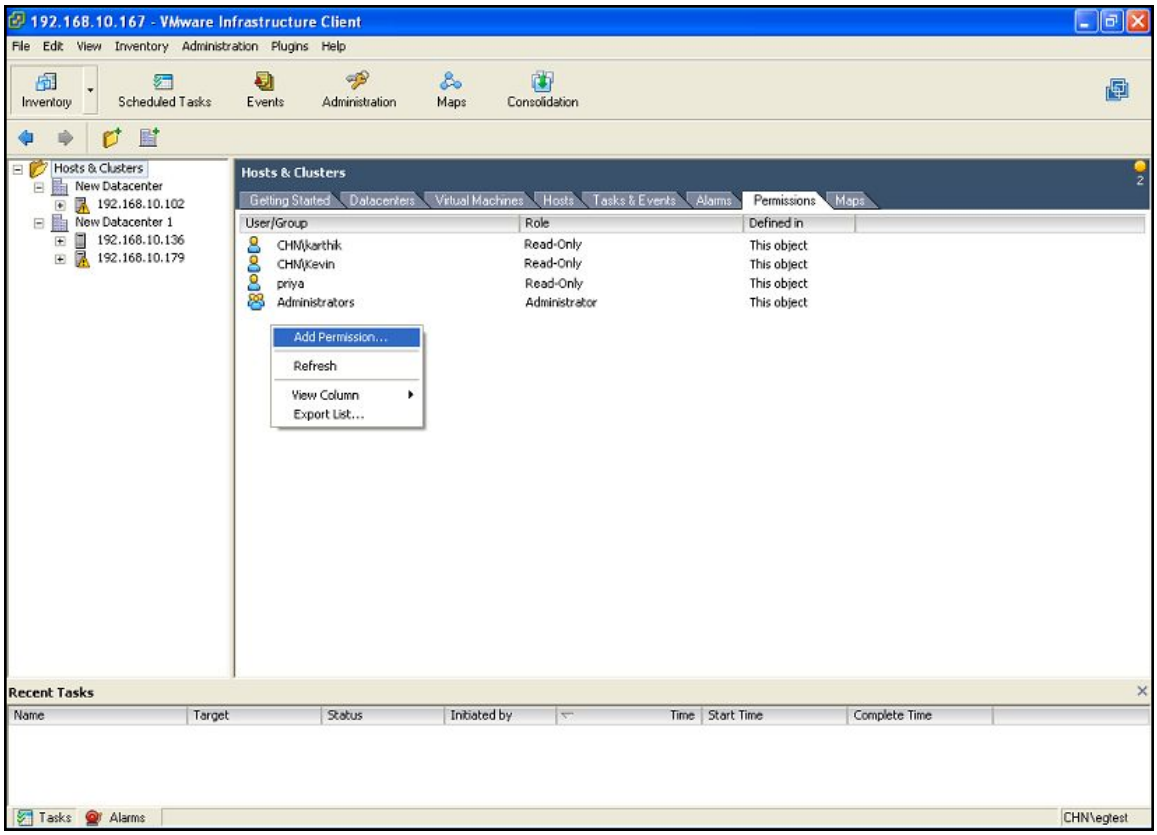


Figure 2.18: Details of users to vCenter

11. When the **Permissions** tab page opens, right-click anywhere within the tab page, and pick the **Add Permission** option from the shortcut menu that appears (see Figure 2.18). Figure 2.19 then opens. From the list box in the **Assigned Role** section of Figure 2.19, select the newly created role.

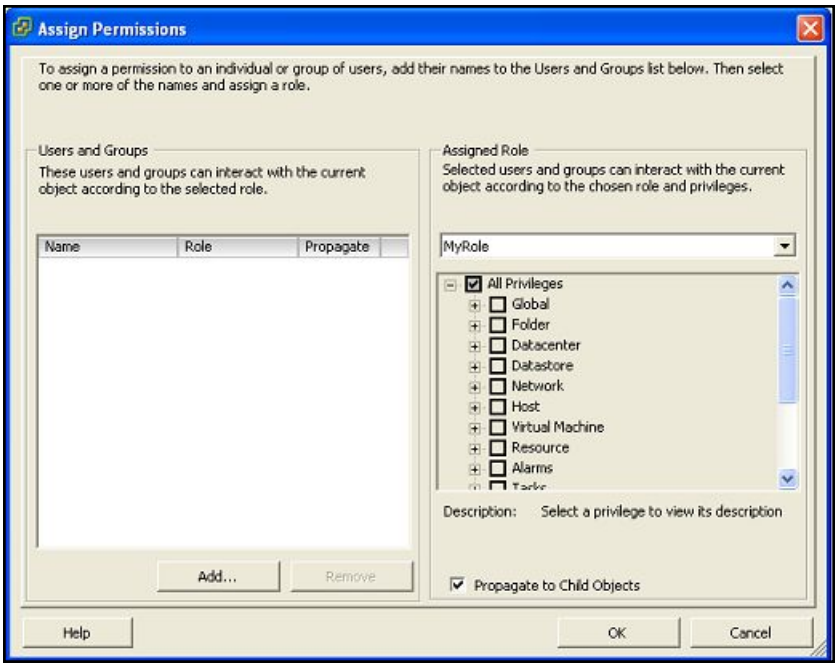


Figure 2.19: Selecting the Add Permission option

12. Since no users have been assigned this role yet, the **Users and Groups** section of Figure 2.19 will appear empty. To map this role to a user, click the **Add** button in Figure 2.19.
13. Figure 2.20 that then appears allows you to select a local/domain user. If you want to grant a local user the right to View and stop sessions, just select **<servers>** from the **Domain** list. All valid users to the Windows system hosting the vCenter will then be listed in the **Users and Groups** section in Figure 2.20. Select one from this list (see Figure 2.20). To grant the same privilege to a domain user, select the domain from the **Domain** list, and then select a domain user from the **Users and Groups** section (not shown in Figure 2.20).

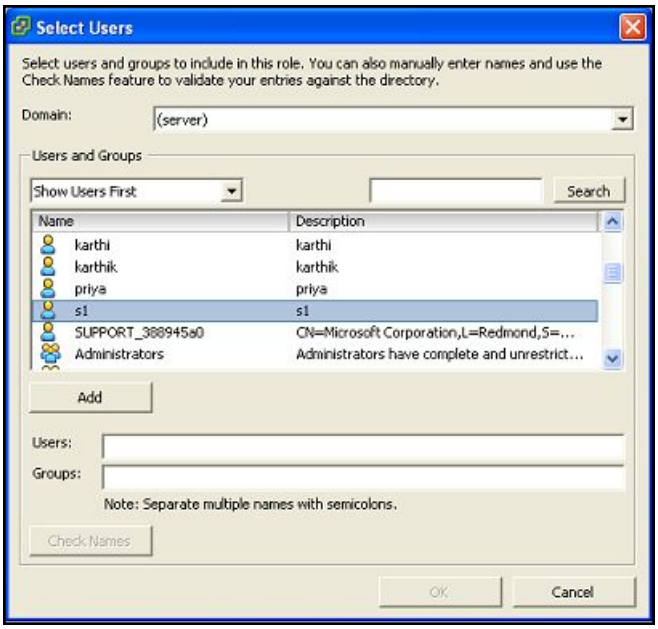


Figure 2.20: Selecting a local user

14. Then, click the **Add** button in Figure 2.20. The chosen user will then be added to the **Users** box as depicted by Figure 2.21.

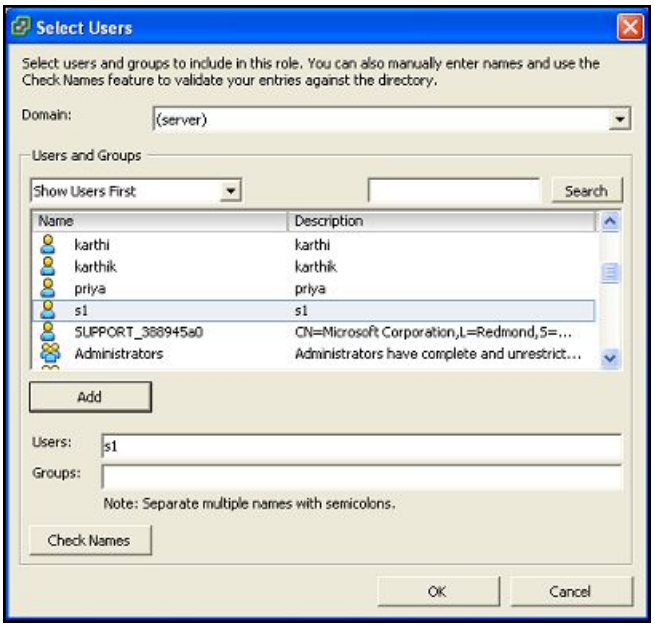


Figure 2.21: Adding the chosen user

15. Next, click the **OK** button in Figure 2.21. 2.3 then appears displaying the local/domain user that you added previously, in the **Users and Groups** section.

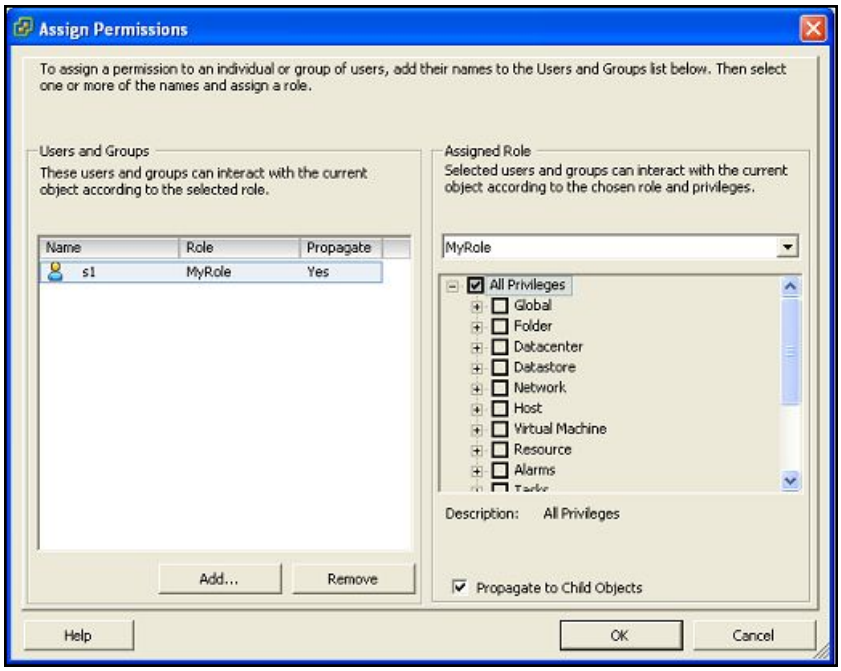


Figure 2.22: Assigning the Read-Only role to a chosen user

16. Finally, click the **OK** button.

2.4 Increasing the Memory Settings of the eG Agent that Monitors ESX Servers

The AXIS library used by the VI APIs is memory-intensive, and therefore, while monitoring ESX servers using the VI APIs, the default memory setting of 128 MB will not be sufficient for the eG agent to pull out the required metrics. The maximum memory setting of the eG agent should hence be increased to any value between 256 MB and 1024 MB, but the exact value will depend upon the number of ESX servers being managed in your environment.

Note:

If the eG agent is being installed on the same box as the eG manager, then, by default, the eG agent setup process will automatically increase the memory setting of the agent. If the eG agent is being installed alongside a Unix manager, the agent memory will be automatically reset to 256 MB by default. On a Windows manager host, this default value is dynamically determined, but will not be higher than 512 MB.

The steps to be followed to alter the agent's memory settings vary according to the operating system (OS) on which the agent is executing. The sections to come discuss the procedures specific to each OS.

2.4.1 Increasing the Memory Settings of the eG Agent on Windows

To increase the memory settings of the eG agent on Windows, do the following:

1. Stop the eG agent.
2. Edit the **debugon.bat** file in the **<EG_AGENT_INSTALL_DIR>\lib** directory.

3. In the file, search for the **-Xrs** entry.
4. Next to **-Xrs**, insert an entry of the following format: **-Xmx<memory>m**. For instance, to increase the maximum memory setting to 256 MB, the entry to be inserted should be: **-Xmx256m** (see Figure 2.23).

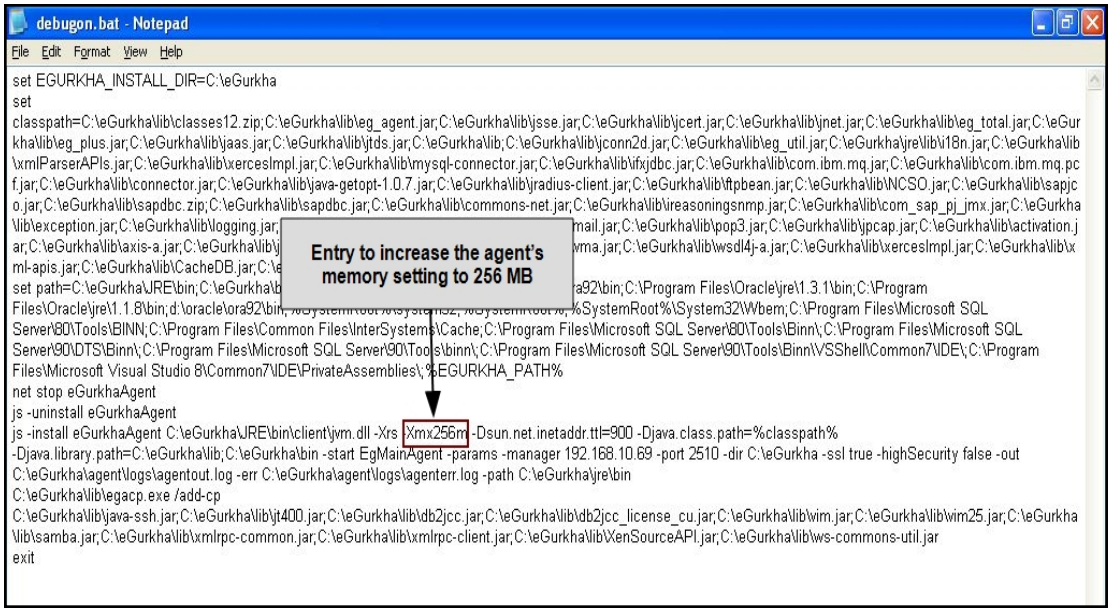


Figure 2.23: Increasing the memory setting of an eG agent on Windows by editing the debugon.bat file

5. Then, save the file.
6. Next, open the **debugoff.bat** file (in the <EG_AGENT_INSTALL_DIR>\lib directory), and repeat steps 3 and 4 above (see Figure 2.24).

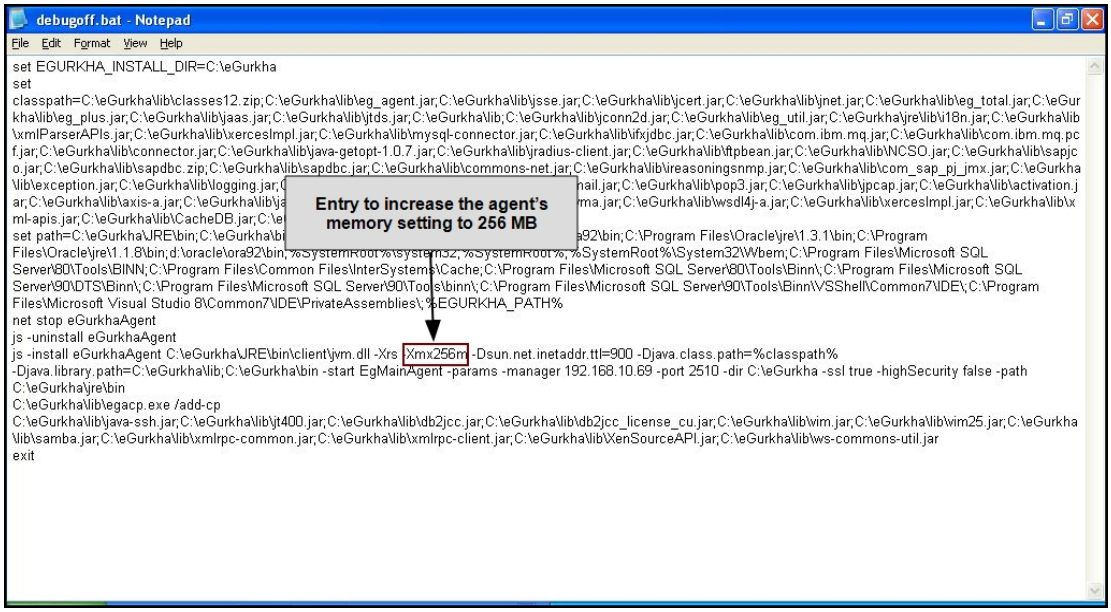


Figure 2.24: Increasing the memory settings of an eG agent on Windows by editing the debugoff.bat file

7. Finally, save the file, execute **debugoff.bat**, and restart the eG agent.

2.4.2 Increasing the Memory Settings of an eG Agent on Solaris

To increase the memory settings of the eG agent on Solaris, do the following:

1. Stop the eG agent.
2. Edit the **starta** file in the **/opt/egurkha/bin** directory.
3. In the file, search for the **-Xrs** entry.
4. Next to **-Xrs**, insert an entry of the following format: **-Xmx<memory>m**. For instance, to increase the eG agent's RAM size to 256 MB, the entry to be inserted should be: **-Xmx256m** (see Figure 2.25).

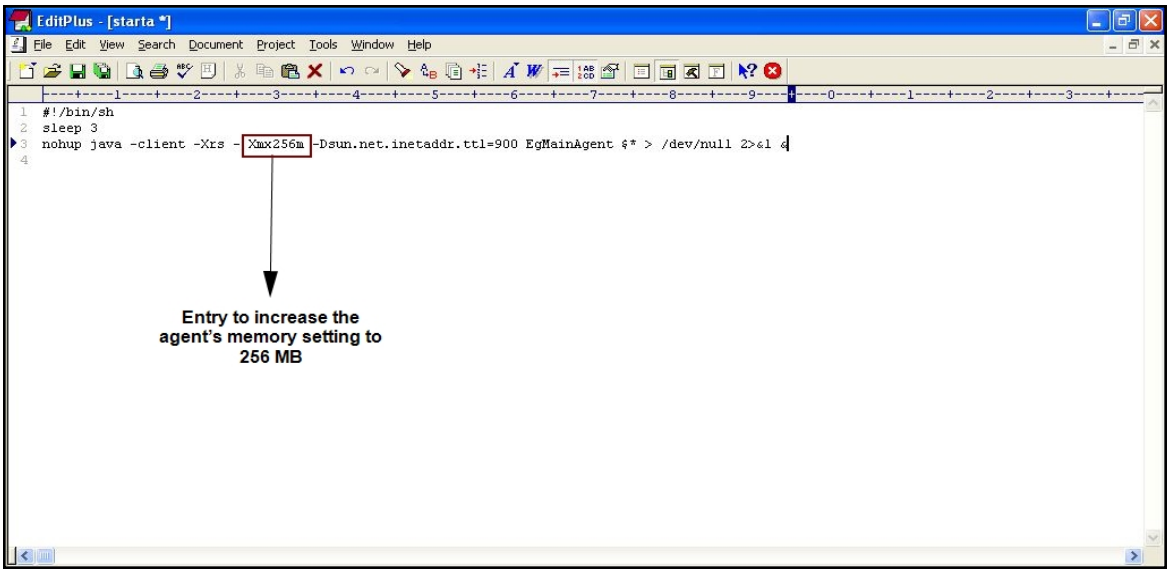


Figure 2.25: Increasing the memory setting of an eG agent on Solaris by editing the starta file

5. Then, save the file.
6. Finally, restart the eG agent.

2.4.3 Increasing the Memory Settings of an eG Agent on Linux

To increase the memory settings of the eG agent on Linux, do the following:

1. Stop the eG agent.
2. Edit the **start_agent** file in the **/opt/egurkha/bin** directory.
3. In the file, search for the **-Xrs** entry.
4. Next to **-Xrs**, insert an entry of the following format: **-Xmx<memory>m**. For instance, to increase the eG

agent's RAM size to 256 MB, the entry to be inserted should be: **-Xmx256m** (see Figure 2.26).

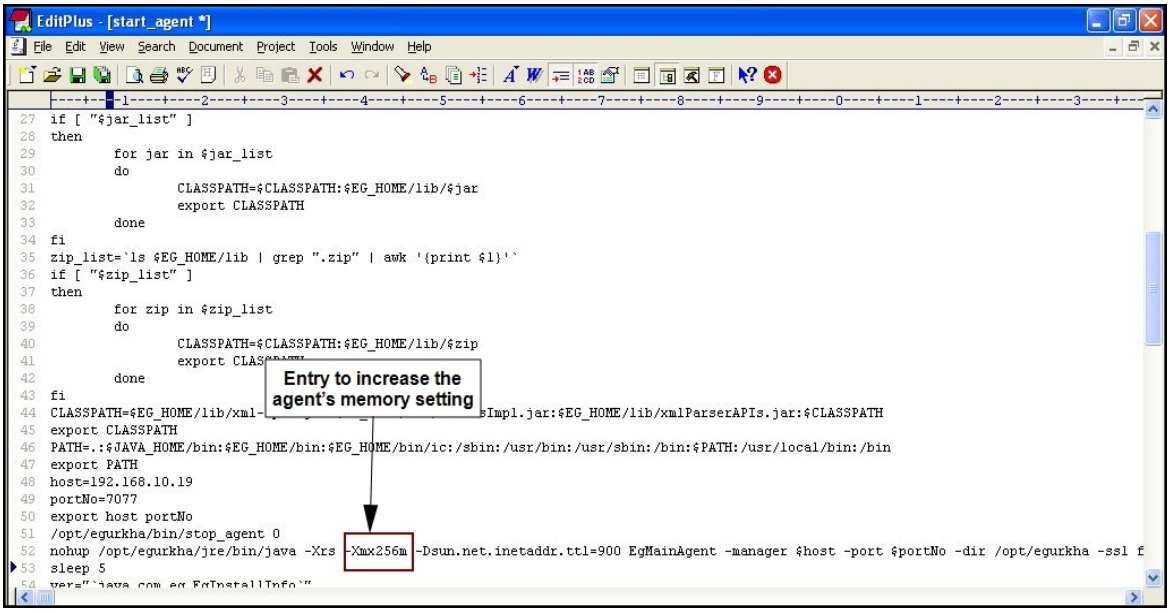


Figure 2.26: Increasing the memory setting of an eG agent on Linux by editing the start_agent file

5. Finally, save the file, and restart the eG agent.

2.5 Creating a New User with Read-Only Privileges to the ESX Server

If the eG agent is to use VI SDK for collecting statistics or just to auto discover the VMs, then at the time of test configuration, you will have to provide the name and password of an ESX user who has been assigned the 'Read-Only' role. By default, the **root** user of the ESX server is authorized to execute the tests. However, it is preferable that you create a new user on the monitored ESX server with the said privileges, using the steps detailed below:

1. Login to a system on which the **VMware Infrastructure Client** is installed.
2. Double-click on the **VMware Infrastructure Client** icon on your desktop.
3. Figure 2.27 then appears. To connect to the monitored ESX host, select the **IP address / Name** of the ESX host from the list box, and then provide the login information. To create a new user on an ESX host, you will have to login to the host as a **root** user. Therefore, provide the **User name** and **Password** of the **root** user in Figure 2.27.



Figure 2.27: Connecting to an ESX host

4. The **VMware Infrastructure Client** console then opens (see Figure 2.28). This console has a tree-structured listing of VMs and resource pools on the left hand side and a series of tab pages on the right. The topmost node of the tree-structure represents the ESX host, which has the virtual machines (or guests) as its sub-nodes. To create a new user on the ESX host, first click on the **Users & Groups** tab page in the right panel (see Figure 2.28). The list of existing users on the ESX host is then displayed.

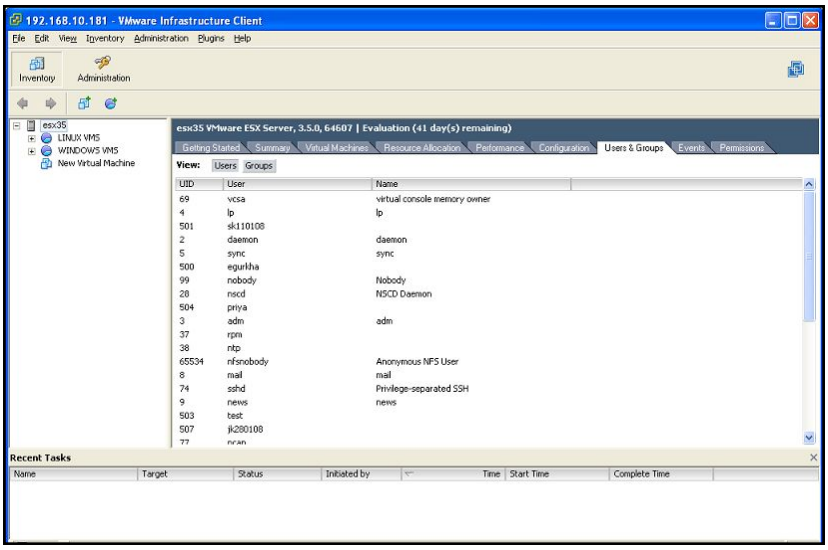


Figure 2.28: The VI Client console

5. Next, right-click any where within the **Users & Groups** tab page to invoke a shortcut menu. Select the **Add** option from the menu to add a new user (see Figure 2.29).

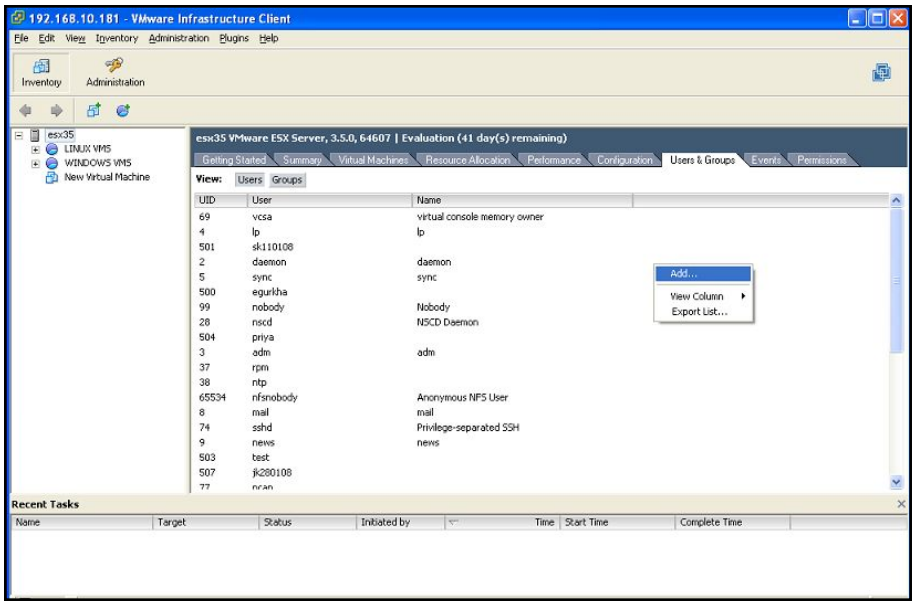


Figure 2.29: Selecting the Add option

6. Doing so opens Figure 2.30, wherein you need to provide the **Login** name of the new user, the **User Name**, and the **Password**. Confirm the password by retyping it in the **Confirm** text box, and click the **OK** button to complete the addition of the new user.

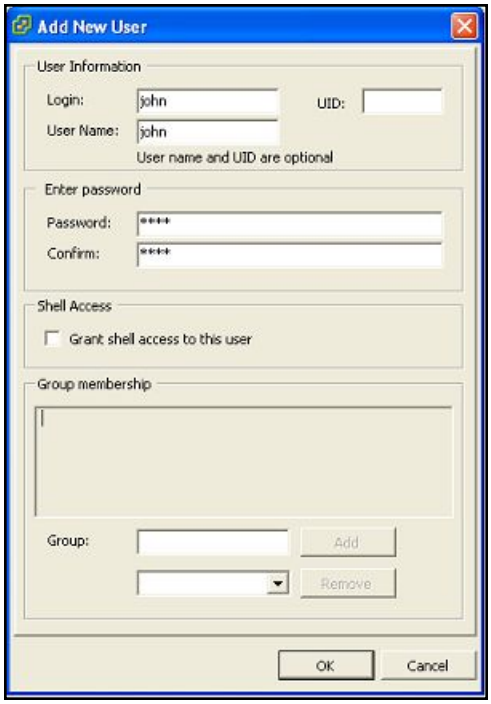


Figure 2.30: Adding a new user

7. Figure 2.31 then appears, where you will find the newly added user name displayed.

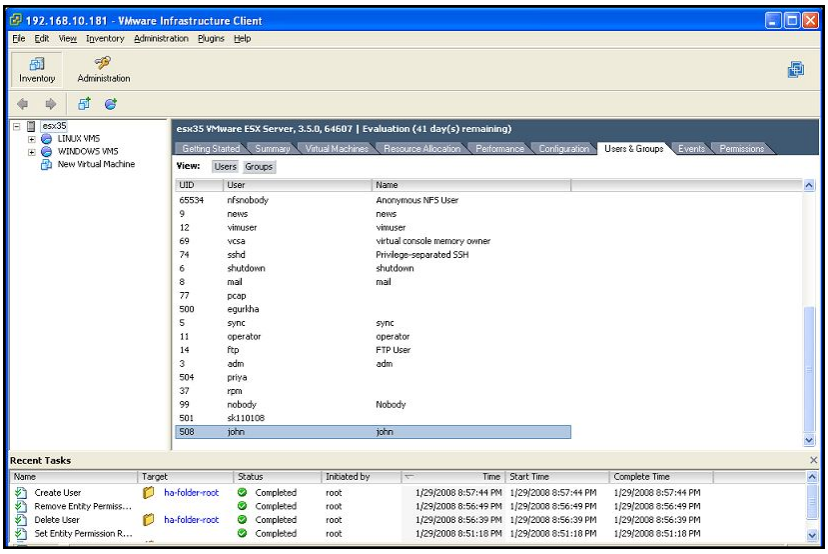


Figure 2.31: The newly added user name displayed in the list of users to the ESX host

8. Next, to assign a role to the new user, click on the **Permissions** tab page in Figure 2.32, right-click anywhere within the tab page, and select the **Add Permission** option (see Figure 2.32).

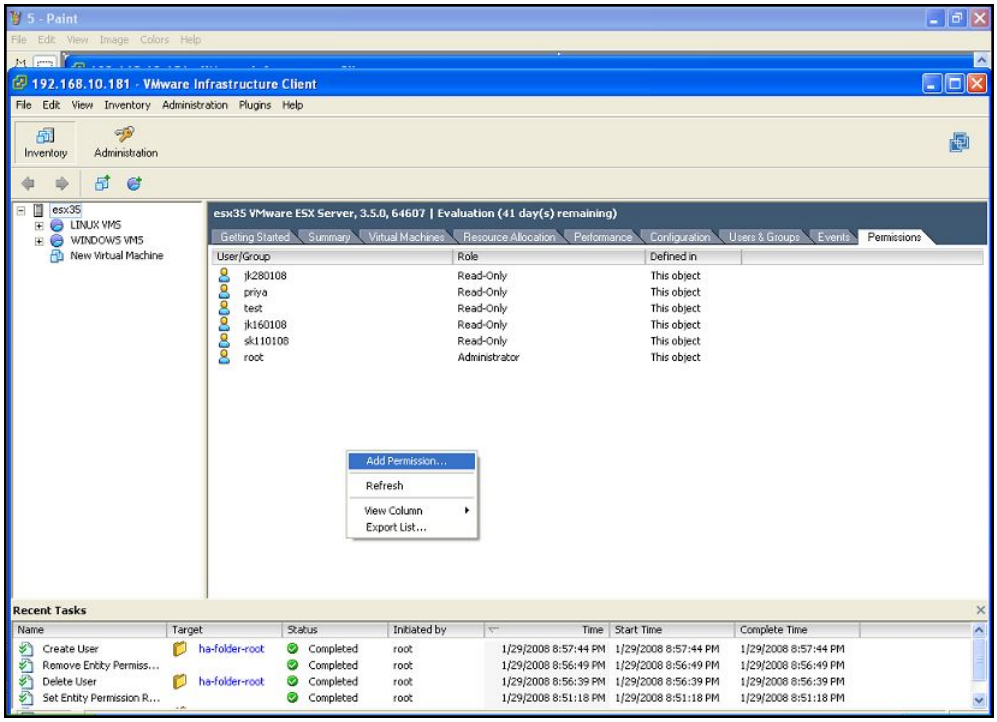


Figure 2.32: Figure 2.32: Selecting the Add Permission option

9. Figure 2.33 then appears. Click on the **Add** button in Figure 2.33.

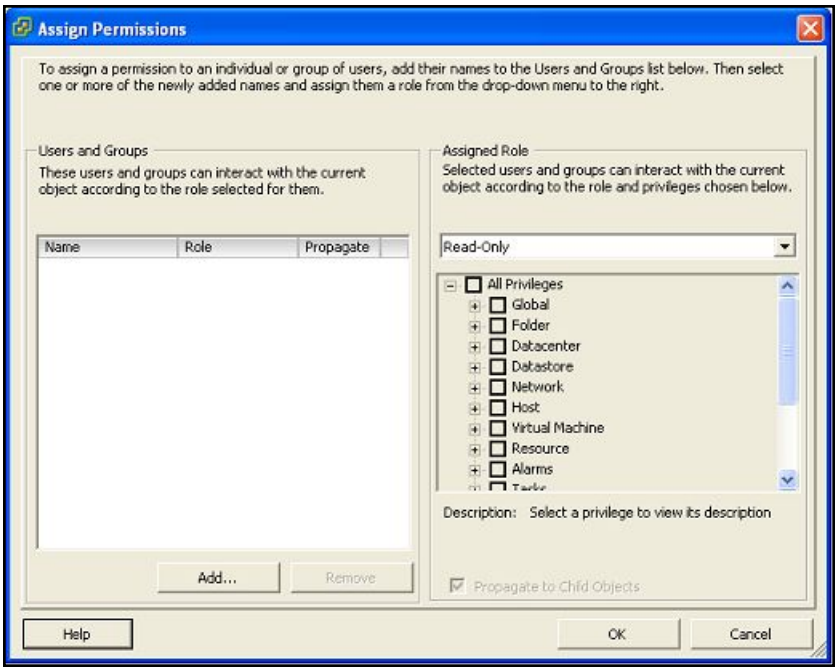


Figure 2.33: Clicking on the Add button

10. From the **Users and Groups** list that then appears, select the user that you just created, and click the **Add** button therein (see Figure 2.34).

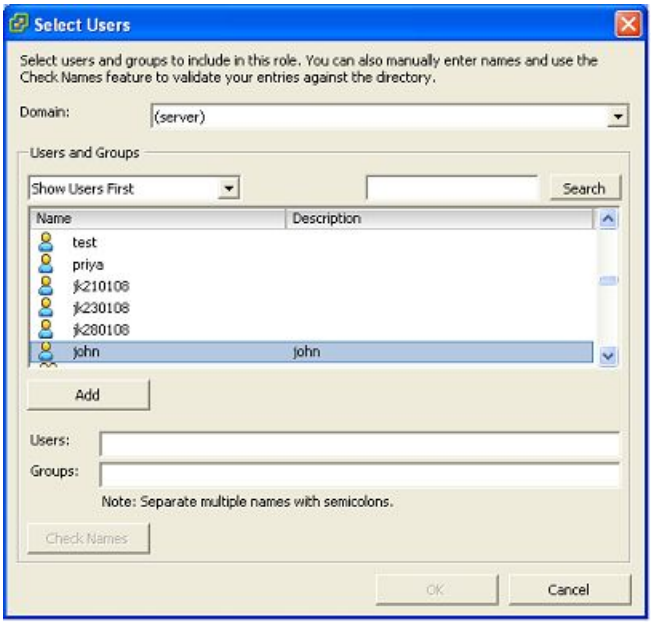


Figure 2.34: Selecting the user for whom a role is to be assigned.

11. Upon clicking, the selected user's name will appear against the **Users** box, as depicted by Figure 2.35.

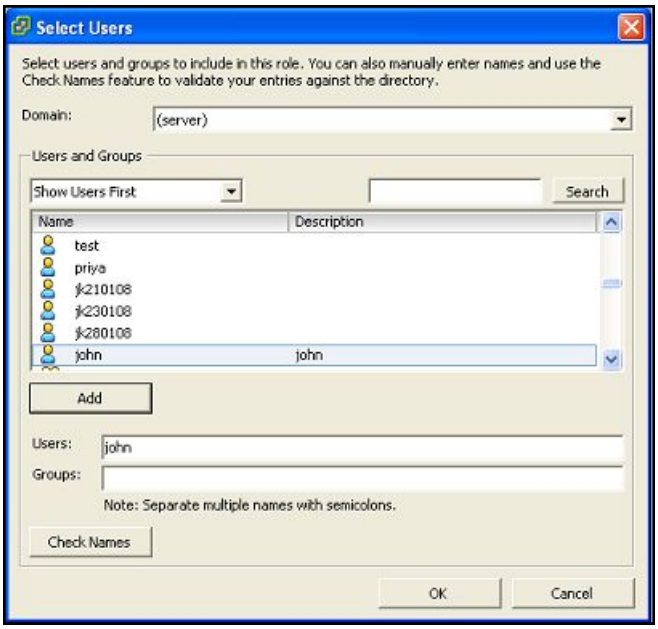


Figure 2.35: The user name displayed in the Users box

12. Then, click the **OK** button in Figure 2.35. Figure 2.36 then appears, where you will find the selected user name displayed in the **Users and Groups** section. To assign the read-only role to this user, select the **Read-Only** option from the **Assigned Role** list in Figure 2.36, and click the **OK** button therein.

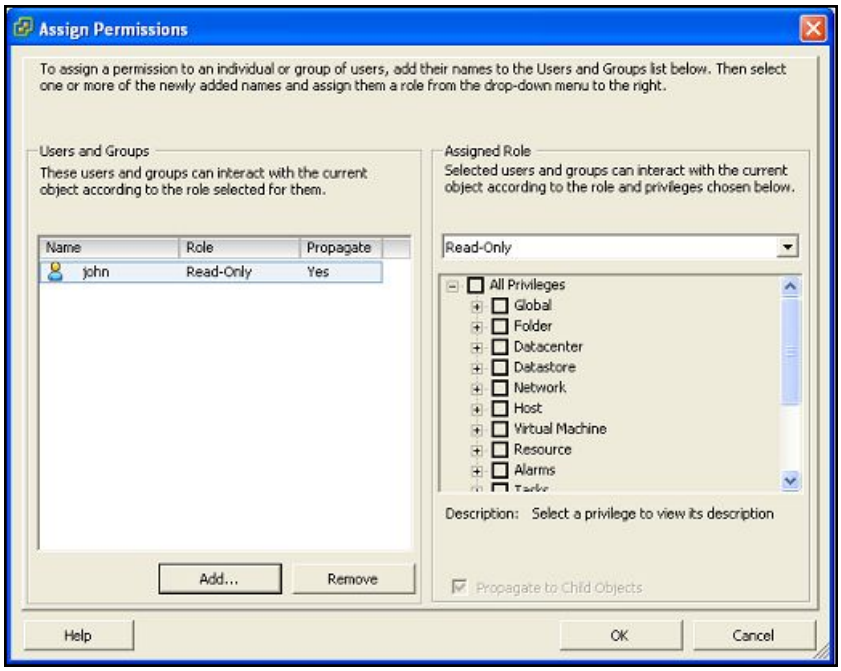


Figure 2.36: Assigning the Read-Only role to the user

13. Figure 2.37 then appears indicating the role assigned to the newly created user.

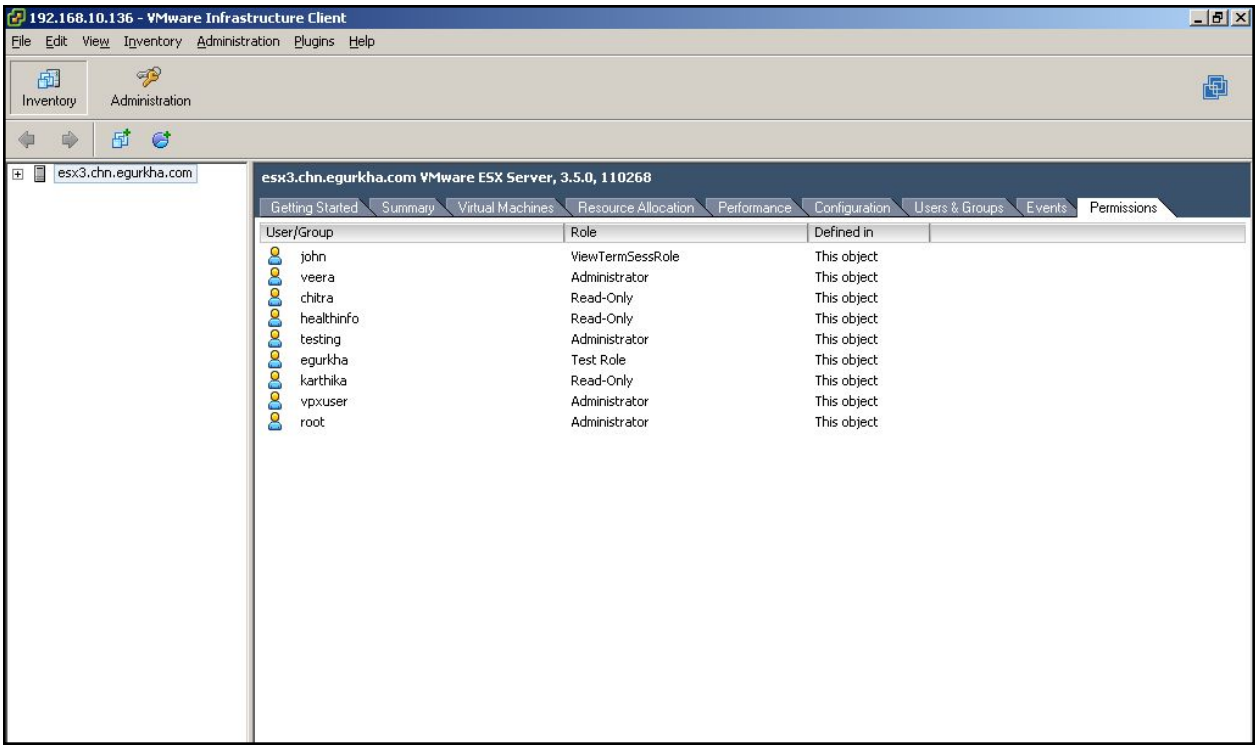


Figure 2.37: The new user and the role assigned to the user being displayed

2.6 Creating a Special Role on an ESX Server and Assigning the Role to a New User to the Server

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions** (from vSphere 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a new ESX user, do the following:

1. Login to a system on which the **VMware Infrastructure Client** is installed.
2. Double-click on the **VMware Infrastructure Client** icon on your desktop.
3. A login screen then appears. To connect to the monitored ESX host, select the **IP address / Name** of the ESX host from the list box, and then provide the login information. To create a new user on an ESX host, you will have to login to the host as a **root** user. Therefore, provide the **User name** and **Password** of the **root** user in the login screen.
4. The **VMware Infrastructure Client** console then opens (see 2.6). This console has a tree-structured listing of VMs and resource pools on the left hand side and a series of tab pages on the right. The topmost node of the tree-structure represents the ESX host, which has the virtual machines (or guests) as its sub-nodes. To create a new user on the ESX host, first click on the **Users & Groups** tab page in the right

panel (see 2.6). The list of existing users on the ESX host is then displayed.

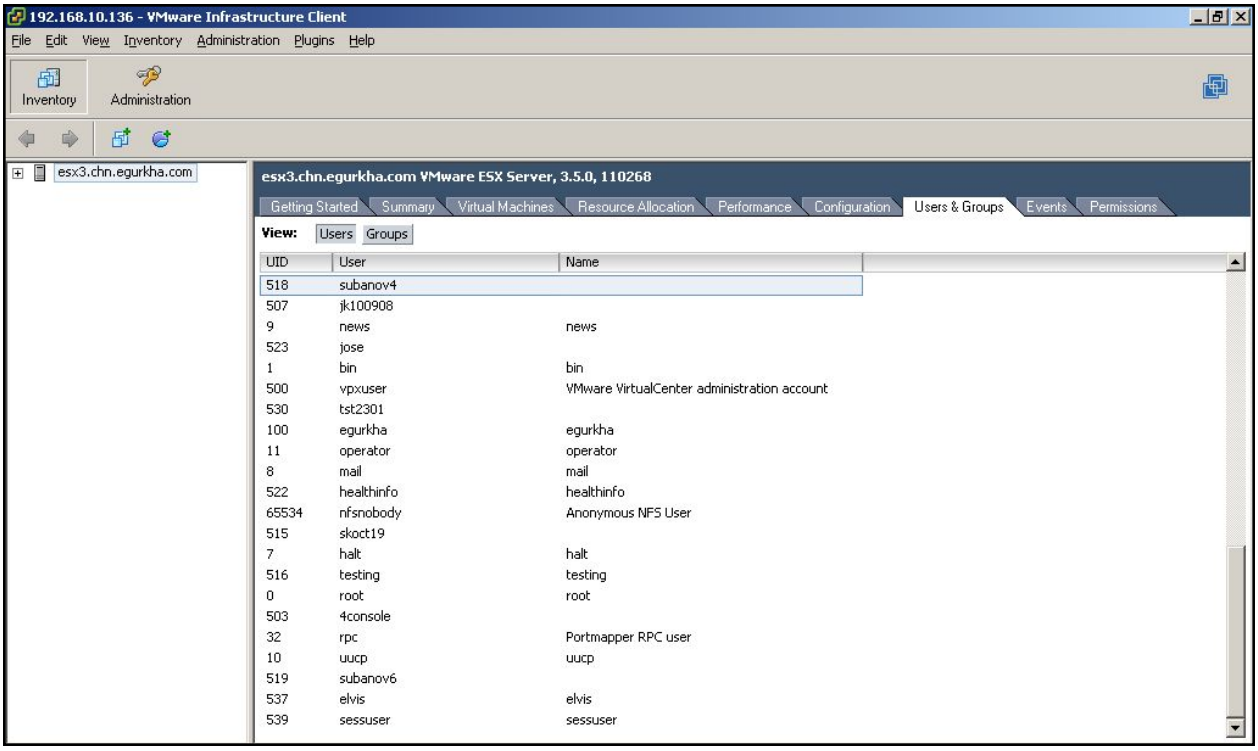


Figure 2.38: The VI Client console - Users & Groups tab page

5. Next, right-click any where within the **Users & Groups** tab page to invoke a shortcut menu. Select the **Add** option from the menu to add a new user (see Figure 2.39).

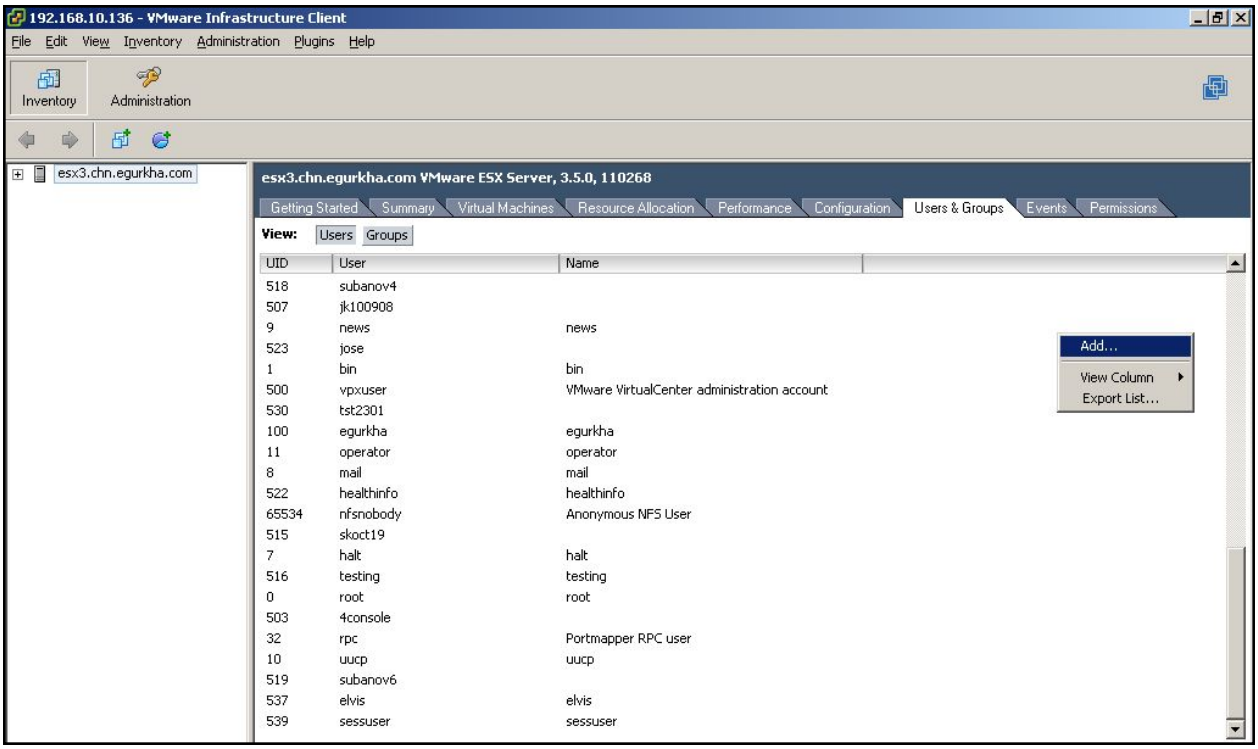


Figure 2.39: Selecting the Add option

6. Doing so opens Figure 2.40, wherein you need to provide the **Login** name of the new user, the **User Name**, and the **Password**. Confirm the password by retyping it in the **Confirm** text box, and click the **OK** button to complete the addition of the new user.

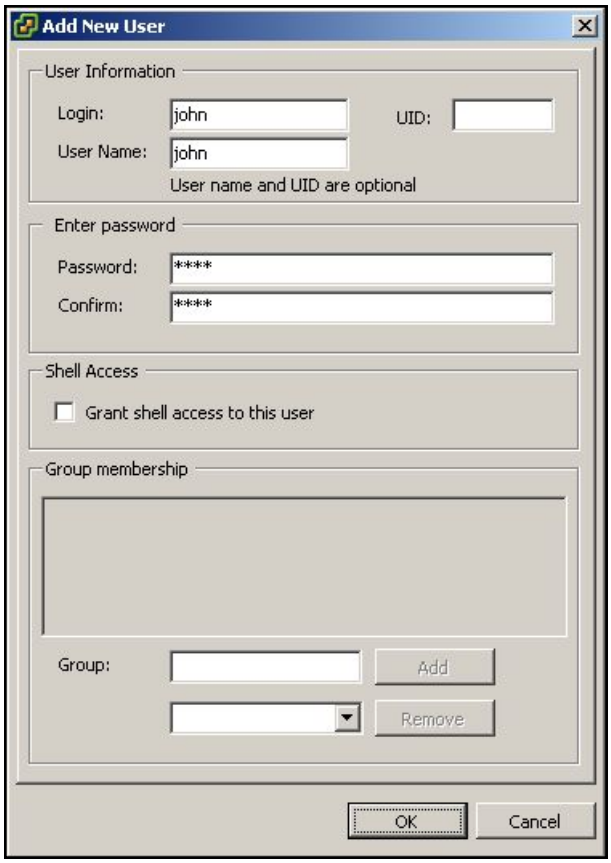


Figure 2.40: Adding a new user

7. 2.6 then appears, where you will find the newly added user name displayed.

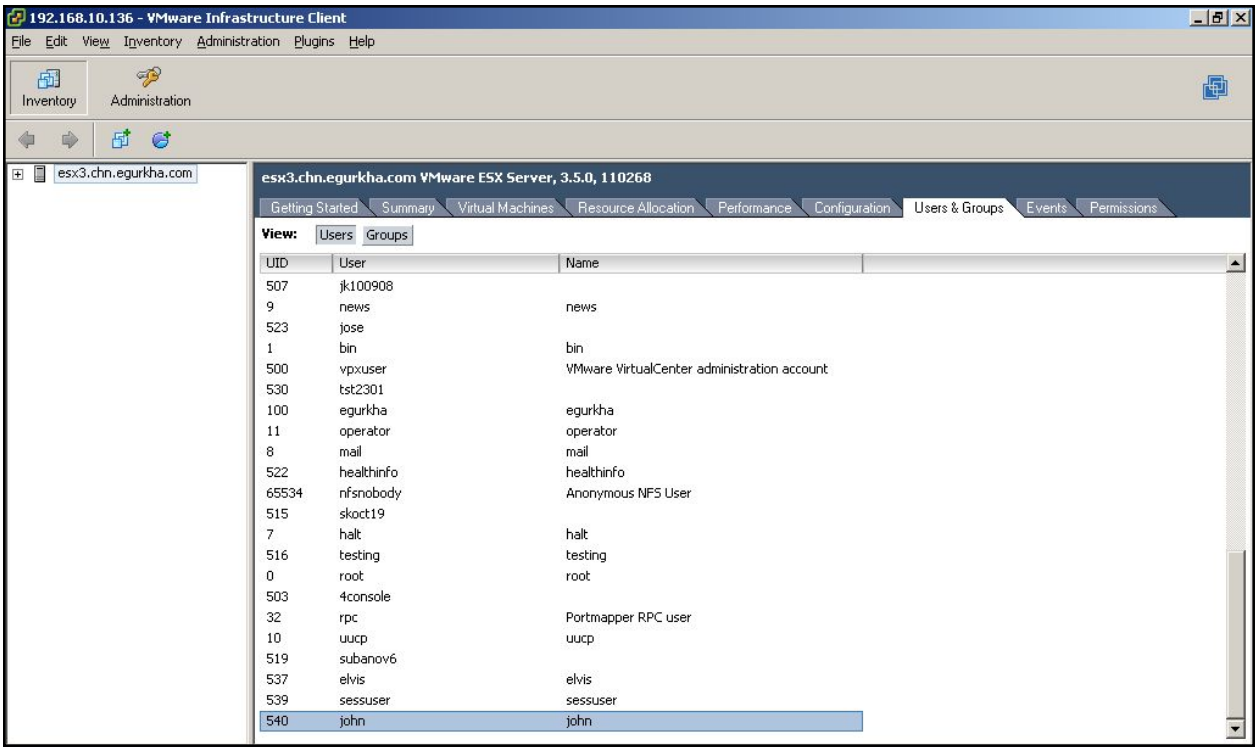


Figure 2.41: The newly added user name displayed in the list of users to the ESX host

8. Next, we need to assign a role with the **View and Terminate Sessions** privilege to this user. For that, we first need to create such a role. To achieve this, click on the **Administration** icon in the tool bar of Figure 2.41. Figure 2.42 then appears listing the roles that pre-exist on the ESX server.

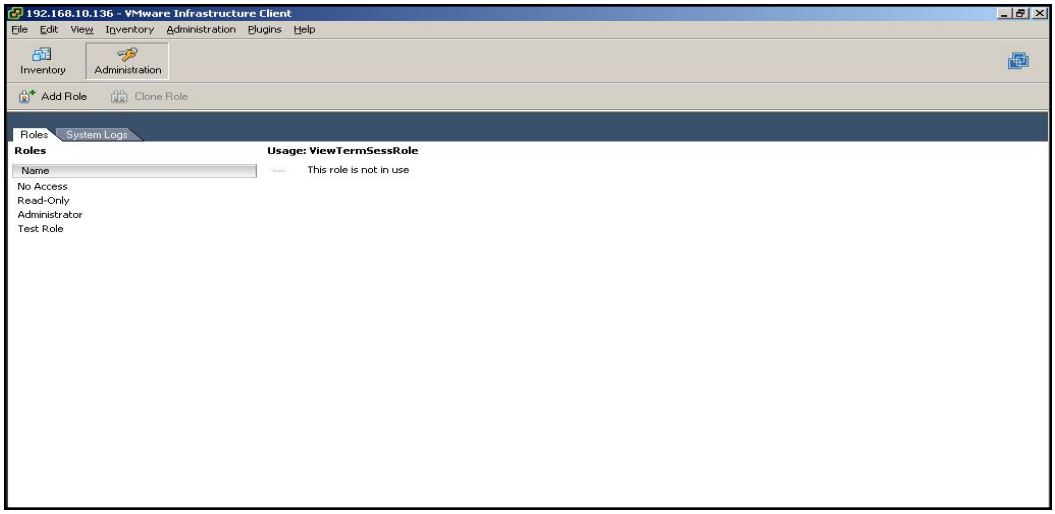


Figure 2.42: The list of roles

9. To create a new role, right-click anywhere in Figure 2.42, and pick the **Add** option from the shortcut menu that appears (see Figure 2.43).

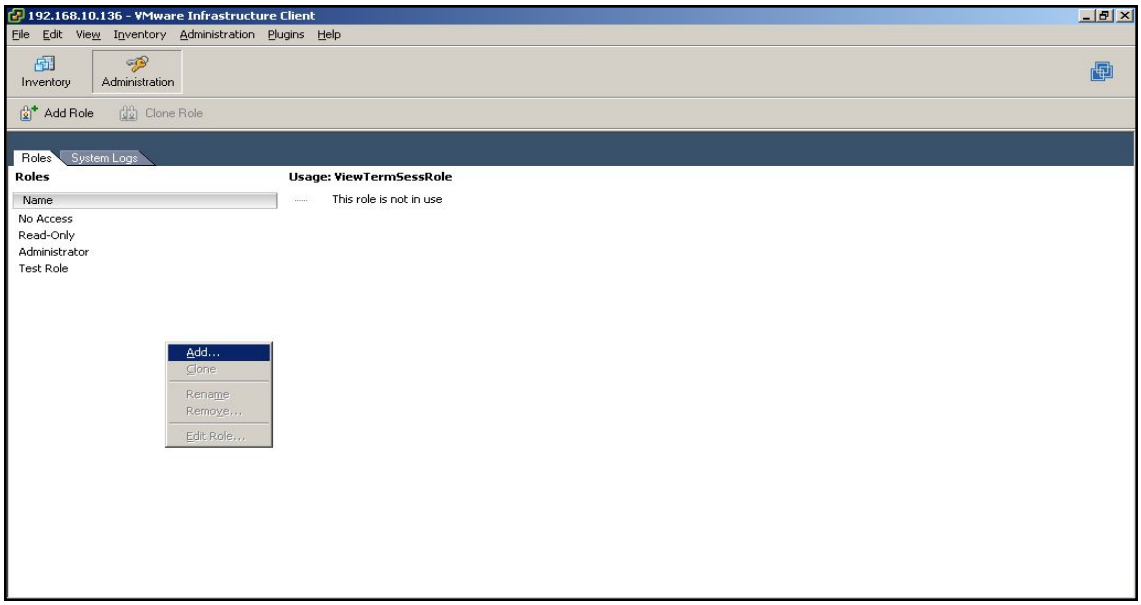


Figure 2.43: Selecting the Add option to add a new role

10. In Figure 2.44 that appears next, enter the name of the new role. Then, expand the **Sessions** node in the **Privileges** tree-structure below, and click the check box corresponding to the **View and Terminate Sessions** node (in an ESX server 3/3.5), or the **View and stop sessions** node (in vSphere 4.1 or above) (see Figure 2.45). Click the **OK** button to save the changes.

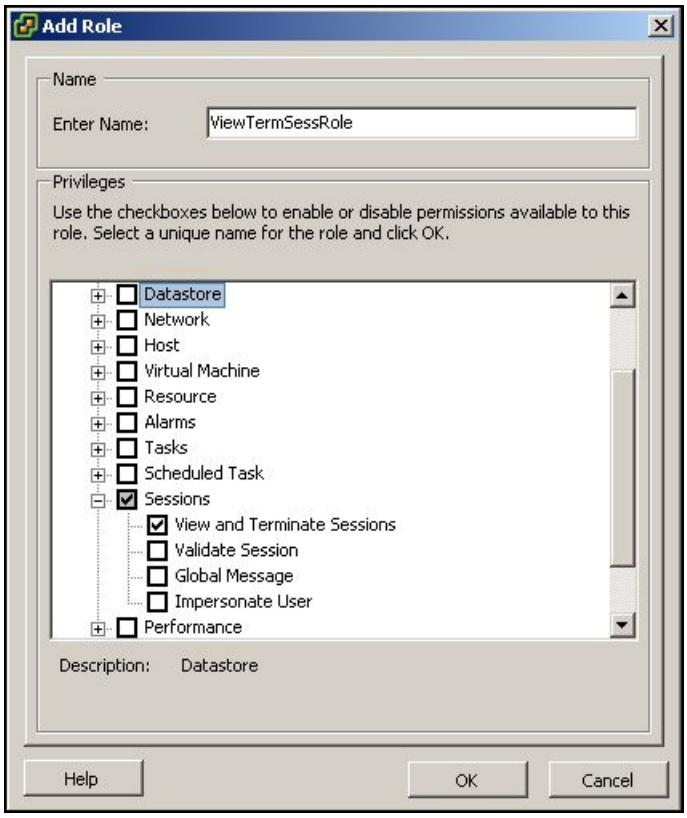


Figure 2.44: Creating a new role on an ESX server 3/3.5 with the 'View and Terminate Sessions' privilege

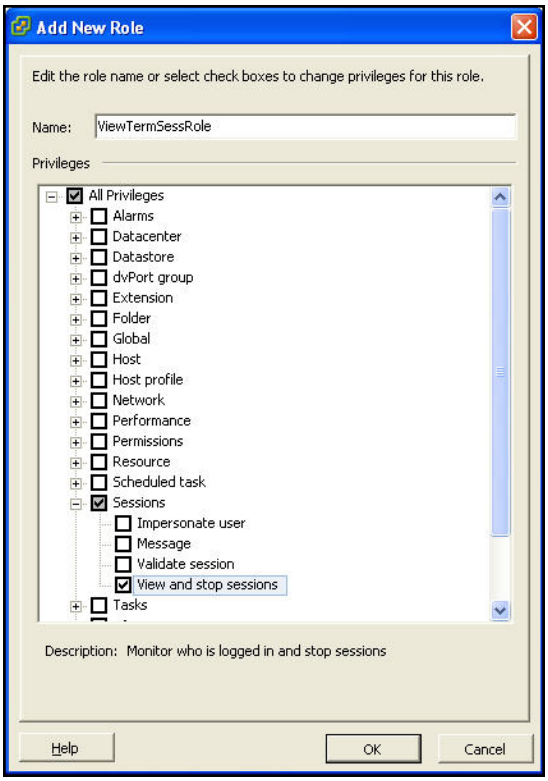


Figure 2.45: Creating a new role on vSphere 4.1 (and above) with the 'View and stop sessions' privilege

11. Figure 2.46 then appears displaying the newly created role.

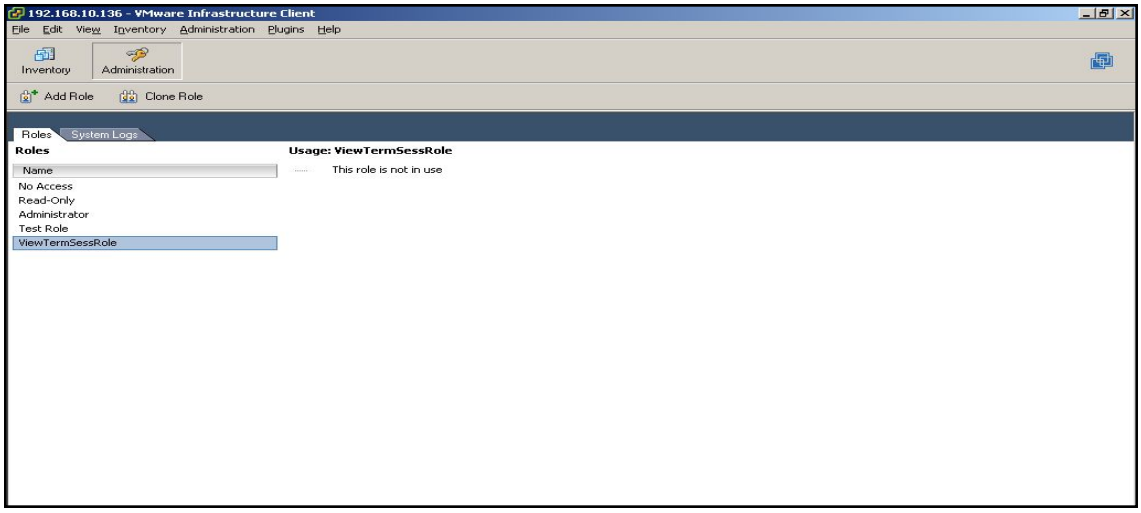


Figure 2.46: The newly created role appended to the list of roles

12. Now, proceed to assign this role to the new user we created earlier. For that, click on the **Inventory** icon indicated by Figure 2.46, and then click on the **Permissions** tab page. To add a new permission, right-click anywhere within the **Permissions** tab page, and choose the **Add Permission** option (see Figure 2.47).

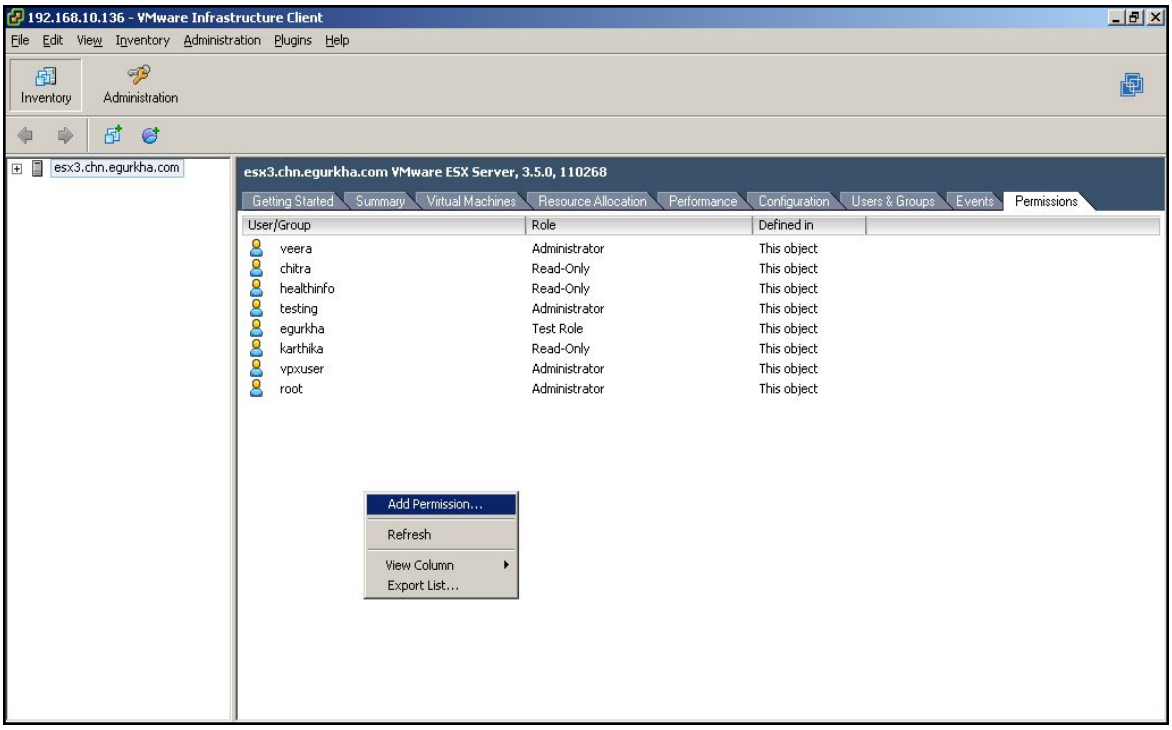


Figure 2.47: Figure 2.47: Selecting the Add Permission option

13. Figure 2.48 then appears. Click on the **Add** button in Figure 2.48.

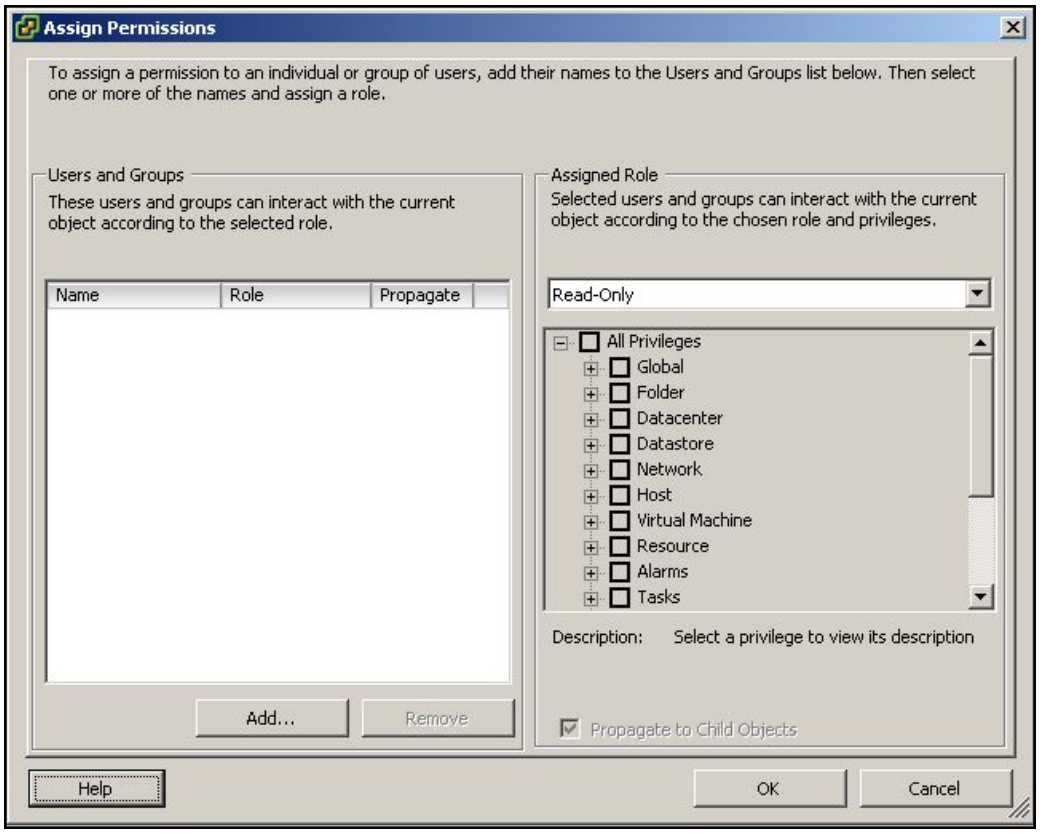


Figure 2.48: Clicking on the Add button

14. From the **Users and Groups** list that then appears, select the user that you just created, and click the **Add** button therein (see Figure 2.49).

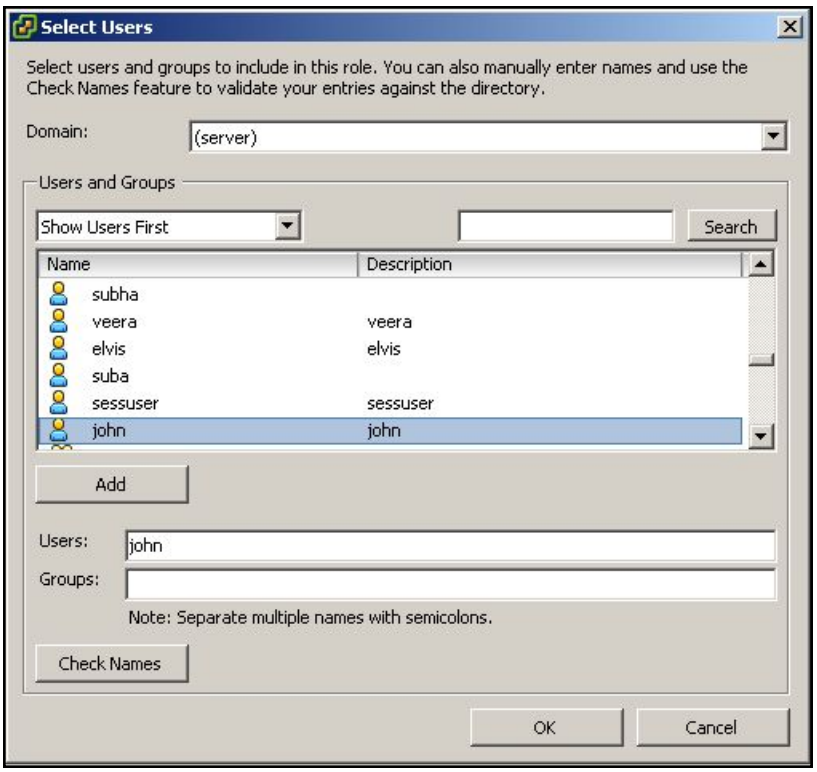


Figure 2.49: Selecting the user for whom a role is to be assigned.

15. Upon clicking, the selected user's name will appear against the **Users** box, as depicted by Figure 2.49.
16. Then, click the **OK** button in Figure 2.49. Figure 2.50 then appears, where you will find the selected user name displayed in the **Users and Groups** section. To assign the new role you created to this user, select the role name from the **Assigned Role** list in Figure 2.50, and click the **OK** button therein.

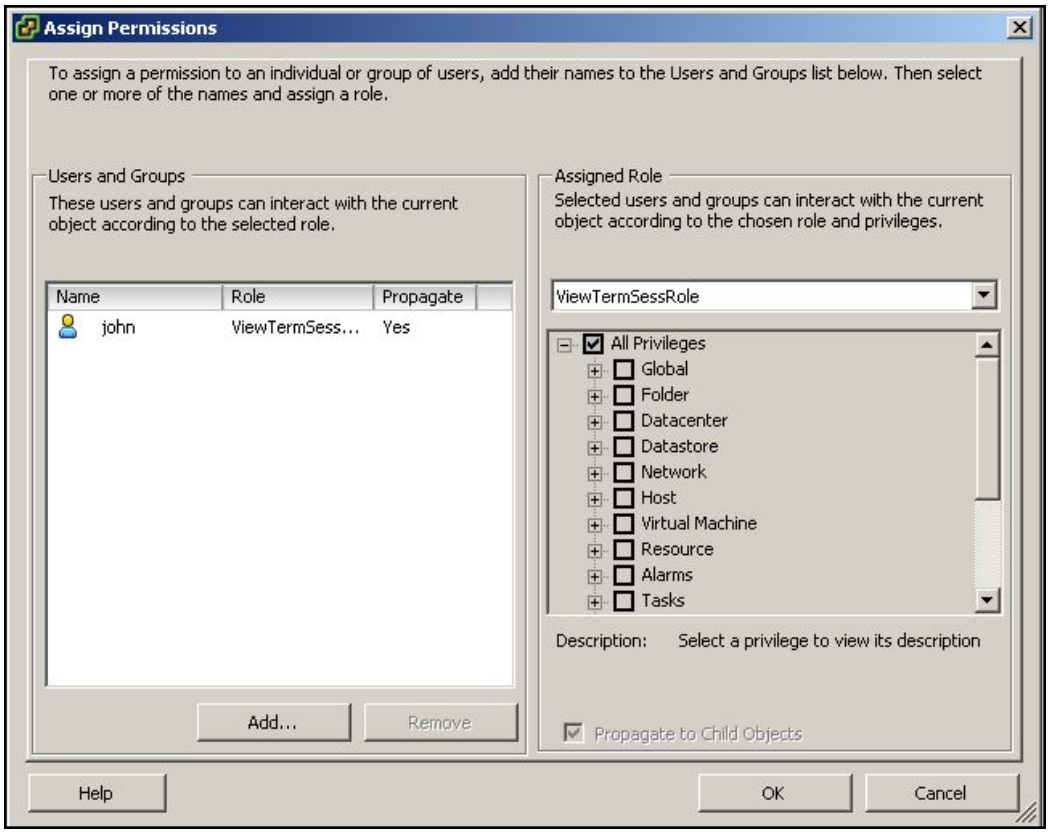


Figure 2.50: ssigning the new role to the user

17. Figure 2.51 then appears indicating the role assigned to the newly created user.

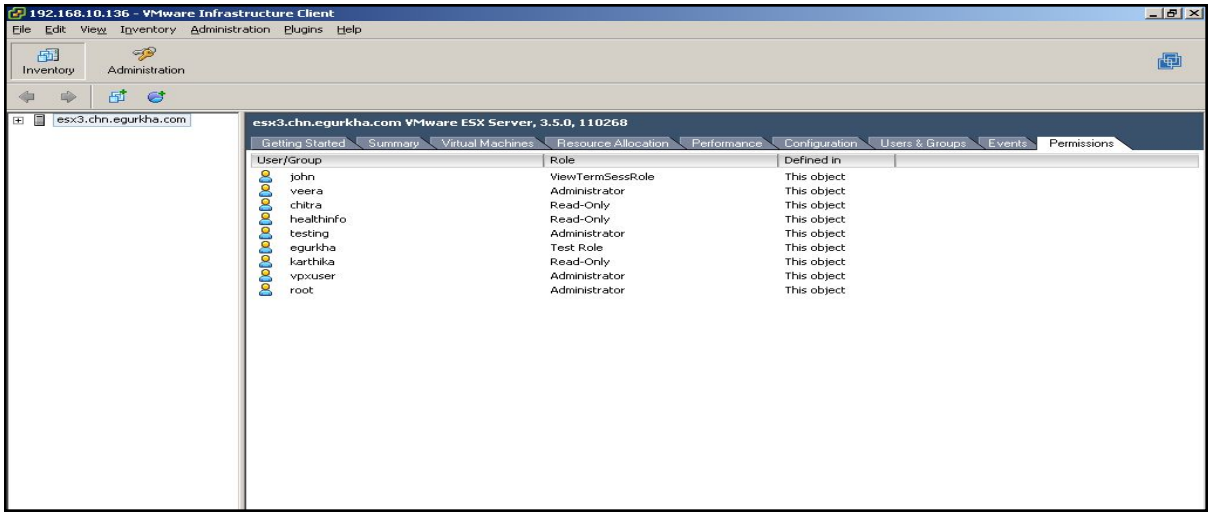


Figure 2.51: The new user and the role assigned to the user being displayed

2.7 Configuring the eG Agent to Collect Current Hardware Status Metrics

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters of all hardware tests must be configured with the credentials of a user on the vSphere server / vCenter server (depending upon whether monitoring is performed by connecting to the target vSphere server directly or to the vCenter server managing the target vSphere server) with the **Change Settings** privilege. For that, you need to create a special role on vSphere/vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new/existing user on vSphere/vCenter.

To create this special role on vCenter and assign it to a user, follow the steps detailed below:

1. Login to a system on which the **VMware Infrastructure Client** is installed.
2. Double-click on the **VMware Infrastructure Client** icon on your desktop.
3. Figure 2.52 then appears. To connect to the vCenter, select the **IP address / Name** of the vCenter, and then provide the login information. To grant access permissions to a user, you will have to login to vCenter as a user with rights to grant permissions to other users. Therefore, provide the **User name** and **Password** of such a user in Figure 2.52.



Figure 2.52: Connecting to vCenter

4. When the **Virtual Infrastructure Client** opens, click on the **Administration** icon indicated by Figure 2.53.

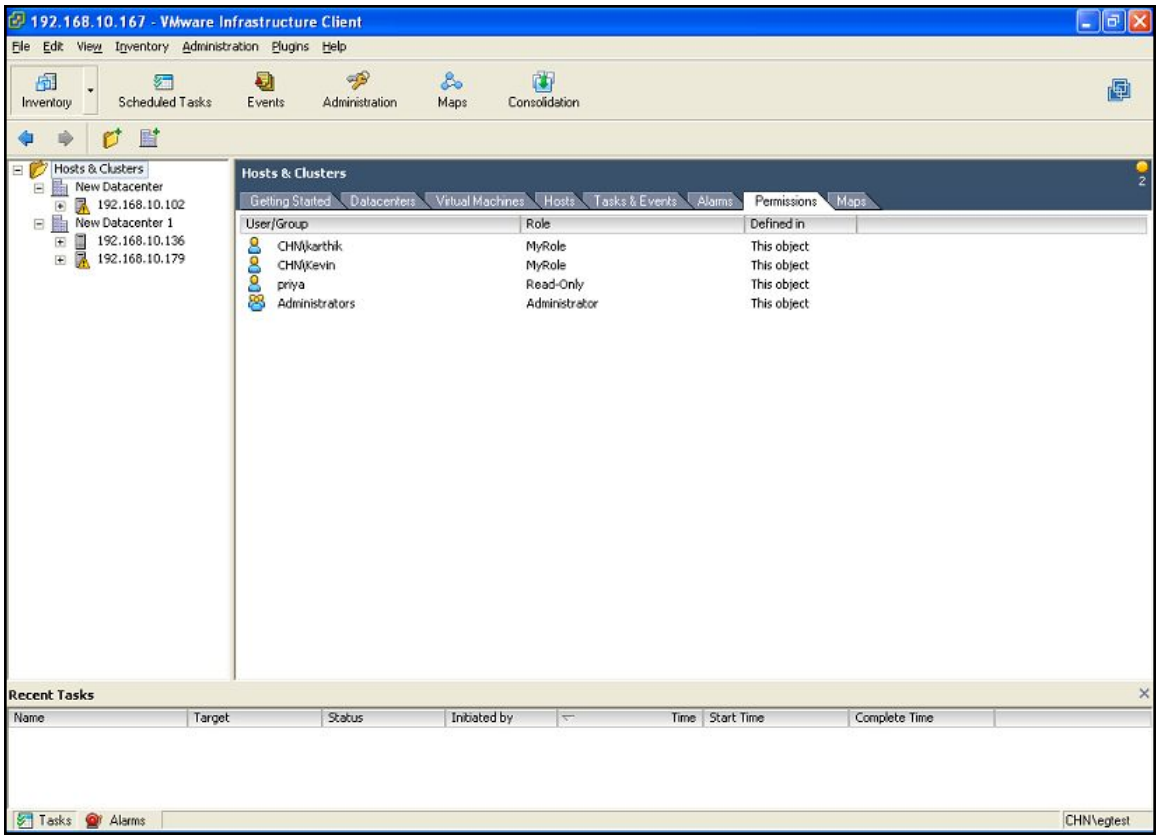


Figure 2.53: Clicking on the Administration icon

5. The **Roles** tab page opens by default, revealing the roles that pre-exist on vCenter (see Figure 2.54).

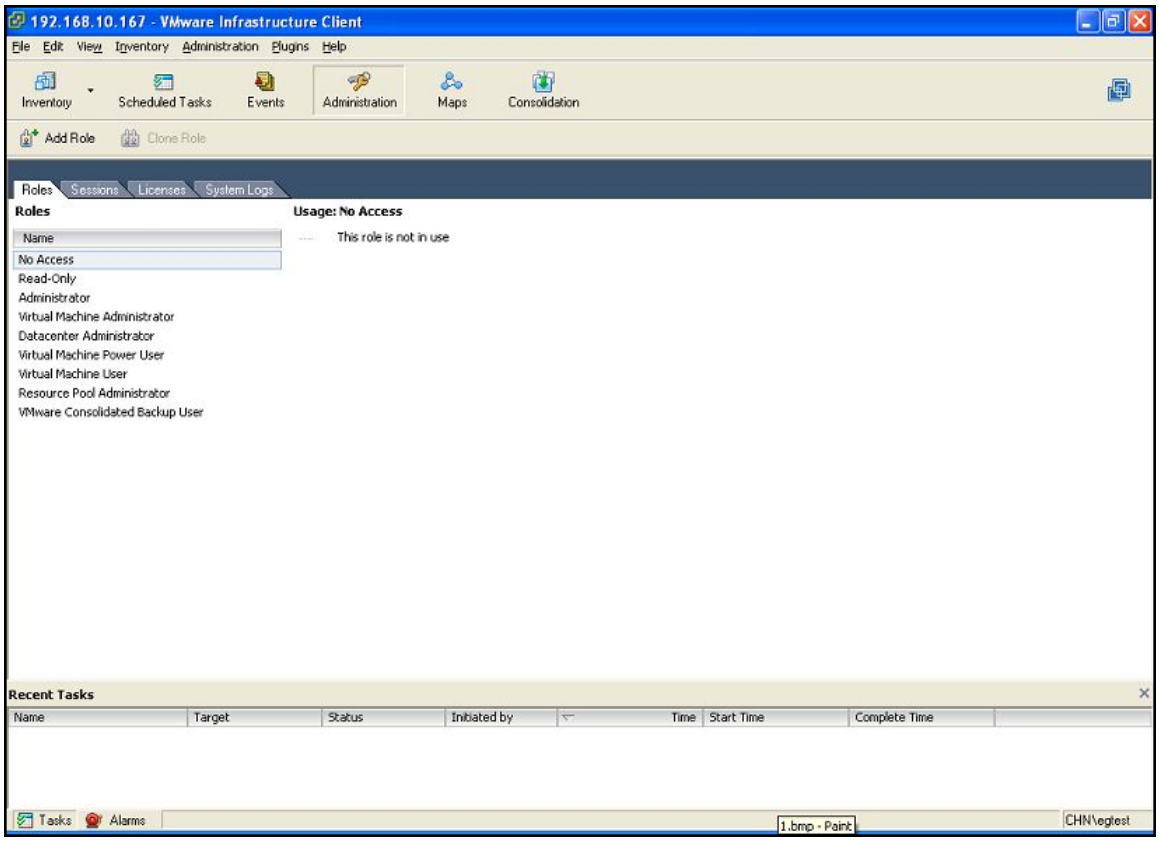


Figure 2.54: Viewing the roles available on the vCenter

6. To create a special role, right-click anywhere within Figure 2.54, and select **Add** from the shortcut menu that appears (see Figure 2.55).

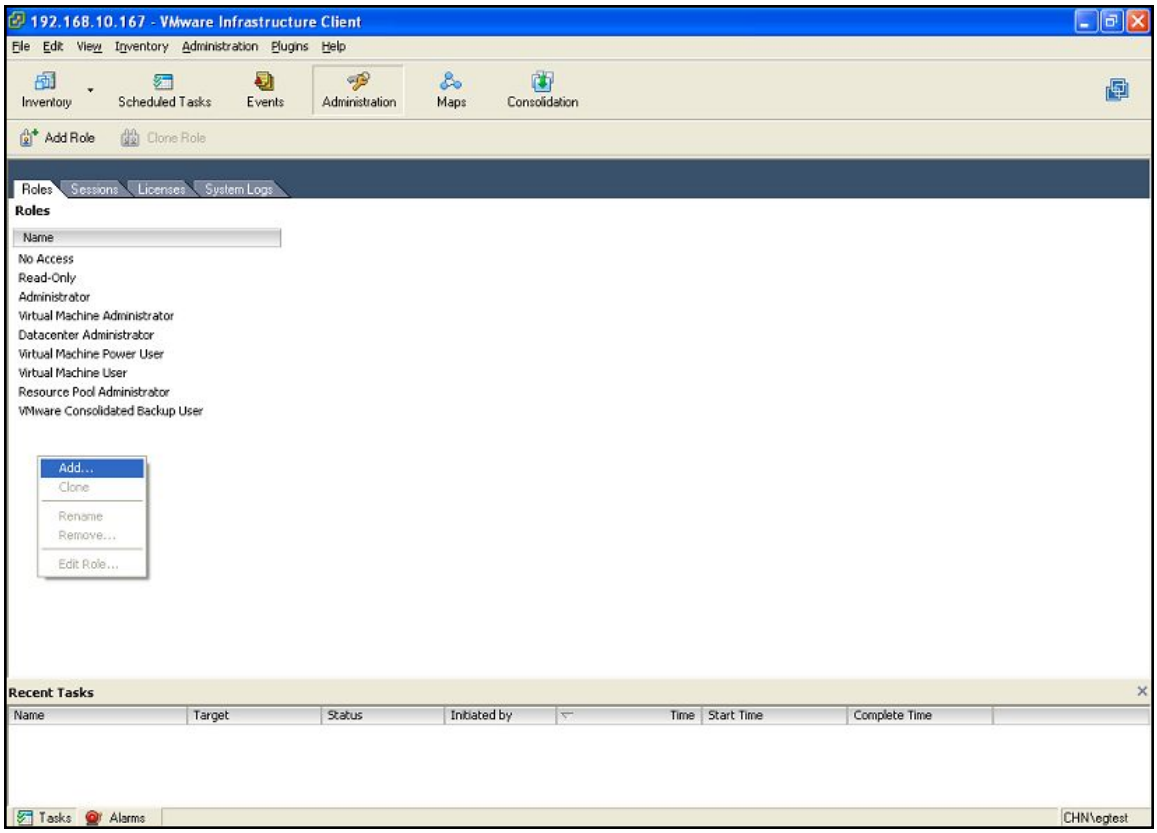


Figure 2.55: Selecting Add from the shortcut menu

7. Figure 2.56 then appears wherein you need to **Enter Name** of the new role. Then, expand the **Host** node in the **Privileges** tree-structure below, expand the **Configuration** sun-node of the **Host** node, and click the check box corresponding to the **Change Settings** node.

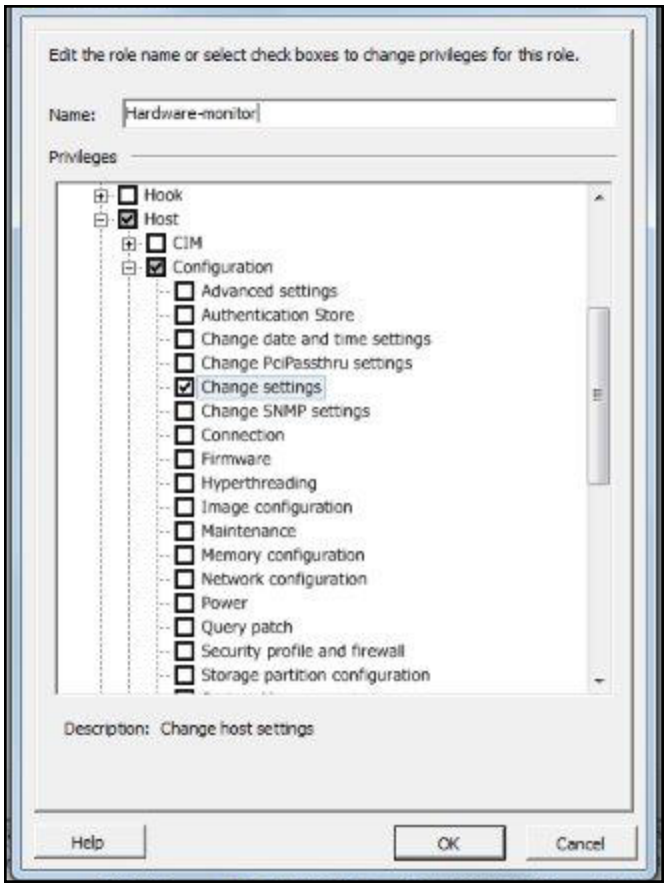


Figure 2.56: Creating a new role on a vCenter server with the 'Change Settings' privilege

8. Then, click the **OK** button in Figure 2.56 to save the changes. Now, to create a new user on vCenter and assign the new role to the new user, follow the steps 9-16 in Section **2.3**.

To create a special role for hardware monitoring on the target vSphere server and assign it to a user, follow the steps detailed below:

1. Login to a system on which the **VMware Infrastructure Client** is installed.
2. Double-click on the **VMware Infrastructure Client** icon on your desktop.
3. A login screen then appears. To connect to the monitored ESX host, select the **IP address / Name** of the ESX host from the list box, and then provide the login information. To create a new user on an ESX host, you will have to login to the host as a **root** user. Therefore, provide the **User name** and **Password** of the **root** user in the login screen.
4. The **VMware Infrastructure Client** console then opens (see Figure 2.57). This console has a tree-structured listing of VMs and resource pools on the left hand side and a series of tab pages on the right. The topmost node of the tree-structure represents the ESX host, which has the virtual machines (or guests) as its sub-nodes. To create a new user on the ESX host, first click on the **Users & Groups** tab page in the right panel (see Figure 2.57). The list of existing users on the ESX host is then displayed.

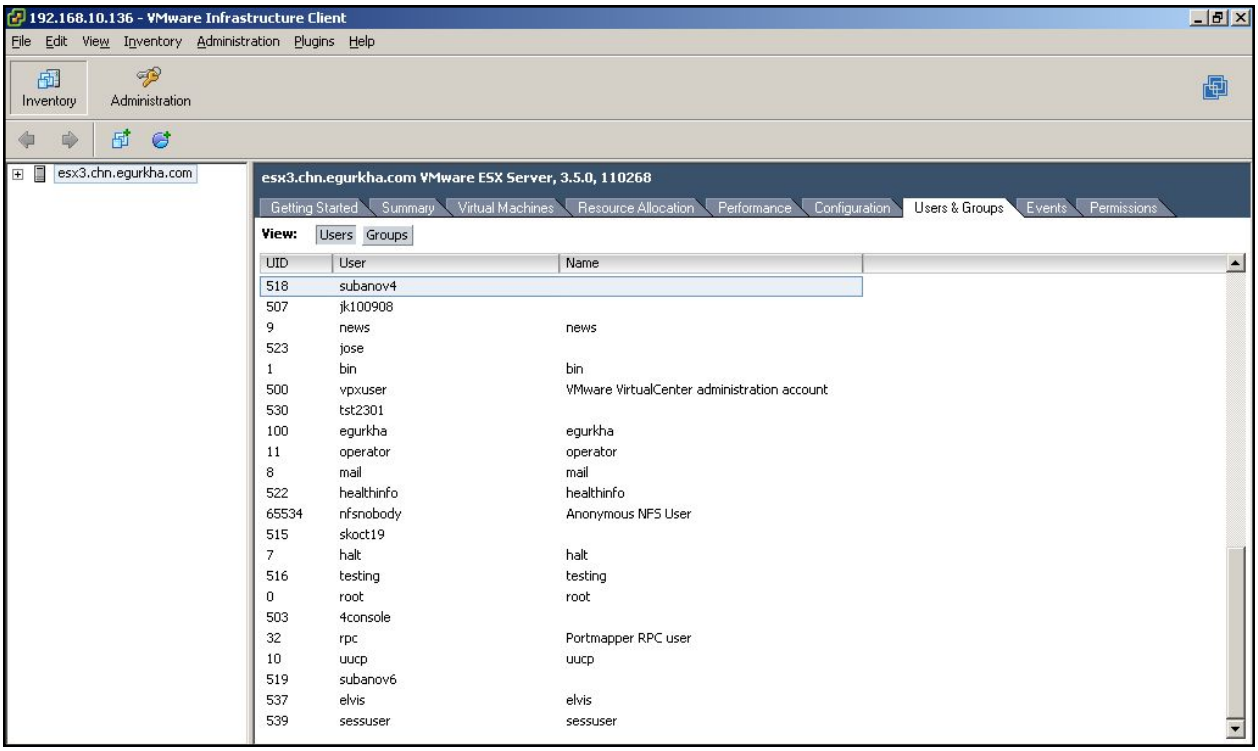


Figure 2.57: The VI Client console - Users & Groups tab page

5. Next, right-click any where within the **Users & Groups** tab page to invoke a shortcut menu. Select the **Add** option from the menu to add a new user (see Figure 2.58).

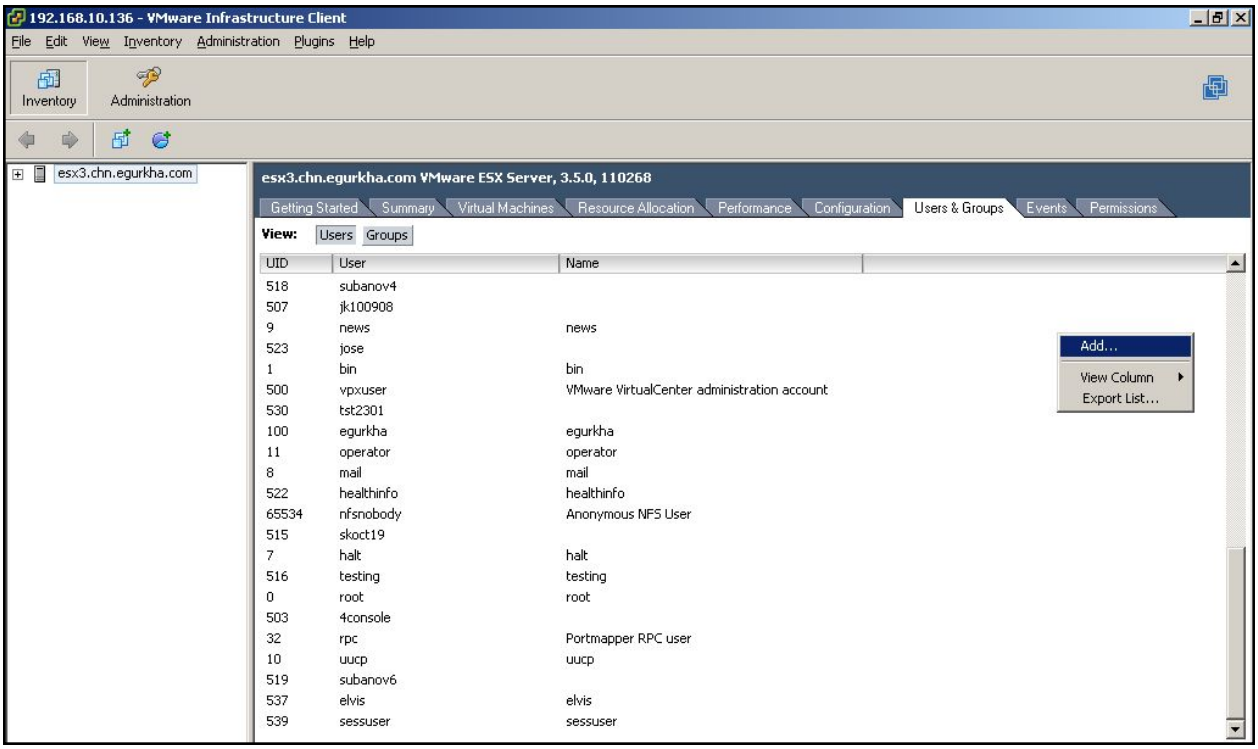


Figure 2.58: Selecting the Add option

6. Doing so opens Figure 2.59, wherein you need to provide the **Login** name of the new user, the **User Name**, and the **Password**. Confirm the password by retyping it in the **Confirm** text box, and click the **OK** button to complete the addition of the new user.

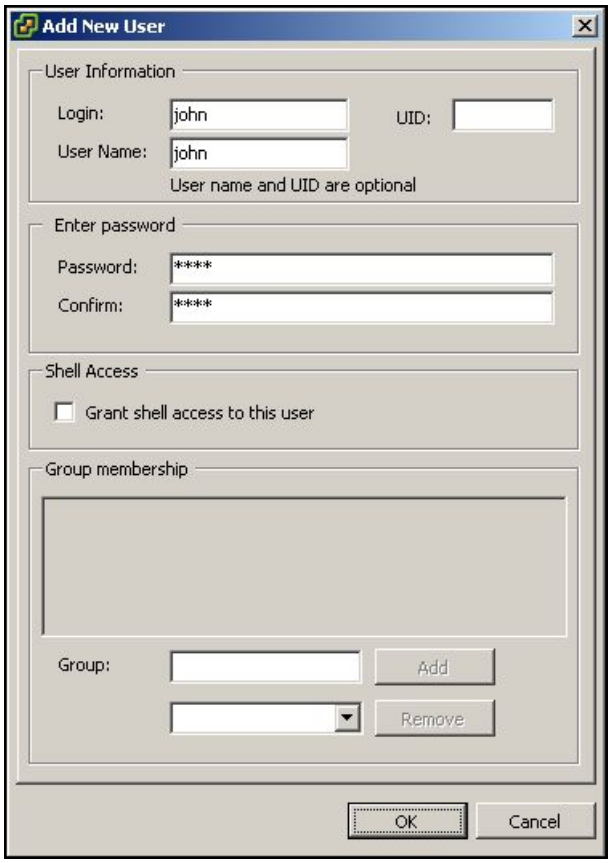


Figure 2.59: Adding a new user

7. Figure 2.60 then appears, where you will find the newly added user name displayed.

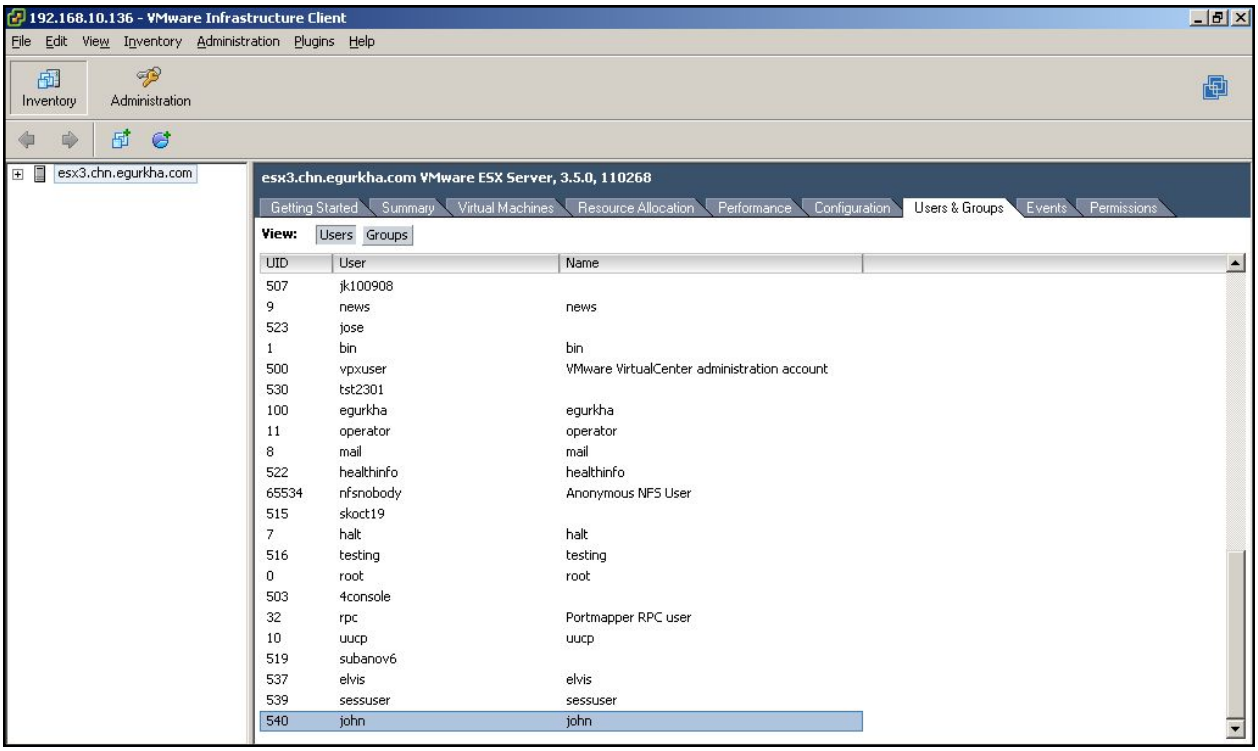


Figure 2.60: The newly added user name displayed in the list of users to the ESX host

8. Next, we need to assign a role with the **View and Terminate Sessions** privilege to this user. For that, we first need to create such a role. To achieve this, click on the **Administration** icon in the tool bar of Figure 2.60. Figure 2.61 then appears listing the roles that pre-exist on the ESX server.

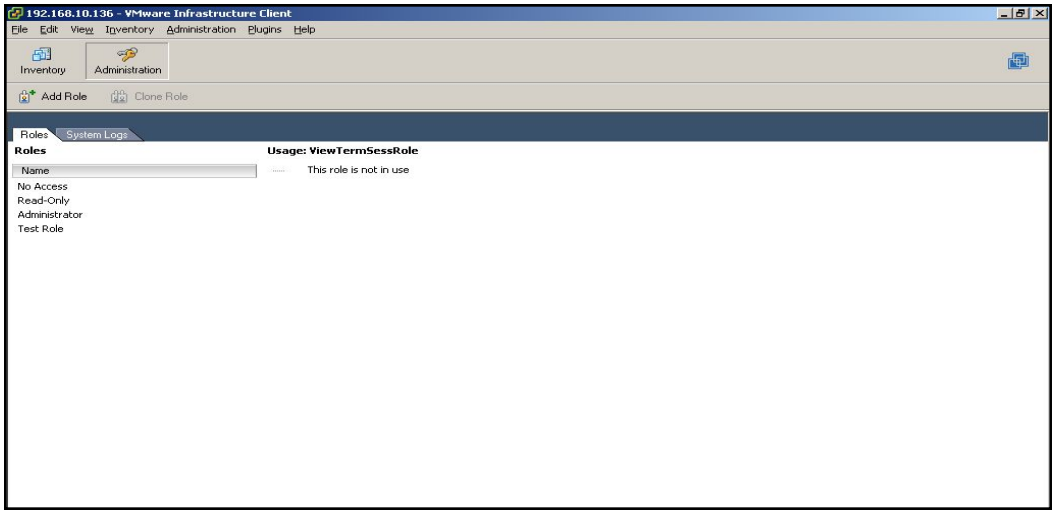


Figure 2.61: The list of roles

9. To create a new role, right-click anywhere in Figure 2.61, and pick the **Add** option from the shortcut menu that appears (see Figure 2.62).

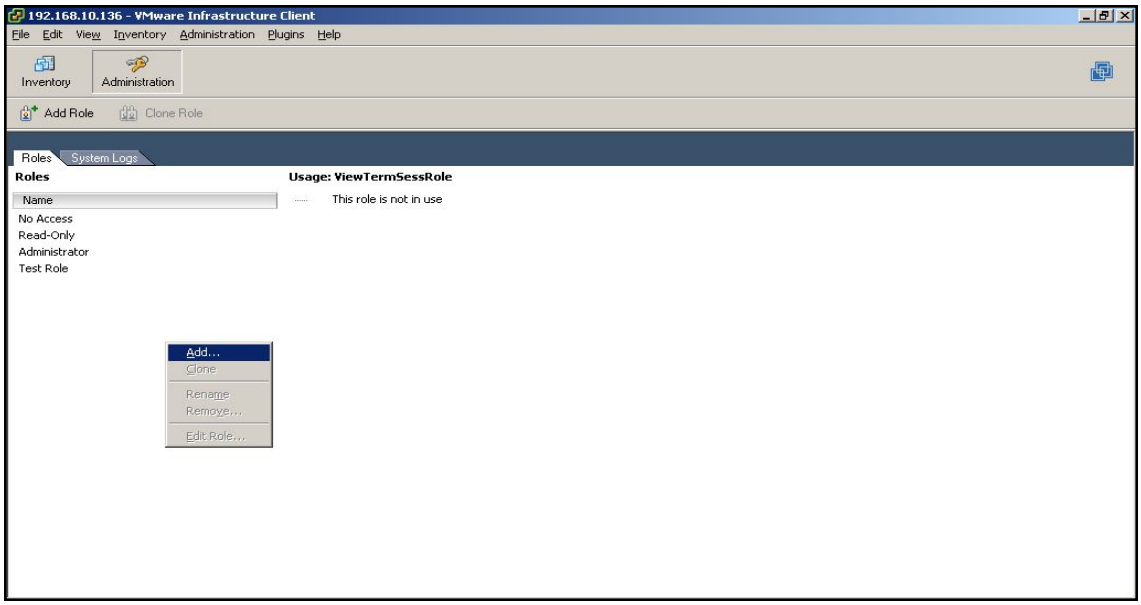


Figure 2.62: Selecting the Add option to add a new role

10. In Figure 2.63 that appears next, enter the name of the new role. Then, expand the **Host** node in the **Privileges** tree-structure below, expand the **Configuration** sun-node of the **Host** node, and click the check box corresponding to the **Change Settings** node. Click the **OK** button to save the changes.

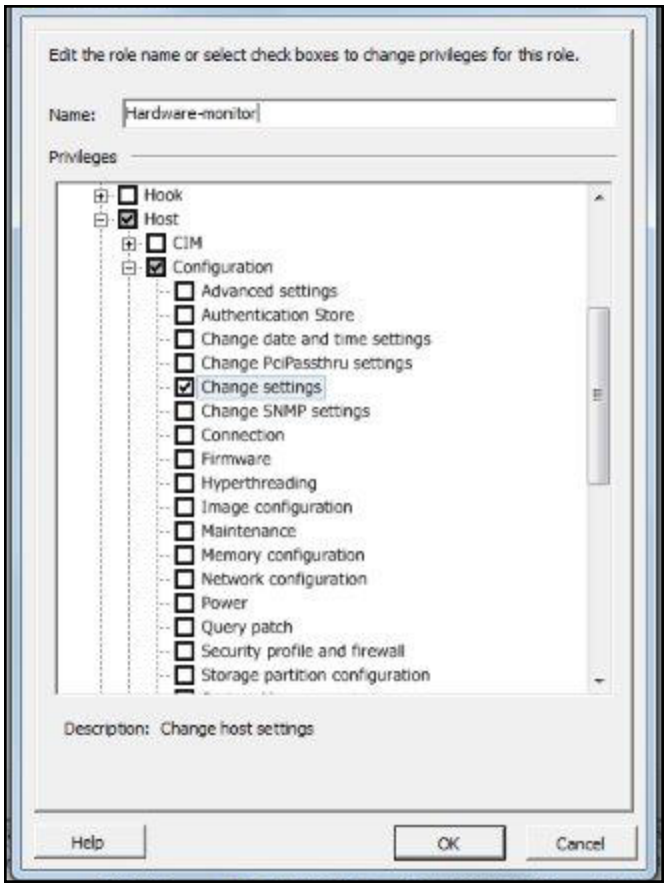


Figure 2.63: Creating a new role on a vSphere server with the 'Change Settings' privilege

11. Then, to assign this new role to a new user on vSphere, follow the steps 11-17 in Section **2.6** of this document.

2.8 Configuring the Remote Agent to Obtain the Inside View of Windows VMs, using the eG VM Agent

To provide the inside view of a Unix VM, the eG agent uses secure shell (SSH). To obtain the inside view of a Windows VM, the eG agent offers two options. The first option uses Windows File & Print Sharing services to push monitoring components to the VMs. These monitoring components are then executed on the VM to collect metrics from the VMs. To push monitoring components to the VM and to periodically invoke these components, the eG agent requires **domain administrator privileges** to all the VMs being monitored.

In many production environments, strict security restrictions are enforced, and it may not be possible to configure a monitoring solution with domain administration privileges for each of the VMs. To handle such environments, the eG VM monitor uses a lightweight monitoring component called the **eG VM Agent**, which is installed inside each of the VMs to obtain metrics regarding the health of the VMs. The **eG VM Agent** can be best described as a software that can be installed on the Windows virtual machines of a virtual

infrastructure to allow a single eG agent to obtain an inside view of these VMs, **without domain administrator privileges**.

2.8.1 Installing the eG VM Agent

Users have multiple options to choose from when it comes to installing the eG VM Agent. These options have been discussed below:

- Manually install the eG VM Agent on every Windows VM using the executable that eG Enterprise provides;
- Bundle the eG VM Agent as part of a template VM, and use this template to create multiple VMs; this way, the eG VM Agent is automatically available in all the VMs that are created using the template;
- Use a software distribution solution such as Microsoft System Center to distribute the eG VM Agent software to existing VMs from a central location;
- Connect to each Windows VM and silently install the eG VM Agent on it, without using the executable that eG Enterprise provides.

The first and fourth installation options alone are discussed here.

2.8.1.1 Using the Executable Provided by eG Enterprise

The detailed manual installation procedure has been discussed hereunder:

1. To install the eG VM Agent on a 32-bit VM, double-click on the **eGVMAgent.exe**, and to install the same on a 64-bit VM, double-click the **eGVMAgent_64.exe**.
2. Figure 2.64 then appears. Click on the **Next** button in Figure 2.64 to continue.

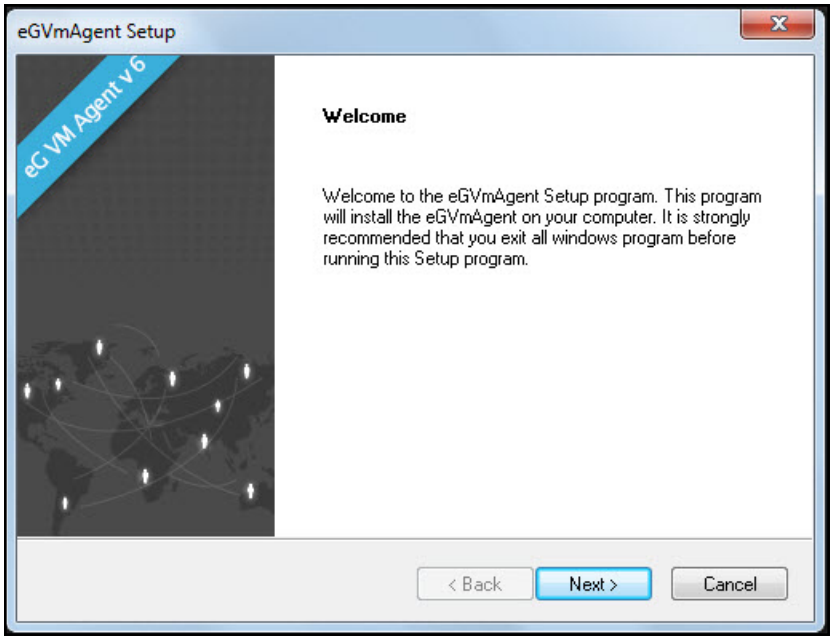


Figure 2.64: Welcome screen of the eG VM Agent installation wizard

3. When Figure 2.65 appears, click on **Yes** to accept the displayed license agreement.

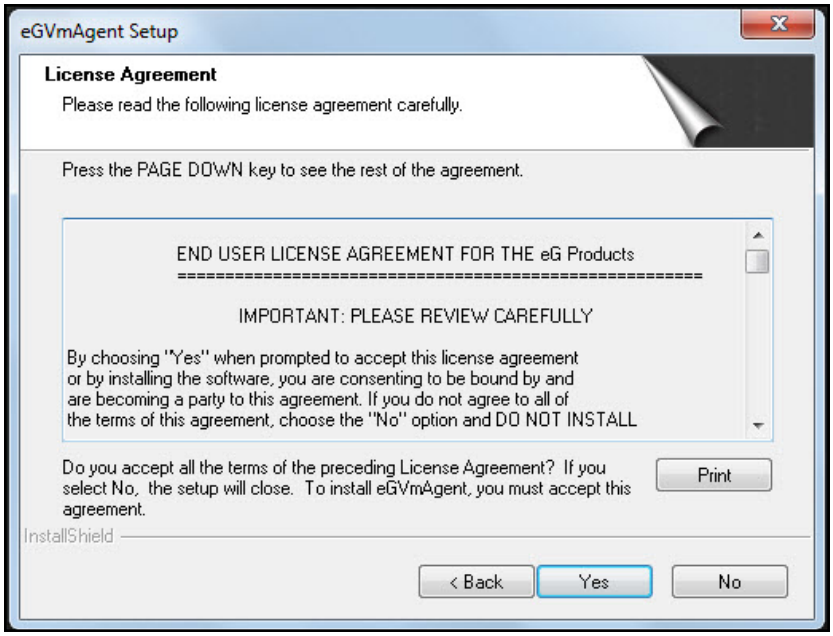


Figure 2.65: Accepting the license agreement

4. Use the **Browse** button in Figure 2.66 to indicate the location in which the agent should be installed, and click the **Next** button to proceed.

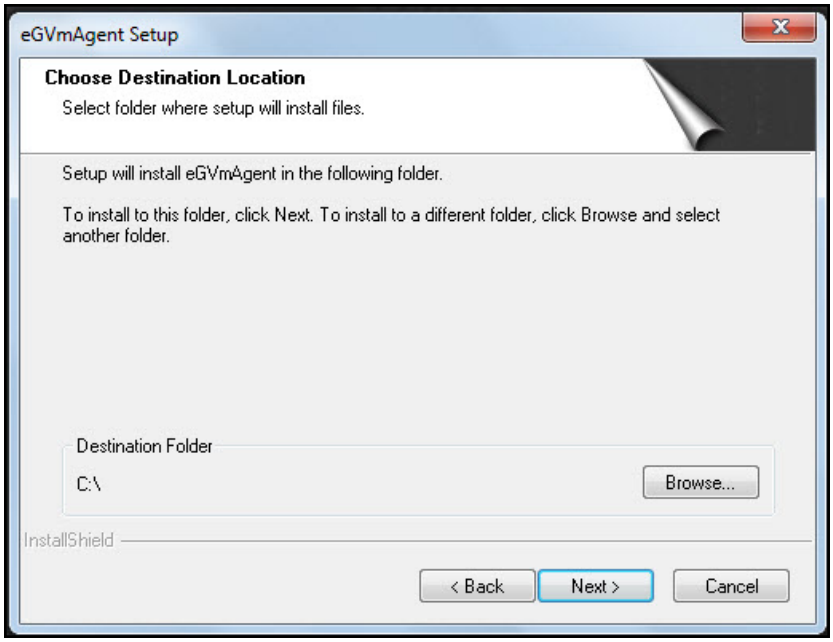


Figure 2.66: Specifying the install directory of the eG VM Agent

5. Next, specify the port at which the VM agent listens for requests from the eG agent. The default port is 60001. After port specification, click on the **Next** button in Figure 2.67 to proceed.

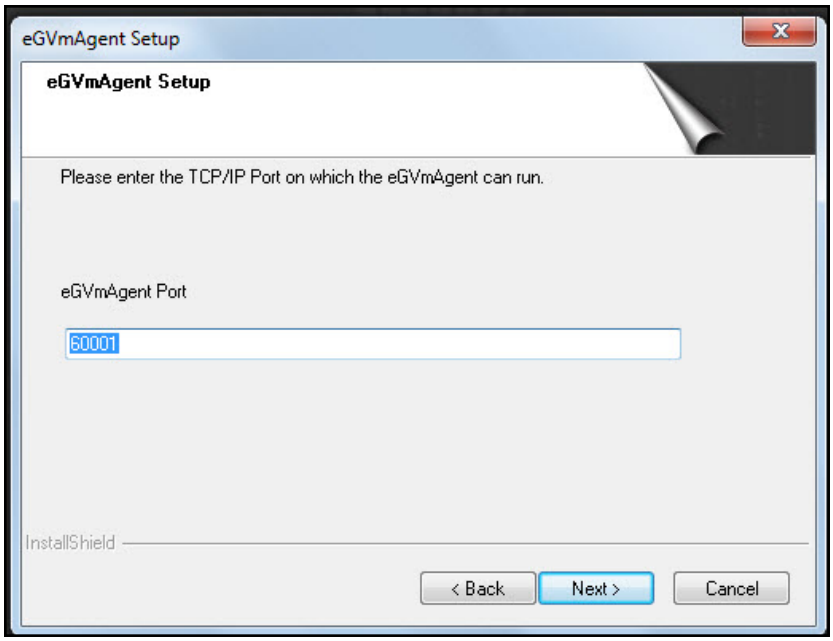


Figure 2.67: Specifying the VM agent port

6. A summary of your specifications then follows (see Figure 2.68). Click **Next** to proceed.

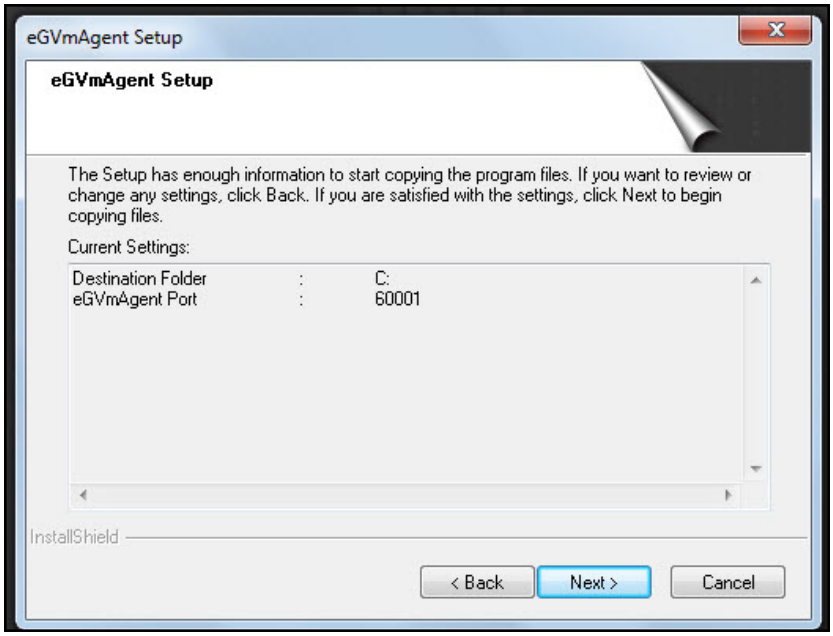


Figure 2.68: A summary of your specifications

7. Finally, click the **Finish** button in Figure 2.69 to complete the installation.

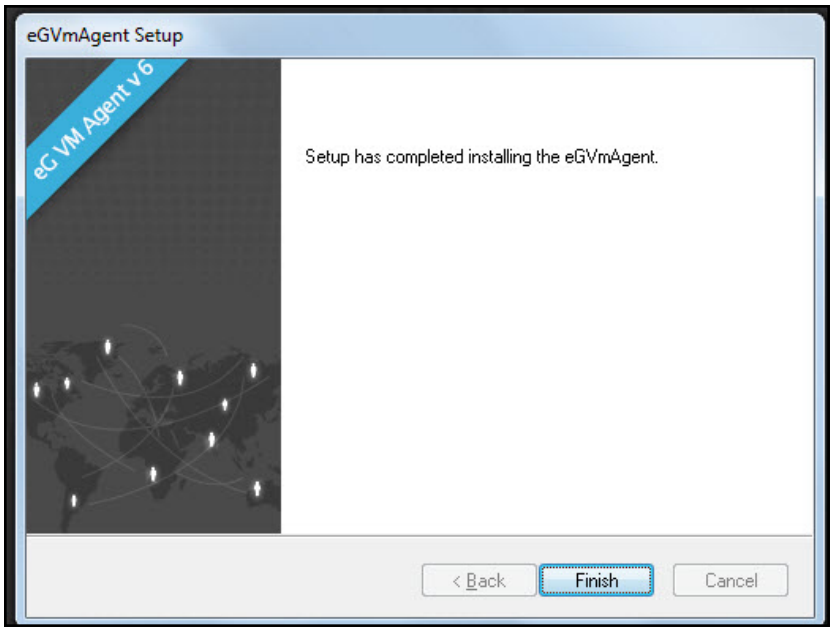


Figure 2.69: Finishing the installation

2.8.2 Silent Installation of the eG VM Agent

To silently install the eG VM agent on Windows VMs, follow the broad steps outlined below:

1. Creating silent mode script for eGVmagent installation
2. Installing eGVmAGent in silent mode

Each of these steps have been discussed elaborately below.

2.8.2.1 Creating a Silent Mode Script

For this, follow the procedure detailed below:

1. Login to a target Windows VM.
2. From the command prompt, run the following command to launch the normal mode installation of the eG VM Agent.

   *eGVMAgent_**<32/64>**.exe /a /r /f1"<Full path to the script file into which the installation inputs will be stored>"*

   For example:

   eGVMAgent_x64.exe /a /r /f1"C:\script\eGVMAgent.iss"
3. Upon execution, this command will automatically create a script file of the given name in the location mentioned in the command.
4. Command execution will also begin the normal mode installation of the eG VM Agent. Provide inputs as and when necessary to proceed with the installation.
5. These inputs will be automatically recorded in the script file that was created in step 3.

2.8.2.2 Installing the eG VM Agent in the Silent Mode

Follow the steps given below to install the eG VM Agent in the silent mode:

1. Login to the Windows VM where the script file containing the inputs for installation resides.
2. Copy the script file from this VM to the Windows VM on which you want to install the eG VM Agent in the silent mode.
3. Copy the eG VM Agent installation executable also to the target Windows VM.
4. Next, on the target Windows VM, run the following command from the command prompt:

   *eGVMAgent_* ***<32/64>****.exe /a /s /f1"****<Full path to the script file containing the inputs for the installation>****"*

   For example:

   eGVmAgent_x64.exe /a /s /f1"C:\script\eGVMAgent.iss"

5. Upon successful execution, this command will automatically install the eG VM Agent on the target Windows VM.
6. You can then repeat steps 1-5 on each Windows VM where you want to install the eG VM Agent.

2.8.3 Communication between the eG Agent and the eG VM Agent

At the time of the installation of the eG VM agent, a folder named **eGVMAgent** is created in the install destination specified. The setup program also creates a Windows Service named **eGVMAgent** on the Windows VM. This service must be running for the eG agent to obtain the inside view of the virtual machine.

Upon successful installation, the eG VM agent starts automatically and begins listening for requests at default TCP port 60001. However, if, during the installation process, you have configured a different port for the eG VM agent, then, after completing the installation, follow the steps below to make sure that the eG agent communicates with the eG VM agent via the port that you have configured:

- Login to the eG manager host.
- Edit the **eg_tests.ini** file in the **<EG_INSTALL_DIR>\manager\config** directory.
- The **WmiInsideViewPort** parameter in the **[AGENT_SETTINGS]** section of the file is set to **60001** by default. If the eG VM agent's port is changed at the time of installation, then you will have to ensure that this parameter reflects the new port. Therefore, change the default port specification accordingly.
- Save the file.

At configured intervals, the eG remote agent issues commands to each of the eG VM Agents (using the TCP port configured during the VM agent installation). The eG VM Agent executes the commands, collects the "inside view" metrics from the Windows VM, and sends the output back to the eG agent. The eG agent then analyzes the metrics and informs the eG manager about the status of the Windows VMs.

2.8.4 Licensing of the eG VM Agent

The eG VM Agent is not license-controlled. Therefore, you can install and use any number of VM agents in your infrastructure.

2.8.5 Benefits of the eG VM Agent

The eG VM Agent offers several key benefits:

- **Ideal for high-security environments:** The eG VM Agent is capable of collecting "inside view" metrics from Windows VMs, without domain administrator privileges. It is hence ideal for high-security environments, where administrators might not be willing to expose the credentials of the domain administrators.
- **Easy to install, configure:** The eG VM Monitor offers users the flexibility to choose from multiple methodologies for installing the eG VM Agent on the target VMs. Even a manual installation procedure, would not take more than a few minutes. Moreover, since the eG VM agent communicates only with the eG agent and not the eG manager, no additional configuration needs to be performed on the VM agent to facilitate the communication. In addition, the VM agent starts automatically upon installation, thereby saving the time and trouble involved in manually starting each of the VM agents.
- **License independent:** Since the eG VM agent is not license-controlled, you can add any number of VM agents, as and when required, to your environment.

2.9 Configuring Windows Virtual Machines to Support the eG Agent's Inside View without the eG VM Agent

For the "inside" view, by default, the eG agent uses SSH/WMI (depending upon the virtual OS to be monitored) to communicate remotely with the virtual machines on the ESX server and collect metrics. To establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. Besides, the **INSIDE VIEW USING** flag of all "inside view" tests should be set to **Remote connection to a VM**.

In addition, the following pre-requisites need to be fulfilled:

a. The **ADMIN$** share will have to be available on the Windows guests

b. The Windows Firewall should be configured to allow Windows File and Print Sharing

The sections to come discuss the procedure to be followed for fulfiliing the 2 requirements above.

2.9.1 Enabling ADMIN$ Share Access on Windows Virtual Guests

2.9.1.1 Enabling ADMIN$ Share Access on Windows 2000/2003 VMs

If the **ADMIN$** share is not available on any Windows-based virtual guest, create the share using the procedure detailed below:

1. Open the Windows Explorer on the virtual machine, browse for the corresponding **Windows** directory in the C drive, right-click on it, and select the **Sharing** option from the shortcut menu.
2. If the **ADMIN$** share does not pre-exist on the Windows guest, then Figure 2.70 appears indicating the same.



Figure 2.70: The ADMIN$ share does not exist

3. On the other hand, if the **ADMIN$** share pre-exists, Figure 2.71 appears. In such a case, first, remove the **ADMIN$** share by selecting the **Do not share this folder** option from Figure 2.71 and clicking the **Apply** and **OK** buttons. After this, you will have to repeat step 1 of this procedure to open Figure 2.70. Then, proceed as indicated by step 3 onwards.

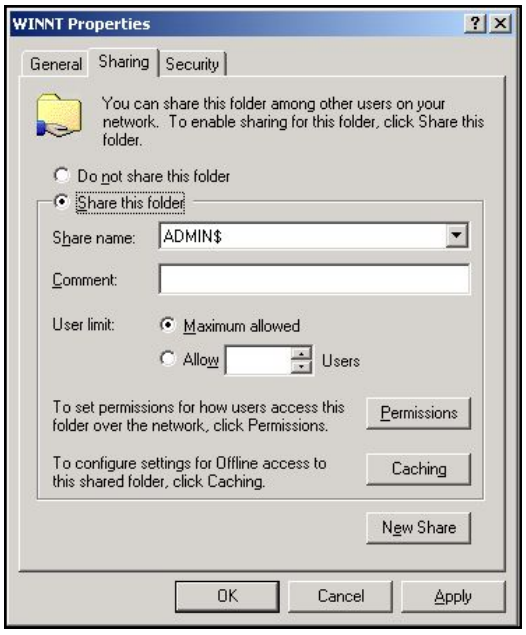


Figure 2.71: Admin$ share pre-exists

4. To create (or re-create) the **ADMIN$** share, select the **Share this folder** option from Figure 2.72, and provide **ADMIN$** share against the **Share name** text box (see Figure 2.72).

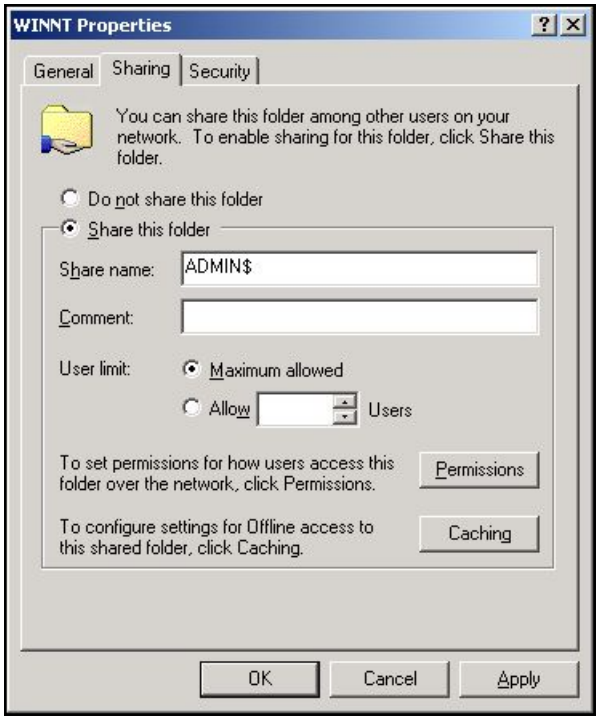


Figure 2.72: Creating the ADMIN$ share

5. Next, to enable the eG agent to communicate effectively with the Windows guest, you need to ensure that the permission to access the **ADMIN$** share is granted to an administrative user (local/domain); also, the **credentials of this user should be passed while configuring the eG monitoring**

capabilities - i.e., while configuring the VMware tests. To grant the access permissions, click on the **Permissions** button in Figure 2.72.

6. By default, the **ADMIN$** share can be accessed by **Everyone** (see Figure 2.73). To grant access rights to a specific administrative (local/domain) user, select the **Add** button in Figure 2.73. When Figure 2.74 appears, select the domain to search from the **Look in** list. The valid user accounts configured on the chosen domain then appear in the box below. From this box, choose the administrator's account and click on the **Add** button to add the chosen user account to the box below the **Add** button.

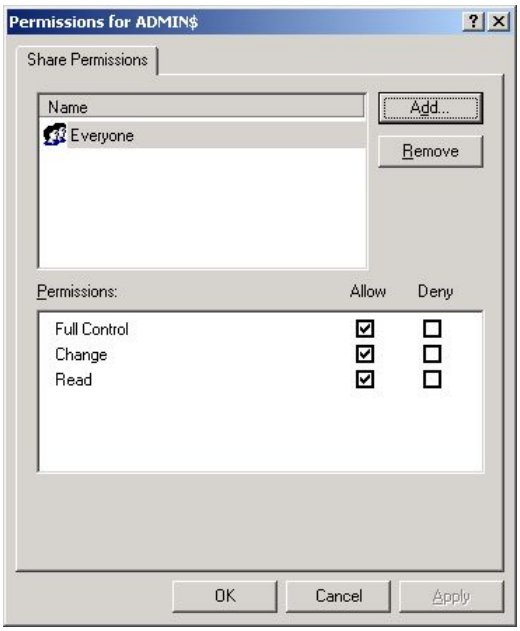


Figure 2.73: Clicking the Add button

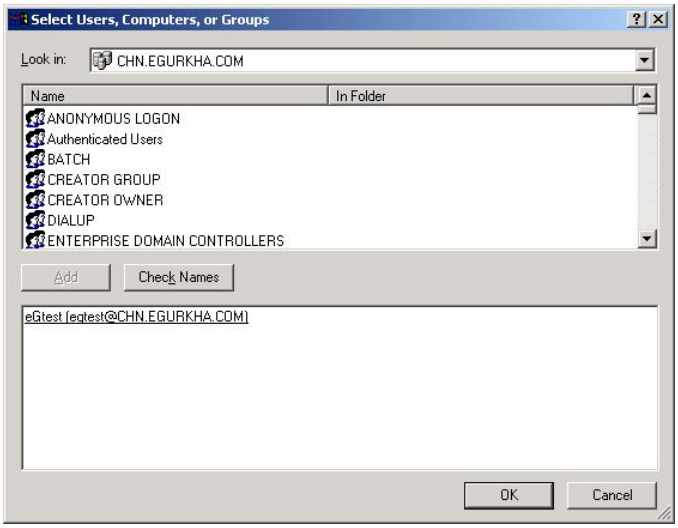


Figure 2.74: Selecting the administrative user to whom access rights are to be granted

7. Finally, click the **OK** button. You will then switch to Figure 2.75, where the newly added administrator account will appear.

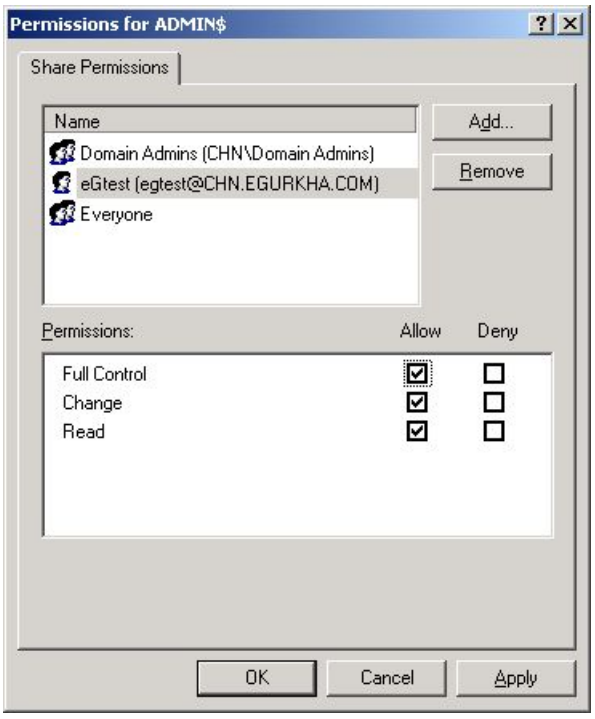


Figure 2.75: The administrator account granted access permissions

8. Select the newly added administrator account from Figure 2.75, and then, using the **Permissions** section, grant the administrator **Full Control**, **Change**, and **Read** permissions.
9. Finally, click the **Apply** and **OK** buttons in Figure 2.75 to register the changes.
10. Once you return to Figure 2.72, click on the **Security** tab to define the security settings for the **ADMIN$** share (see Figure 2.76).

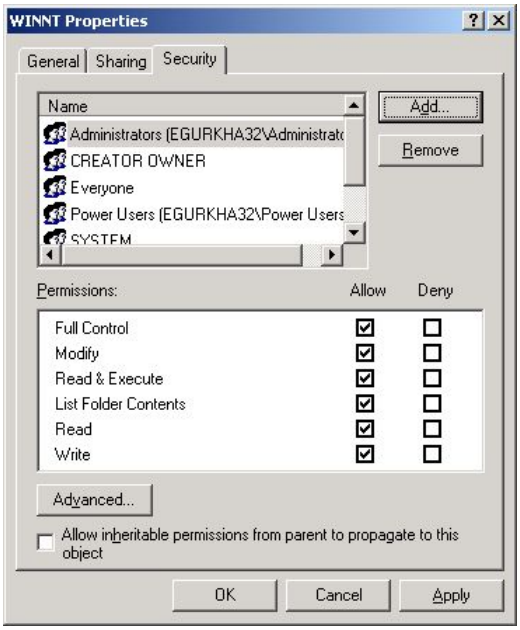


Figure 2.76: Defining the Security settings for the ADMIN$ share

11. Here again, you need to add the same administrator account, which was granted access permissions earlier. To do so, click the **Add** button in Figure 2.76, pick a domain from the **Look in** list of Figure 2.77, select the said administrator account from the domain users list below, and click the **Add** button (in Figure 2.77) to add the chosen account. Then, click the **OK** button in Figure 2.77.

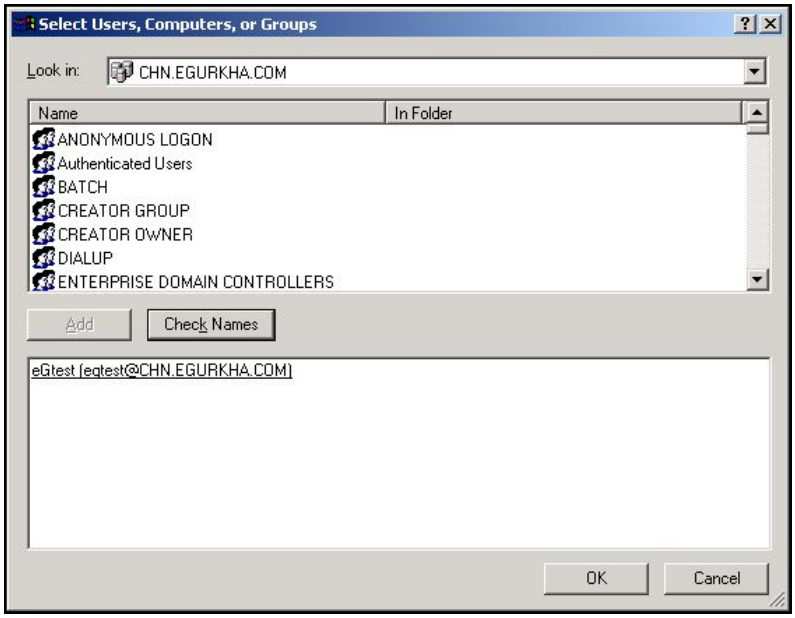


Figure 2.77: Adding the administrator account

12. This will bring you back to Figure 2.76, but this time, the newly added domain administrator account will be listed therein as indicated by Figure 2.78.

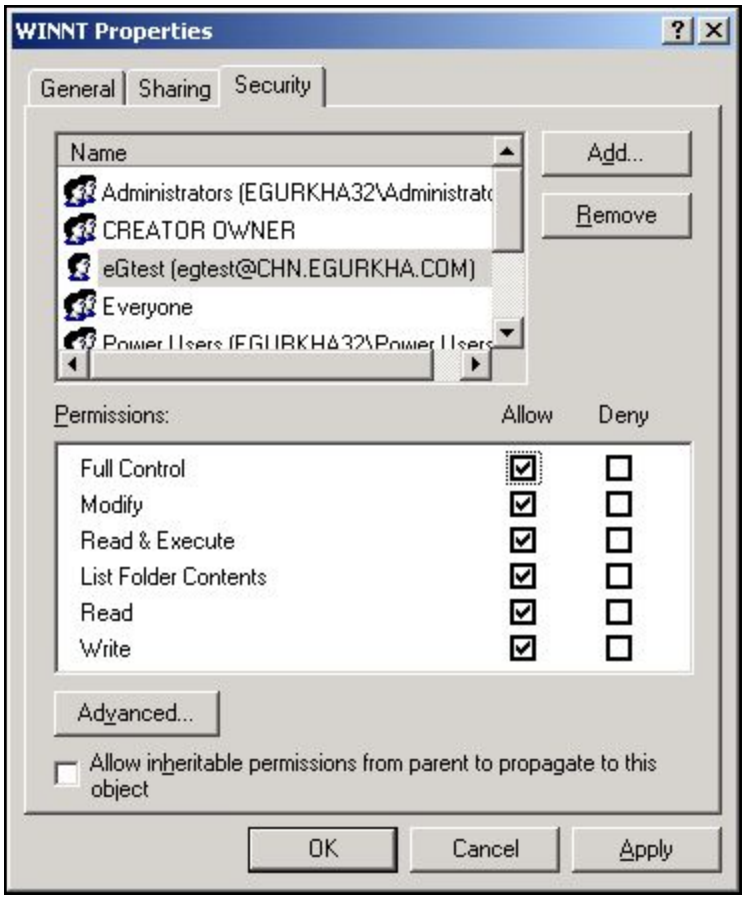


Figure 2.78: The Administrator account in the Security list

13. Finally, click the **Apply** and **OK** buttons in Figure 2.78.

2.9.1.2 Enabling ADMIN$ Share Access on Windows 2008 VMs

To enable the **ADMIN$** share on a Windows 2008 VM, do the following:

1. Open the Windows Explorer on the virtual machine, browse for the corresponding **Windows** directory in the C drive, right-click on it, and select the **Share** option from the shortcut menu.

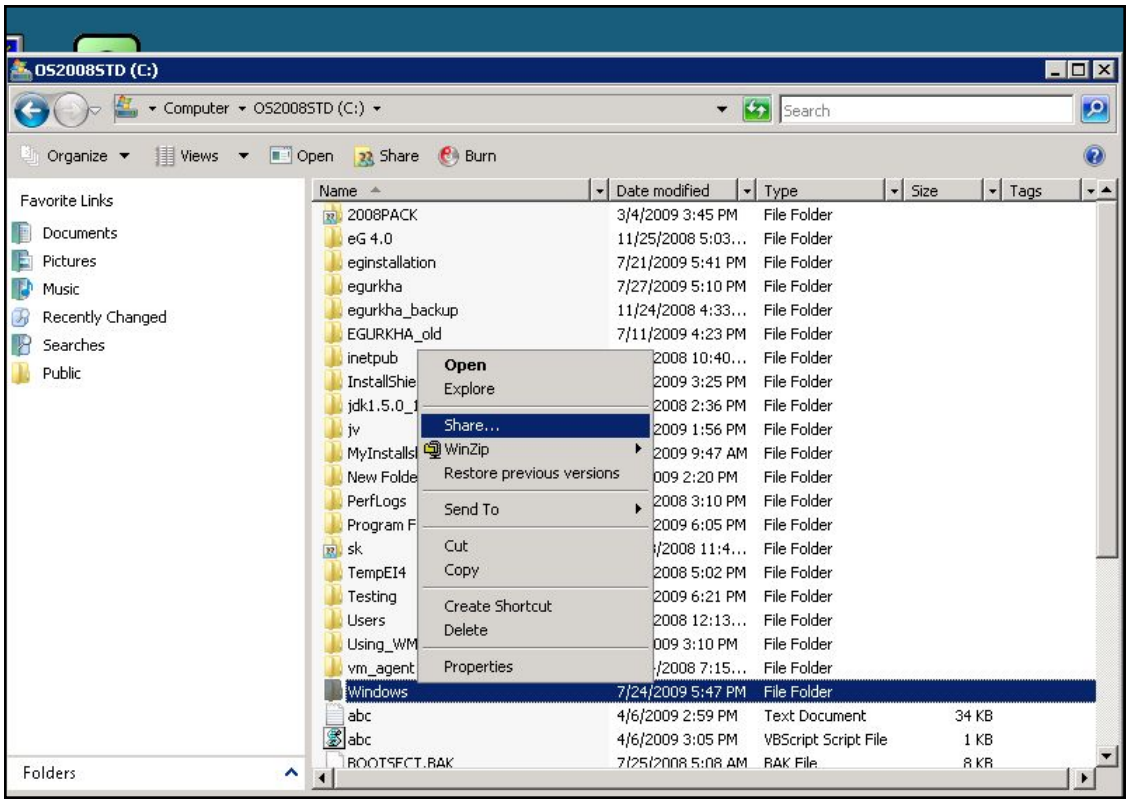


Figure 2.79: Selecting the Share option from the shortcut menu

2. Figure 2.80 will then appear. Click on **Advanced Sharing** in Figure 2.80.

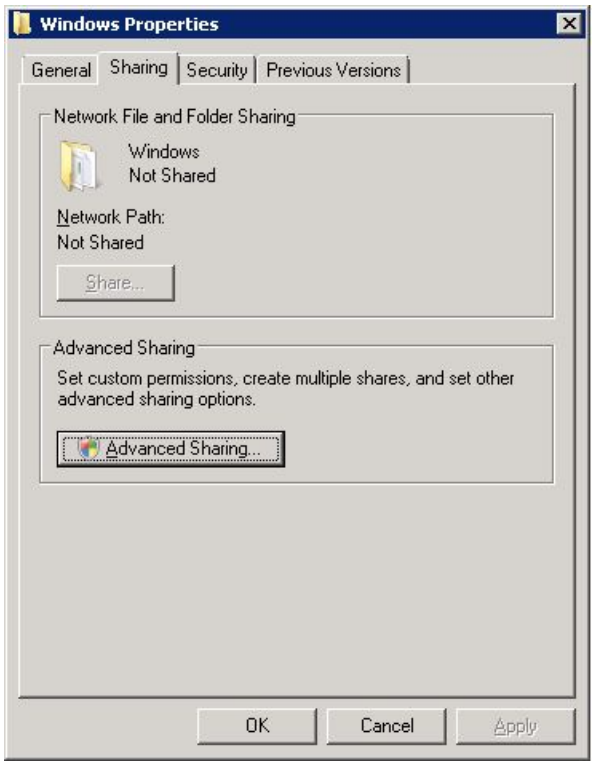


Figure 2.80: Clicking on Advanced Sharing

3. Select the **Share this folder** check box in Figure 2.81 that appears, enter **ADMIN$** against **Share name**, and click on the **Permissions** button in Figure 2.81, to allow only a local/domain administrator to access the folder.

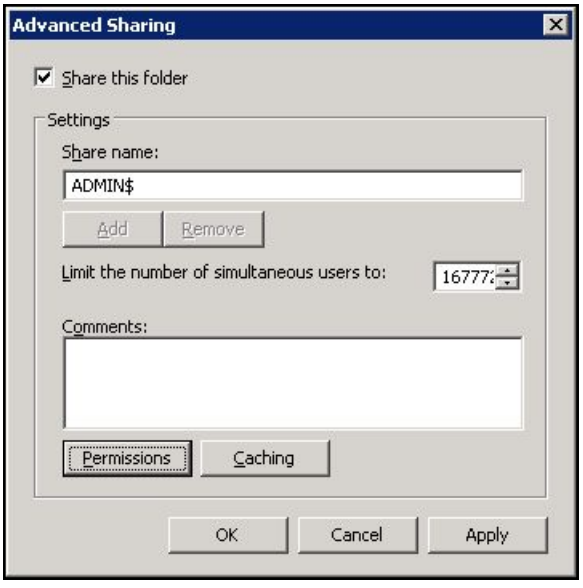


Figure 2.81: Enabling the ADMIN$ share

4. When Figure 2.82 appears, click on the **Add** button therein.

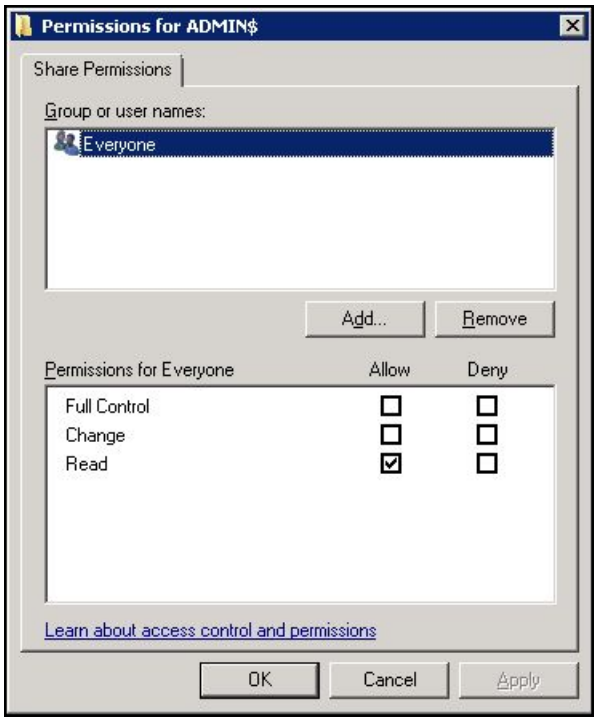


Figure 2.82: Clicking on the Add button

5. To allow a domain administrator to access the folder, first, ensure that a valid domain is specified in the **From this location** box of Figure 2.83. If you want to grant access to a local administrator instead, ensure that the name of the local host is displayed in the **From this location** box. To change this specification, use the **Locations** button in Figure 2.83. Then, enter the name of the local/domain administrator in the **Enter the object names to select** text area, and click the **OK** button.



Figure 2.83: Allowing a domain administrator to access the folder

6. The newly added user will be listed in the **Group or user names** section, as depicted by Figure 2.84. Select this user, and then, check all the three check boxes under **Allow** in the **Permissions for <user>** section in Figure 2.84. Then, click the **Apply** and **OK** buttons therein.

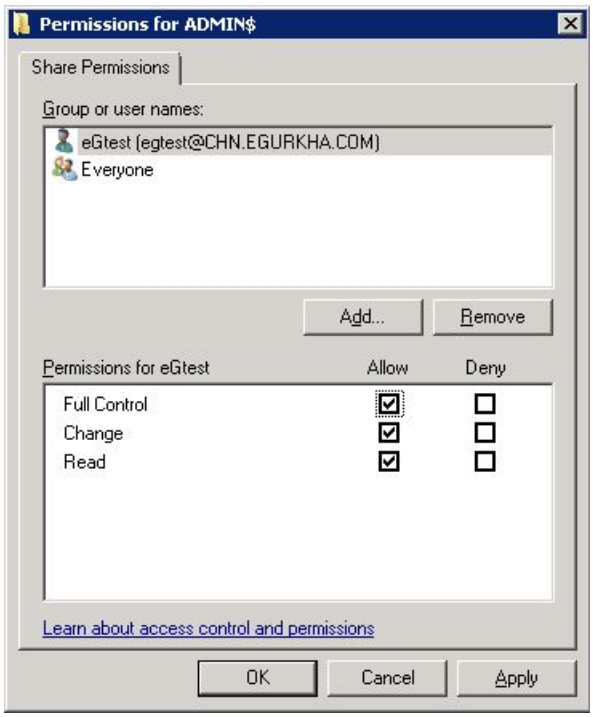


Figure 2.84: Allowing full access to the local/domain administrator

7. When Figure 2.85 appears, click on the **Apply** and **OK** buttons therein to register the changes.

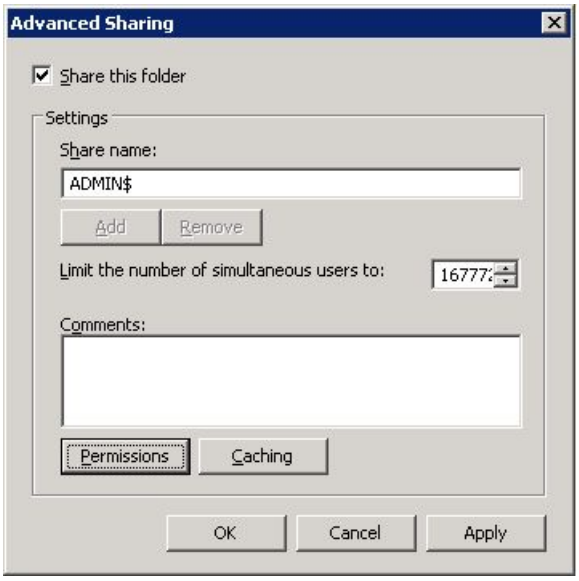


Figure 2.85: Applying the changes

Alternatively, by adding a new entry in the Windows registry, you can quickly enable the **ADMIN$** share. The steps for the same are discussed hereunder:

1. In Run prompt type **regedit** to open registry editor.
2. Browse through the following sub key:

   HKEY_LOCAL_MACHInE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System

3. Create a new entry with the below information
   - Key Name : LocalAccountTokenFilterPolicy
   - Key Type : DWORD (32-bit)
   - Key Value : 1
4. Exit registry editor.

Note:

As with any change to the registry, ensure that the above-mentioned change is also performed with utmost care, so as to avoid problems in the functioning of the operating system.

2.9.2 Configuring Windows Firewalls to Allow File and Print Sharing

In the case of virtual machines operating on Windows XP/Windows 2003/Windows 2008/Windows Vista/Windows 7, the firewall on the guest should be explicitly configured to allow Windows File and Print Sharing services which are required for the eG agent on the ESX host to communicate with the guest operating system.

To achieve this, do the following:

1. Open the Virtual Infrastructure Client console, and from the tree-structure in its left pane, select the guest OS (Windows XP/Windows 2003/Windows Vista/Windows 2008/Windows 7) on which the firewall should be configured (see Figure 2.86).

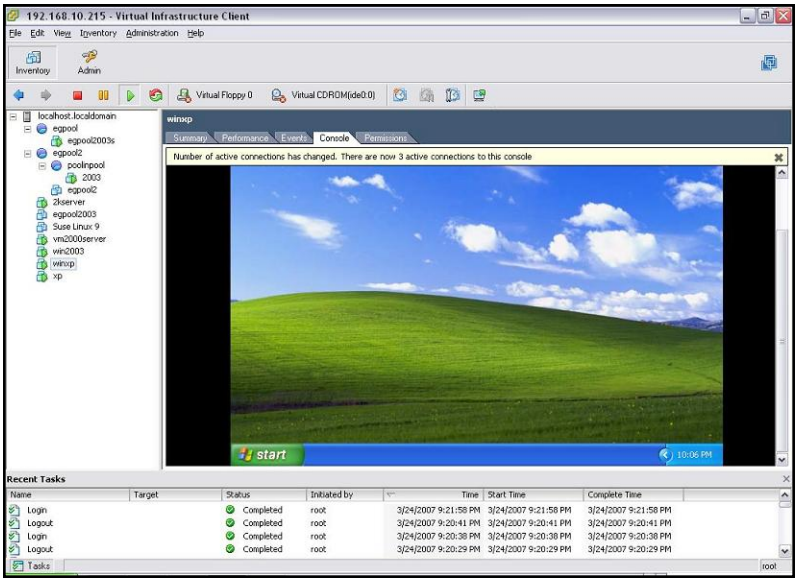


Figure 2.86: Selecting the guest OS

2. Follow the menu sequence: Start -> All Programs -> Control Panel (see Figure 2.87), and then double-click on the **Windows Firewall** option within.

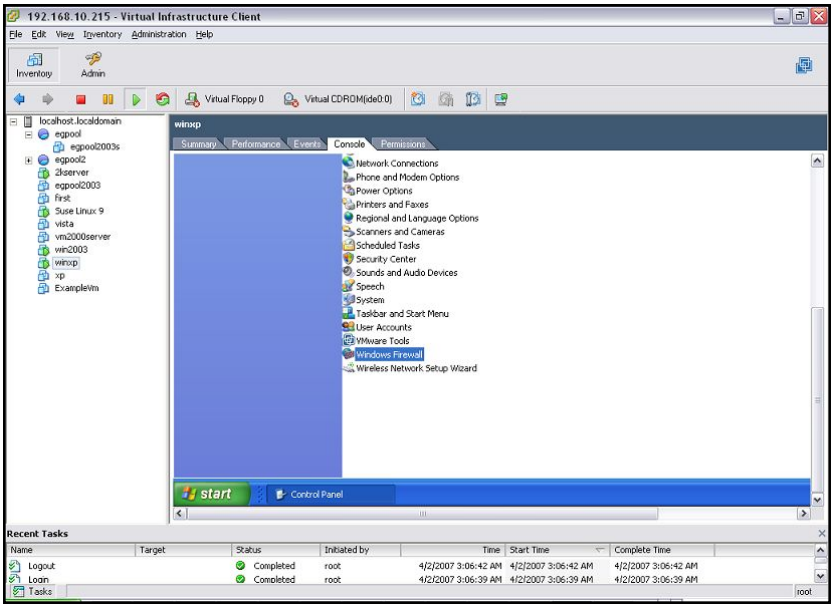


Figure 2.87: Opening the Windows Firewall

3. Figure 2.88 then appears, with the **General** tab selected by default.

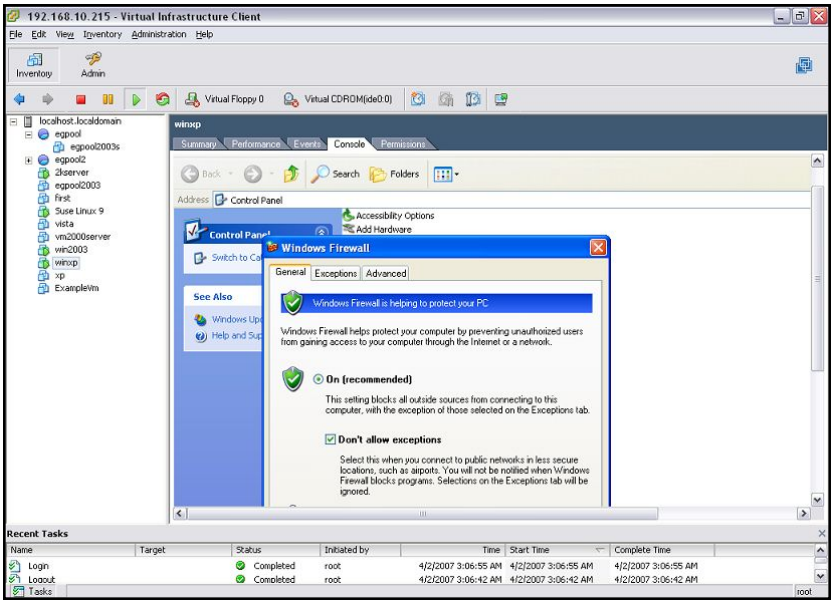


Figure 2.88: The General tab of the Windows Firewall dialog box

4. Deselect the **Don't allow exceptions** check box as indicated by Figure 2.89.

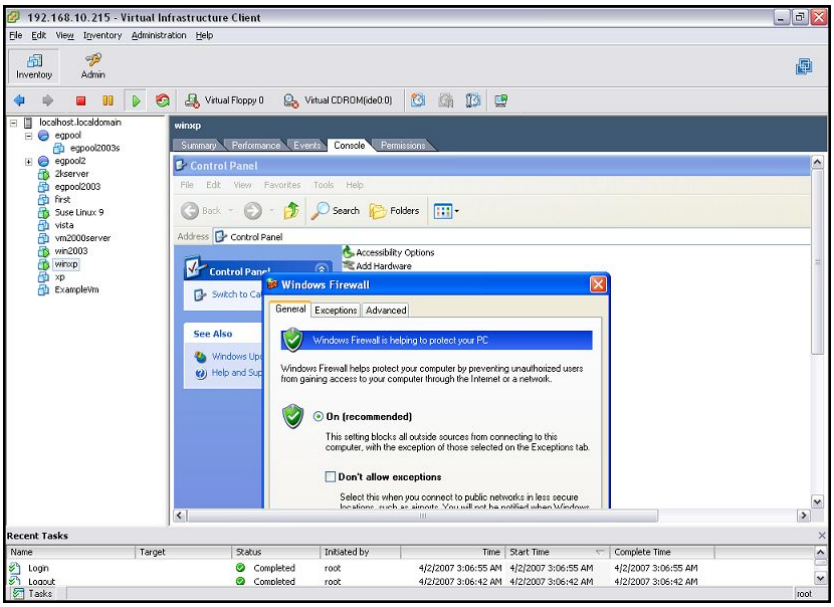


Figure 2.89: Deselecting the 'Don't allow exceptions' check box

5. Next, click on the **Exceptions** tab, and ensure that the **File and Printer Sharing** option is enabled (see Figure 2.90).

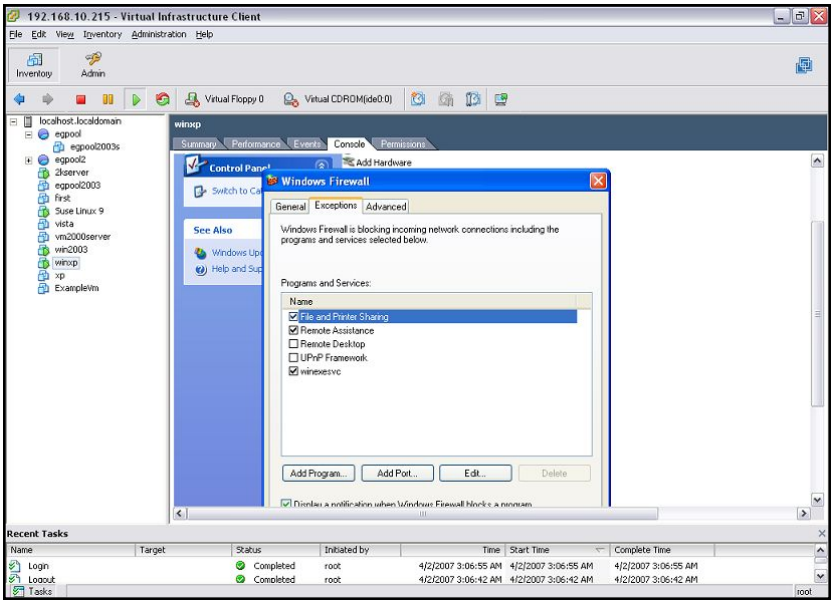


Figure 2.90: Enabling 'File and Printer Sharing'

6. Then, click the **Edit** button in Figure 2.91 to open the ports required for the agent-guest communication. Ensure that at least one of the listed TCP ports are enabled.

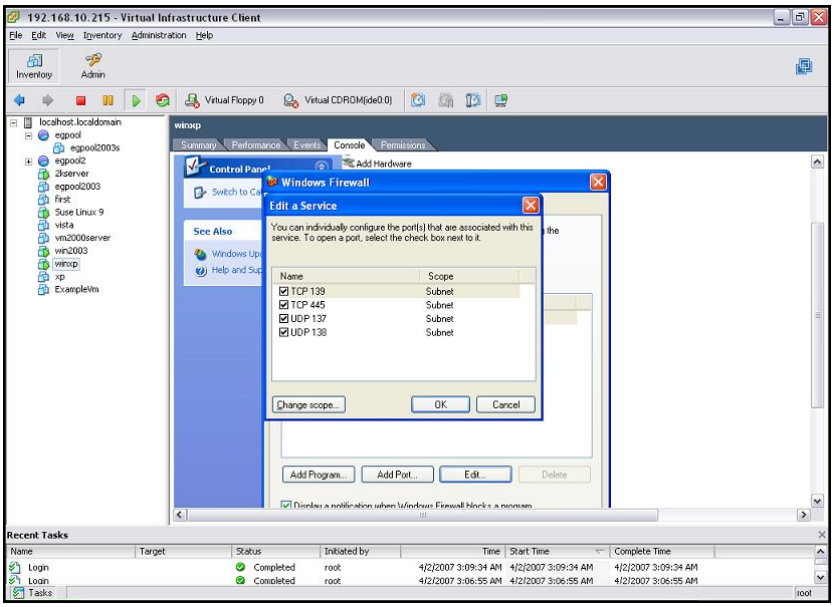


Figure 2.91: Opening ports

7. Finally, click the **OK** button to register the changes.

2.10 Configuring the eG Agent to Monitor NVIDIA Graphics Processing Units (GPUs)

Citrix XenServer employs the NVIDIA GRID Virtual GPU (vGPU) technology to provide exceptional graphics performance for virtual desktops. NVIDIA GRID vGPU enables multiple Virtual Machines (VM) to have simultaneous, direct access to a single physical GPU, using the same NVIDIA graphics drivers that are deployed on non-virtualized Operating Systems. Under the control of NVIDIA's GRID Virtual GPU Manager, which runs in XenServer's Control Domain (dom0), GRID physical GPUs are capable of supporting multiple virtual GPU devices (vGPUs) that can be assigned directly to VMs.

To determine whether/not the vGPUs assigned to a VM are adequate for the graphic processing requirements of the that VM, administrators must understand whether/not memory-intensive I/O operations are performed on that VM and if so, how they impact vGPU usage. To perform this check, administrators can periodically run the **GPU – VM** test on every VM.

For this test to run and report metrics, the **NVWMI** must be installed on every VM.

NVIDIA WMI (NVWMI) is a graphics and display management and control technology that interfaces to Microsoft's Windows Management Instrumentation infrastructure, specific to NVIDIA graphics processing units (GPUs). This allows scripts and programs to be created that configure specific GPU related settings, perform automated tasks, retrieve and display a range of information related to the GPU as well as many other administrative tasks and functions.

The following NVIDIA products support NVWMI:

- NVIDIA Quadro® K600
- Quadro® K6000
- Quadro K5000
- Quadro K4000
- Quadro K2000D
- Quadro K2000
- Quadro FX 5800
- Quadro FX 580
- Quadro FX 570
- Quadro FX 5600
- Quadro FX 4800
- Quadro FX 4700 X2
- Quadro FX 4600
- Quadro FX 380 Low Profile
- Quadro FX 3800
- Quadro FX 380
- Quadro FX 3700
- Quadro FX 370
- Quadro FX 3450
- Quadro FX 1800
- Quadro FX 1700
- Quadro CX
- Quadro 7000
- Quadro 6000
- Quadro 600
- Quadro 5000
- Quadro 410
- Quadro 4000 for Mac
- Quadro 4000
- Quadro 400
- Quadro 2000D
- Quadro 2000
- NVIDIA® NVS® 510

- NVS 450
- NVS 420
- NVS 315
- NVS 310
- NVS 300
- NVS 295
- NVS 290
- Quadro Plex S Series
- Quadro Plex Model IV
- Quadro Plex D Series
- Quadro Plex 7000

NVWMI can be installed in the following three ways:

- When the NVIDIA GPU driver is installed
- Via a standalone install
- Via command line install

Note:

NVWMI is only supported on Windows 7 and later operating systems.

Each of these installation options are detailed in the sub-sections.

2.10.1 Installing NVWMI as Part of the NVIDIA GPU Driver

When installing the full GPU driver, please be aware that NVWMI is not installed by default. To ensure that it is installed, it is necessary to customize the installation and ensure that the NVWMI check box is checked, as shown in Figure 1.19 and Figure 1.20 in the following procedure.

1. Download the full GPU driver from **www.nvidia.com/drivers**.
2. Double click the **setup.exe**.
3. Select **Custom (Advanced Options)** under “Installation options".



Figure 2.92: Custom check box

4. Select the *NVIDIA WMI* checkbox under **Custom installation options**.



Figure 2.93: Selecting the NVIDIA WMI check box

5. Continue to install the driver and follow the on-screen instructions to complete the installation.
6. Repeat steps 1-5 on every VM on which you want to run the **GPU Stats – VM** test.

2.10.2 Installing NVWMI from the Standalone Installer

1. Go to **www.nvidia.com/drivers**.
2. Select the **NVIDIA Enterprise Manager** under **Beta, Older drivers and more**.

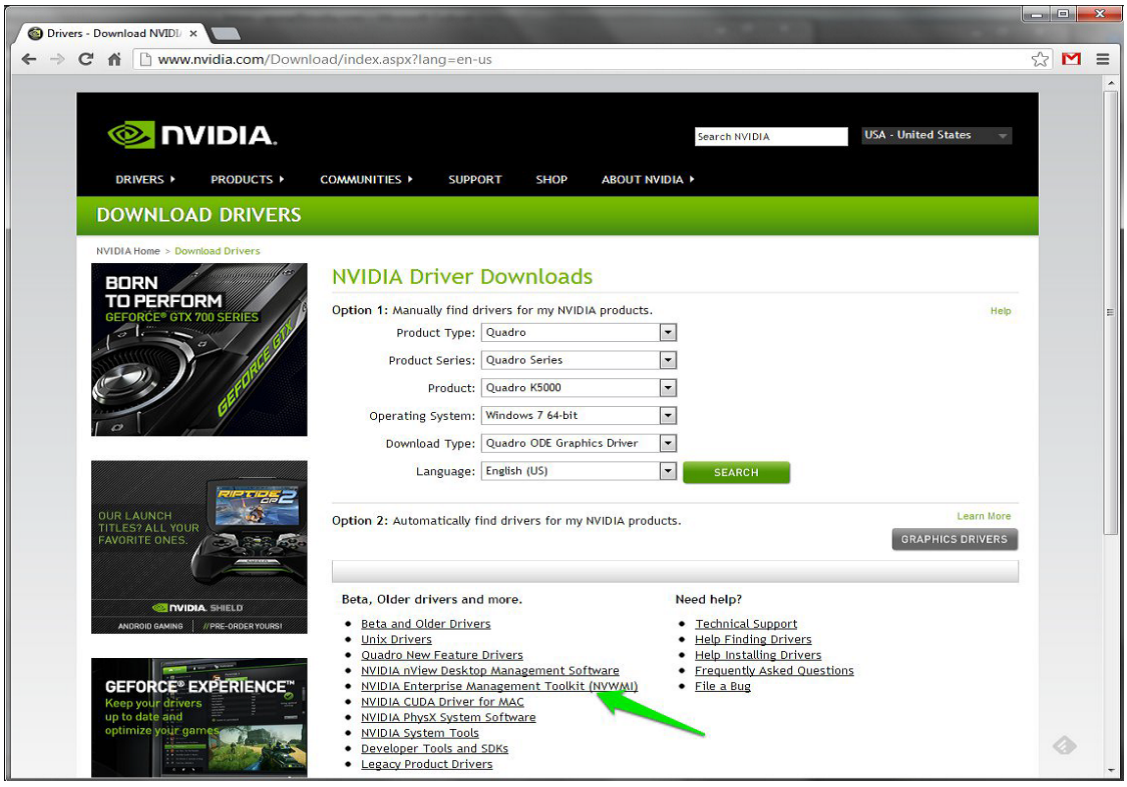


Figure 2.94: NVIDIA Enterprise Management Toolkit (NVWMI)

3. Ensure to verify the minimum driver version is met and download the driver.

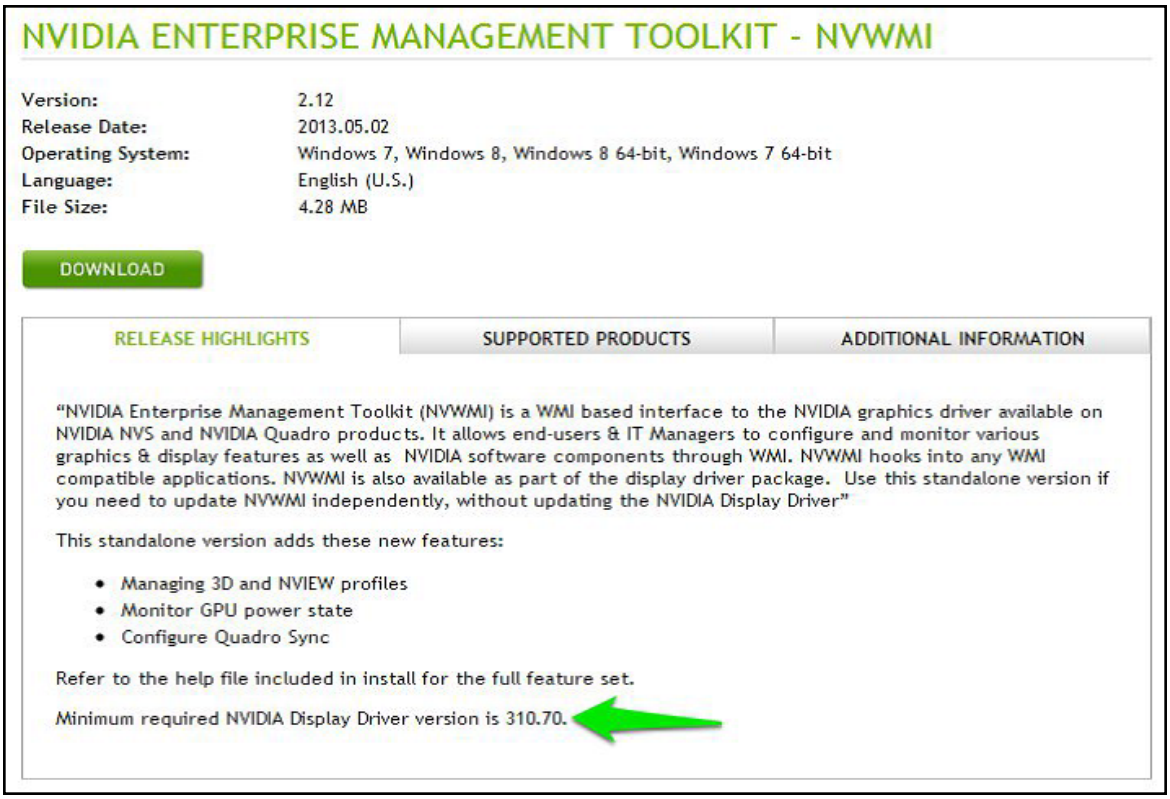


Figure 2.95: NVIDIA WMI minimum driver requirements

4. Continue to install the driver and follow the on-screen instructions to complete the installation.
5. Repeat steps 1-4 on every VM on which you want to run the **GPU Stats – VM** test.

2.10.3 Installing NVWMI via the Command Line

1. Download the driver from the URL **www.nvidia.com/drivers**
2. Using the command line navigate to the extracted directory of the downloaded driver.
3. Issue the following command at the command prompt:

   setup.exe [switches][package]

   Here:

   - **[package]** = Display NVWMI
   - **[switches]** can be one of the following:

Switches	Description
-s	Silent install or uninstall
-k	Force a reboot after install or uninstall
-uninstall	Performs an uninstall instead of an install

To confirm if NVIDIA WMI is already installed, simply look in the list of programs available to change or uninstall in the **Control Panel** under **Programs and Features** heading.

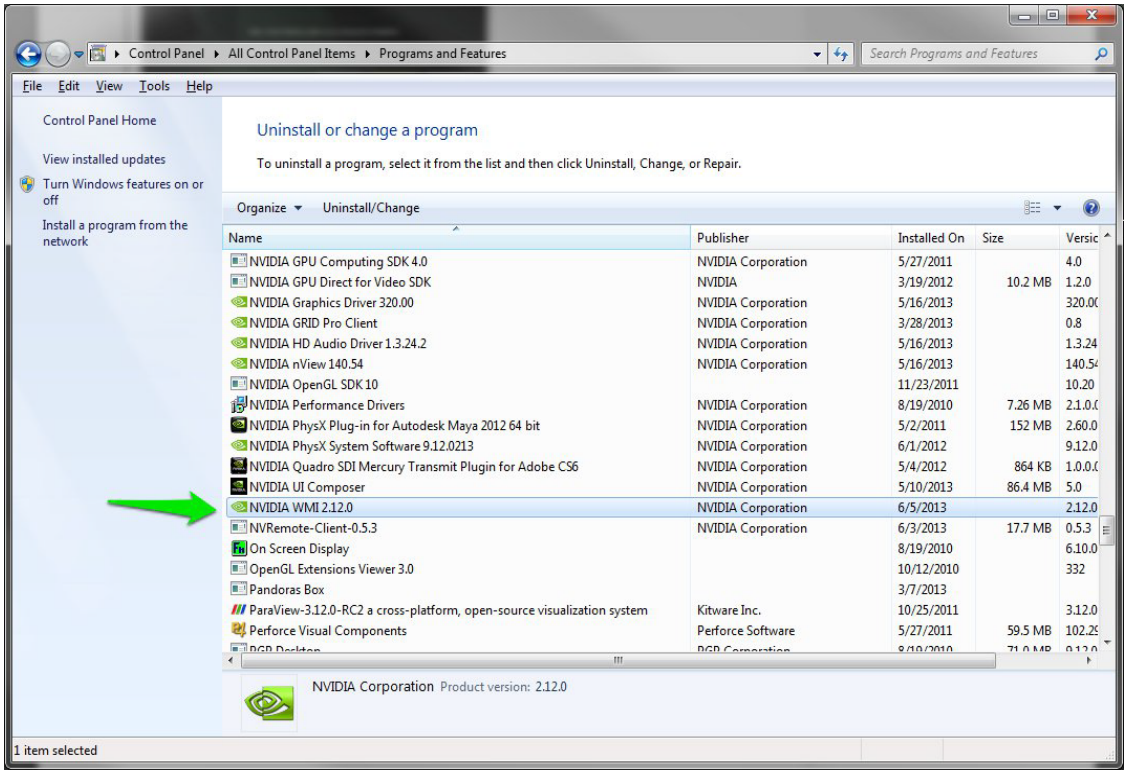


Figure 2.96: Checking the Control Panel for NVIDIA WMI

4. Repeat steps 1-3 on every VM on which you want to run the **GPU Stats – VM** test.

Once the pre-requisites for the "inside" and "outside" views are fulfilled, the eG agent periodically executes tests on the XenServer host and VMs, collects useful resource usage metrics from them, and reports the measurements so collected to the eG manager. The eG manager then consolidates and presents the metrics so received using a layer model representation.

The chapters that follow discuss the layer models that eG Enterprise offers out-of-the-box for the *Citrix XenServer* and *Citrix XenServer - VDI* components, and what each layer reveals.

2.11 Monitoring Models offered by eG Enterprise for VMware Infrastructures

To monitor VMware infrastructures, the eG Enterprise Suite provides two specialized monitoring models - each of these models cater to a unique monitoring requirement of virtualized environments. The *VMware vSphere ESX* server model provides *agentless* insights into the performance of vSphere/ESX servers with VMs hosting server applications. Whereas, the *VMware vSphere VDI* server model on the other hand, focuses on the health of virtual desktop environments - i.e., vSphere/ESX servers with VMs hosting desktop applications.

The chapters that follow will focus on each of these models.

3

Monitoring vSphere/ESX Servers with VMs Hosting Server Applications

As already mentioned, the eG Enterprise Suite offers the *VMware vSphere ESX* model for monitoring of those vSphere/ESX servers that have been configured with VMs on which critical server applications (eg., Oracle, WebLogic, IIS, etc.) have been deployed. Figure 3.1 depicts the *vSphere ESX server* monitoring model.

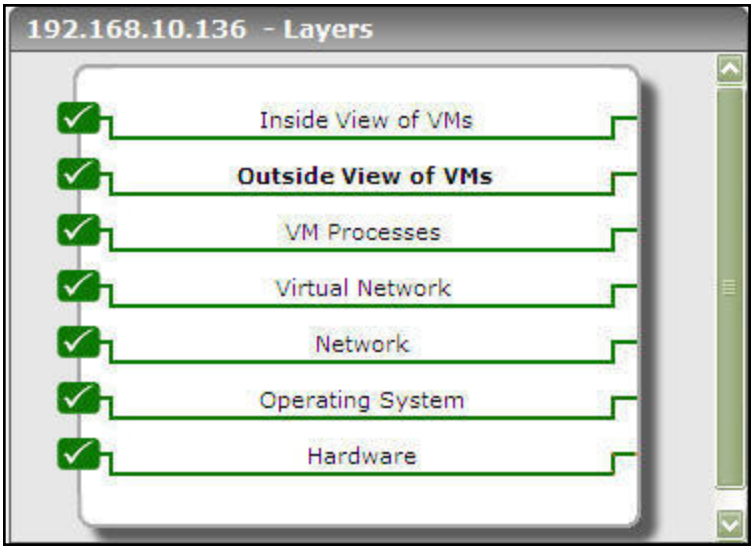


Figure 3.1: The VMware vSphere ESX server layer model

ESX 3, 3.5, ESXi, and vSphere servers can be monitored using this model.

The sections that will follow discuss each layer of Figure 3.1 above.

3.1 The Hardware Layer

The tests mapped to this layer monitor the health of the server hardware, and proactively alert administrators to potential hardware failures.

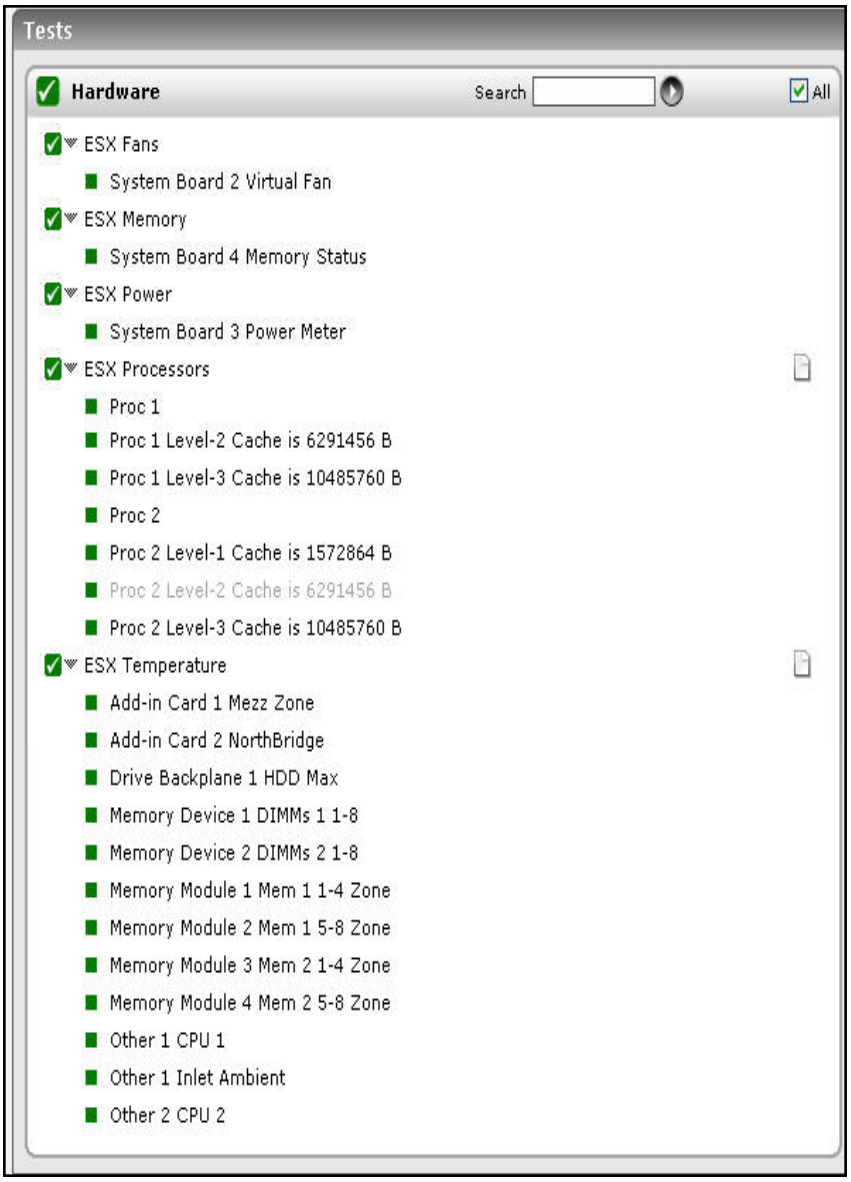


Figure 3.2: The tests mapped to the Hardware layer

3.1.1 Fans - ESX Test

This test reports the current status of the fan and its overall health in terms of the speed with which it operates.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each fan supported by the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** section.

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be

created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Fan status:	Indicates the current status of this fan.		If the status cannot be determined, the Unknown will be displayed here. If the

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>fan is functioning normally, then Normal will be displayed as the value of this measure. If a probable error condition is reported, then the status will be displayed as Warning. If a critical error condition is reported, then the status will be displayed as Warning. If a critical error condition is reported, then the status will be Error.

The numeric values that correspond to each of the states discussed above are listed in the table below:

<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Unknown</td><td>0</td></tr><tr><td>Normal</td><td>1</td></tr><tr><td>Warning</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr></table>
Note:

By default, this measure reports one of the States listed in the table above. The graph of this measure however, represents the status of a fan using the numeric equivalents - ‘0’ to ‘3’.</td></tr>
<tr><td>Fan speed:</td><td>Indicates the speed of this fan in revolutions per minute.</td><td>Rpm</td><td></td></tr>
</table>

3.1.1.1 Adding the Details of a vCenter Server for Guest Discovery

To discover the VMs on an ESX server and to obtain “outside view” metrics using vCenter, all “outside view” tests need to be configured with the IP address of the vCenter server to be used for guest discovery. This IP address needs to be chosen from the **VIRTUAL CENTER** list available for all “outside view” tests. By default, the **VIRTUAL CENTER** list contains the IP address of all the vCenter hosts that have been explicitly added to the eG Enterprise system using the menu sequence: Infrastructure -> Discover. If the IP address of interest to you does not pre-exist in the **VIRTUAL CENTER** list, then you can add the details of the vCenter server, on-the-fly, by selecting the Other option from the list. This will invoke the **ADD THE VCENTER SERVER DETAILS** page (see Figure 3.3), using which a new vCenter server can be added to the eG Enterprise system.

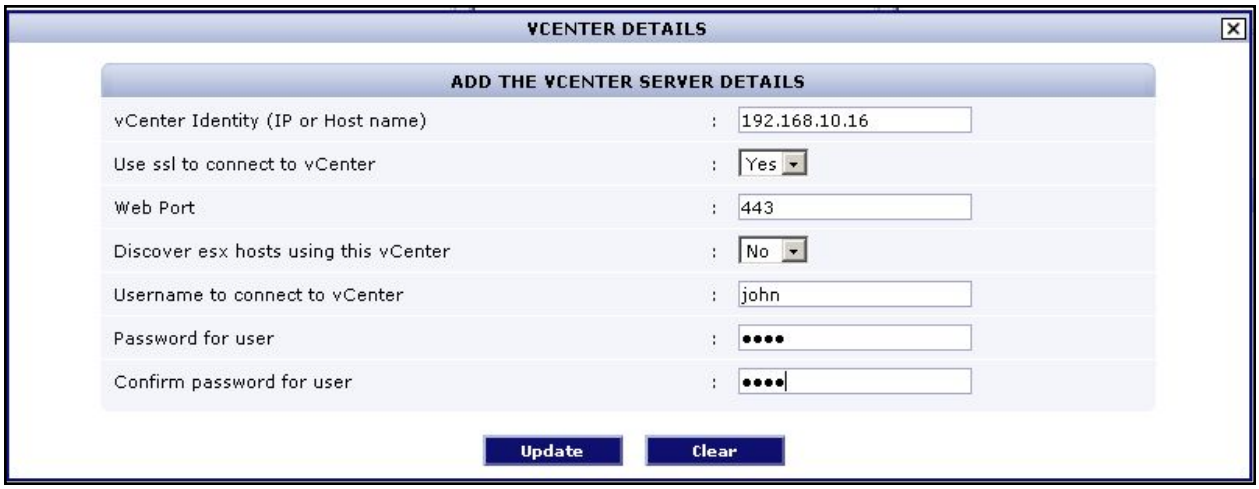


Figure 3.3: Adding the details of a vCenter server

To know how to use this page, follow the instructions provided below.

1. Provide the IP or hostname of the vCenter server being added in the **vCenter Identity (IP or Host name)** text box.
2. Then, indicate whether the eG manager is to connect to vCenter using SSL or not by selecting the **Yes** or **No** option from the **Use SSL to connect to vCenter** list.
3. By default, in most virtualized environments, vCenter listens on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled) only. This implies that while discovering ESX servers using vCenter, eG Enterprise communicates with vCenter via the default ports 80 or 443, depending upon the SSL-enabled status of vCenter. Accordingly, the **Web Port** parameter is set to 443 by default, if the **SSL** flag is set to **Yes**, and displays the default value 80 if the **SSL** flag is set to **No**.
4. In some environments however, the default ports 80 or 443 might not apply. In such a case, in the **Web Port** text box, specify the exact port at which vCenter in your environment listens.
5. If the vCenter being added is to be used for ESX discovery as well, set the **Discover ESX hosts using this vCenter** flag to **Yes**. Otherwise, set this flag to **No**.
6. In order to be able to discover ESX servers using a vCenter server, the eG manager needs to connect to vCenter using the credentials of a valid user to vCenter. Provide the user name and password of such a user in the **Username to connect to vCenter** and **Password for the user** text boxes. This user typically requires **Administrator** or **Virtual Machine Administrator** privileges. However, if you cannot expose the credentials of such a user owing to security constraints, then, you can use the credentials of a user with 'Read-only' privileges to vCenter. If such a user pre-exists, then, provide the name and password of that user in the text boxes mentioned above. Otherwise, assign the 'Read-only' role to a local/domain user to vCenter, and provide the name and password of this user in the **Username to connect to vCenter** and **Password for the user** text boxes. The steps for achieving this are detailed in Section Section **2.2** of this document.
7. Confirm the password of the user by retyping it in the **Confirm password for the user** text box.
8. You can then click the **Update** button to save the vCenter configuration.

3.1.2 Memory Sensors – ESX Test

This test reports the status of the memory sensors on the target ESX server.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each memory sensor supported by the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** of this document.

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** of this document.

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its

contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** of this document.

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** of this document. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware

metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** of this document.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with

each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Memory status:	Indicates the current status of this memory sensor.		If the status cannot be determined, then *Unknown* will be displayed here. If the memory sensor is functioning normally, then *Normal* will be displayed as the value of this measure. If a probable error condition is reported, then the status will be displayed as *Warning*. If a critical error condition is reported, then the status will be *Error*. The numeric values that correspond to each of the states discussed above are listed in the table below: **State** / **Numeric Value** Unknown – 0 Normal – 1 Warning – 2 Error – 3 **Note:** By default, this measure reports one of the States listed in the table above. The graph of this measure however, represents the status of a memory sensor using the numeric equivalents - '0' to '3'.

3.1.3 Power – ESX Test

This test reveals the current status of the power supply units on the ESX server, and also reports the power wattage that is conducted by each unit.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each power supply unit on the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6**.

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for

this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then

map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Power status:</td><td>Indicates the current status of this power supply unit.</td><td></td><td>If the status cannot be determined, then Unknown will be displayed here. If the power supply unit is functioning normally, then Normal will be displayed as the value of this measure. If a probable error condition is reported, then the status will be displayed as Warning. If a critical error condition is reported, then the status will be Error.

The numeric values that correspond to each of the states discussed above are listed in the table below:
<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Unknown</td><td>0</td></tr><tr><td>Normal</td><td>1</td></tr><tr><td>Warning</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr></table>
Note:

By default, this measure reports one of the States listed in the table above. The graph of this measure however, represents the status of a power supply unit using the numeric equivalents - ‘0’ to ‘3’.</td></tr>
<tr><td>Power usage:</td><td>Indicates the wattage currently handled by this power supply unit.</td><td>Watts</td><td></td></tr>
</table>

3.1.4 Processor Sensor - ESX Test

This test helps determine the status of the processors supported by the ESX server.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each processor supported by the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server:** Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in 2.5 of this document.

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6**.

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the reset and refresh SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the Change Settings privilege. For that you need to create a special role on vSphere, assign the Change Settings privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7**.

   - **Monitoring using the vCenter in the target environment:** By default, the eG agent connects to

each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an **Administrator** or **Virtual Machine Administrator** in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** of this document. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the reset and refresh SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.

6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least '*Read-only*' privileges to the target ESX server.

Measurements made by the test

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Processor status:</td><td>Indicates the current status of this processor.</td><td></td><td>If the status cannot be determined, then Unknown will be displayed here. If the processor is functioning in an error-free manner, then Normal will be displayed as the value of this measure. If a probable error condition is reported, then the status will be displayed as Warning. If a critical error condition is reported, then the status will be Error.

The numeric values that correspond to each of the states discussed above are listed in the table below:
<table><tr><th>State</th><th>Value</th></tr><tr><td>Unknown</td><td>0</td></tr><tr><td>Normal</td><td>1</td></tr><tr><td>Warning</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr></table>
Note:

By default, this measure reports one of the States listed in the table above. The graph of this measure however, represents the status of a processor using the numeric equivalents - ‘0’ to ‘3’.</td></tr>
</table>

3.1.5 Temperature - ESX Test

Using this test, you can instantly detect changes in the status of the temperature units on the ESX server, and also track the variations in the temperature readings of the individual units.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each temperature unit on the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be

created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Temperature status:	Indicates the current status of this temperature unit.		If the status cannot be determined, then *Unknown* will be displayed here. If the

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>unit is functioning in an error- free manner, then Normal will be displayed as the value of this measure. If a probable error condition is reported, then the status will be displayed as Warning. If a critical error condition is reported, then the status will be Error.

The numeric values that correspond to each of the states discussed above are listed in the table below:

<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Unknown</td><td>0</td></tr><tr><td>Normal</td><td>1</td></tr><tr><td>Warning</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr></table>
Note:

By default, this measure reports one of the States listed in the table above. The graph of this measure however, represents the status of a temperature unit using the numeric equivalents - ‘0’ to ‘3’.</td></tr>
<tr><td>Current temperature:</td><td>Indicates the current temperature reading of this unit.</td><td>DegC</td><td>Compare the temperature readings registered by the temperature units to accurately identify units that could be experiencing abnormal temperatures. Such units might have to be subjected to closer observation to figure out the root-cause of the anomaly.</td></tr>
</table>

3.1.6 Voltage – ESX Test

This test reports the current status of the voltage probes on the ESX server, and also reveals how much voltage each voltage probe handles.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each voltage probe on the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. To know the procedure for this refer to, Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Voltage status:</td><td>Indicates the current status of this voltage probe.</td><td></td><td>If the status cannot be determined, then Unknown will be displayed here. If the voltage unit is functioning in an error-free manner, then Normal will be displayed as the value of this measure. If a probable error condition is reported, then the status will be displayed as Warning. If a critical error condition is reported, then the status will be Error.

The numeric values that correspond to each of the states discussed above are listed in the table below:
<table><tr><th>State</th><th>Value</th></tr><tr><td>Unknown</td><td>0</td></tr><tr><td>Normal</td><td>1</td></tr><tr><td>Warning</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr></table>
Note:

By default, this measure reports one of the States listed in the table above. The graph of this measure however, represents the status of a voltage probe using the numeric equivalents - ‘0’ to ‘3’.</td></tr>
<tr><td>Current voltage:</td><td>Indicates the current voltage reading of this probe.</td><td>Volts</td><td>Compare the voltage readings registered by the units to accurately identify units that could be experiencing abnormal voltage. Such units might have to be subjected to closer observation to figure out the root-cause of the anomaly.</td></tr>
</table>

3.1.7 Software Components - ESX Test

This test reports the current status of each of the software components operating on a VMware vSphere ESX host. Software components in an abnormal state can thus be isolated.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for vSphere/ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **REPORTMANAGERTIME** - By default, this flag is set to **Yes**, indicating that, by default, the detailed diagnosis of this test, if enabled, will report the shutdown and reboot times of the vSphere/ESX host in the manager's time zone. If this flag is set to **No**, then the shutdown and reboot times are shown in the time zone of the system where the agent is running(i.e., the system being managed for agent-based monitoring, and the system on which the remote agent is running - for agentless monitoring).
5. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4**.

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

6. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
7. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
8. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
9. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section

3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Software component status:	Indicates the current status of this software component.		The table below displays the States that can be reported by this measure, and their numeric equivalents: State: Unknown = 0; Normal = 1; Warning = 2; Error = 3 **Note:** By default, this measure reports one of the States listed in the table above. The graph of this measure however will represent the component status using the numeric equivalents - '0' to '3'.

3.1.8 Slots/Connectors - ESX Test

This test auto-discovers the slots on a vSphere host and reports the state of each slot.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each slot on a vSphere/ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed

2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **REPORTMANAGERTIME** - By default, this flag is set to **Yes**, indicating that, by default, the detailed diagnosis of this test, if enabled, will report the shutdown and reboot times of the vSphere/ESX host in the manager's time zone. If this flag is set to **No**, then the shutdown and reboot times are shown in the time zone of the system where the agent is running(i.e., the system being managed for agent-based monitoring, and the system on which the remote agent is running - for agentless monitoring).
5. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to

each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

6. **CONFIRM PASSWORD** - Confirm the password by retyping it here.

7. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

8. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

9. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Slot/Connector Status:	Indicates the current state of this slot/connector.		The table below displays the States that can be reported by this measure, and their numeric equivalents: State: Value Unknown: 0 Normal: 1 Warning: 2 Error: 3 **Note:** By default, this measure reports one of the States listed in the table above. The graph of this measure however will represent the slot status using the numeric equivalents - '0' to '3'.

3.1.9 System – ESX Test

A bus is a set of physical connections (cables, printed circuits, etc.) which can be shared by multiple hardware components in order to communicate with one another. The motherboard is the principal board that has connectors for attaching devices to the bus. Typically, it contains the CPU, memory and basic controllers for the system. While a bus failure can halt all inter-hardware communication, errors in the motherboard, if not attended to in time, can even crash the system!

To periodically monitor the health of the motherboard and each of the buses supported by a vSphere host, so that you are promptly alerted to potential hardware failures, use the **System - ESX** test.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each voltage probe on the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.

4. **REPORTMANAGERTIME** - By default, this flag is set to **Yes**, indicating that, by default, the detailed diagnosis of this test, if enabled, will report the shutdown and reboot times of the vSphere/ESX host in the manager's time zone. If this flag is set to **No**, then the shutdown and reboot times are shown in the time zone of the system where the agent is running(i.e., the system being managed for agent-based monitoring, and the system on which the remote agent is running - for agentless monitoring).
5. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized

installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

6. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
7. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

8. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

9. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
System status:	Indicates the current status of this system.		If the status cannot be determined, then *Unknown* will be displayed here. If the

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>voltage unit is functioning in an error-free manner, then Normal will be displayed as the value of this measure. If a probable error condition is reported, then the status will be displayed as Warning. If a critical error condition is reported, then the status will be Error.

The numeric values that correspond to each of the states discussed above are listed in the table below:

<table><tr><th>State</th><th>Value</th></tr><tr><td>Unknown</td><td>0</td></tr><tr><td>Normal</td><td>1</td></tr><tr><td>Warning</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr></table>
Note:

By default, this measure reports one of the States listed in the table above. The graph of this measure however, represents the status of a voltage probe using the numeric equivalents - ‘0’ to ‘3’.</td></tr>
</table>

3.1.10 Battery - ESX Test

A battery keeps the vSphere host running, when a power failure occurs. A healthy battery is hence essential to provide users with uninterrupted access to the host and its VMs.

By continuously tracking the battery status, this test proactively alerts you to the real/potential failure of the vSphere host’s battery.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each battery on a vSphere/ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.

4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with

'*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the

vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Battery status:	Indicates the current state of this battery.		The table below displays the States that can be reported by this measure, and their numeric equivalents:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>State</th><th>Value</th></tr>
<tr><td>Unknown</td><td>0</td></tr>
<tr><td>Normal</td><td>1</td></tr>
<tr><td>Warning</td><td>2</td></tr>
<tr><td>Error</td><td>3</td></tr>
</table>
Note:
By default, this measure reports one of the States listed in the table above. The graph of this measure however will represent the battery status using the numeric equivalents - ‘0’ to ‘3’.
</td></tr>
</table>

3.1.11 Chassis - ESX Test

The chassis refers to the rigid framework onto which the motherboard, memory, disk drives, and other equipment are mounted.

This test monitors the chassis of a vSphere/ESX host and reports whether it is healthy or not. An unhealthy chassis can expose all the vital internal components of the system to dust, moisture, and tampering.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the chassis on a vSphere/ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface.

To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4**.

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6**.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7**.

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain

open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7**.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Chassis status:</td><td>Indicates the current state of this chassis.</td><td></td><td>The table below displays the States that can be reported by this measure, and their numeric equivalents:
<table>
<tr><th>State</th><th>Value</th></tr>
<tr><td>Unknown</td><td>0</td></tr>
<tr><td>Normal</td><td>1</td></tr>
<tr><td>Warning</td><td>2</td></tr>
<tr><td>Error</td><td>3</td></tr>
</table>
Note:
By default, this measure reports one of the States listed in the table above. The graph of this measure however will represent the chassis status using the numeric equivalents - '0' to '3'.</td></tr>
</table>

3.1.12 Watchdog – ESX Test

A watchdog timer (or computer operating properly (COP) timer) is a computer hardware or software timer that triggers a system reset or other corrective action if the main program, due to some fault condition, such as a hang, neglects to regularly service the watchdog (writing a "service pulse" to it, also referred to as "kicking the dog", "petting the dog", "feeding the watchdog" or "waking the watchdog"). The intention is to bring the system back from the unresponsive state into normal operation.

By closely monitoring the state of the watchdog, you can easily isolate when your system will not be able to recover from a problem condition - i.e., when the watchdog timer is in an abnormal state.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the watch dog timer on a vSphere/ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX

user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user

(including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Watchdog status:	Indicates the current state of this watchdog.		The table below displays the *States* that can be reported by this measure, and their numeric equivalents: State: Value Unknown: 0 Normal: 1 Warning: 2 Error: 3 **Note:** By default, this measure reports one of the States listed in the table above. The graph of this measure however will represent the watchdog status using the numeric equivalents - '0' to '3'.

3.1.13 Power Usage - ESX Test

This test reports how much power and energy is been consumed by the vSphere/ESX host. This way, you can be promptly alerted when power usage exceeds permissible usage limits; in the process, you can accurately identify the power supply unit that is being over-utilized. Moreover, the detailed diagnosis of this measure reports the energy and power usage of each of the VMs on the target virtual host. You can pit the power usage of physical machines in your environment against the power consumption of VMs to understand the power savings that may accrue to you in a virtualized environment.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each power supply unit on a vSphere/ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to

each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.

6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

- The eG manager license should allow the detailed diagnosis capability
- Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total energy used since last stats reset:	Indicates the amount of energy used by this unit after the reset of the power meter in the host operating system.	Watts	Use the detailed diagnosis of this measure to know how much energy is consumed by each VM executing on the target vSphere/ESX host. You can compare the energy usage of VMs with that of physical machines to better understand the energy savings of virtualization.
Current power usage:	Indicates the current power usage by this power supply unit.	Watts	Comparing the value of this measure across the power supply points can help you instantly identify which point is consuming the maximum power. Use the detailed diagnosis of this measure to know how much power is consumed by each VM executing on the target vSphere/ESX host. You can compare the power usage of VMs with that of physical machines to better understand the power savings of virtualization.
Maximum allowed power usage:	Indicates the maximum allowed power usage for this power supply unit.	Watts	If the current power usage grows close to or exceeds the value of this measure, it is indicative of excessive power usage by a power supply unit. This is a cause for concern and may require further investigation.

3.1.14 Storage Hardware – ESX Test

This test auto-discovers the storage adapters of a vSphere/ESX server, and reports the current status of each adapter. This brings to light those adapters on which errors/warnings may have occurred.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence: Agents -> Tests -> Enable/Disable, pick *VMware vSphere ESX* as the **Component type**, set *Performance* as the **Test type**, choose this test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each storage adapter on a vSphere/ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6**.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section

3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Storage status:	Indicates the current status of this storage adapter.		The table below displays the *States* that can be reported by this measure, and their numeric equivalents: **State** / **Numeric Value** Unknown – 0 Normal – 1 Warning – 2 Error – 3 **Note:** By default, this measure reports one of the States listed in the table above. The graph of this measure however will represent the storage status using the numeric equivalents - '0' to '3'.

3.2 The Operating System Layer

Using the tests mapped to this layer, administrators can determine the percentage of physical CPU/disk/memory resources that are used by the ESX host at the base.

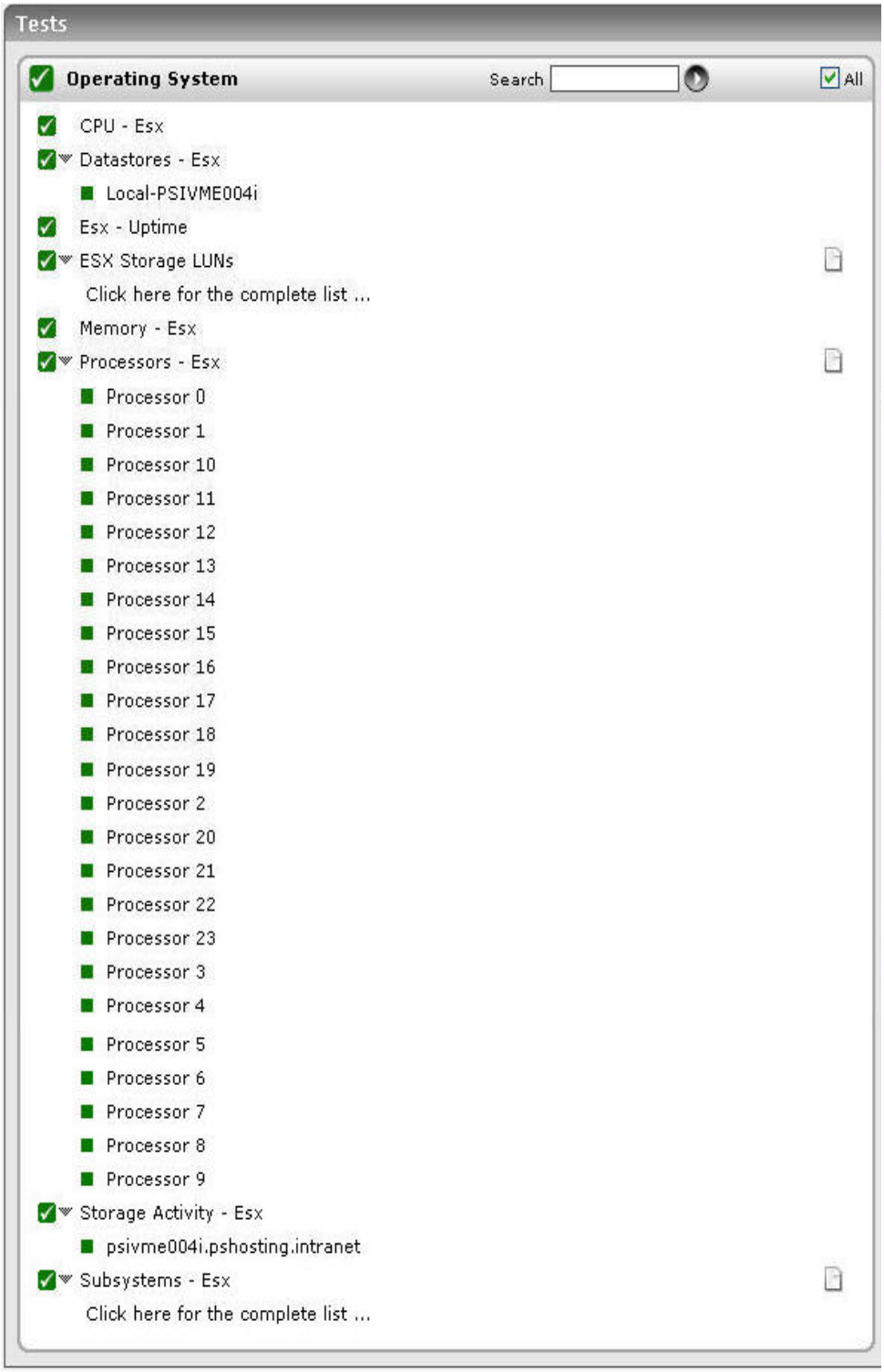


Figure 3.4: The tests mapped to the Operating System layer of the VMware vSphere ESX model

3.2.1 Datastores - ESX Test

This test monitors the space usage of VMFS partitions on a vSphere/ESXi server host.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every VMFS partition/datastore supported by the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change**

Settings privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7**.

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you

need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.
9. **REPORT SHARED DATASTORES** - If the target vSphere/ESX server is being monitored via vCenter, then, you can set this flag to **Yes** to instruct the test to monitor both shared and local datastores, or set

it to **No** so that, only thelocal datastores are monitored.

However, if the target vSphere/ESX server is not monitored via vCenter, then the test will report metrics for all shared and local datastores, regardless of the status of this flag.

10. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormalfrequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Datastore availability:	Indicates whether/not this datastore is available.	Percent	The value 100 indicates that the datastore is available and the value 0 indicates that it is not. The detailed diagnosis of this measure reveals the datastore type.
Number of LUNs for the physical storage:	Indicates the number of LUNs in this datastore.	Number	The detailed diagnosis of this measure reveals the names of the LUNs in the datastore.
Physical disk capacity:	Indicates the total capacity of a datastore. For the Total descriptor, this is the total capacity across all datastores used by the target host.	MB	
Used space:	Indicates the amount of space used in a datastore. For the Total descriptor, this is the total space used	MB	

Measurement	Description	Measurement Unit	Interpretation
	across all datastores.		
Free space:	Indicates the current free space available on every datastore of a host. For the Total descriptor, this is the sum of unused space in all datastores.	MB	Ideally, the value of this measure should be high.
Physical disk space usage:	Indicates the percentage of disk space utilized on every datastore. To know the percentage of total space utilized across all datastores, click on the Total descriptor.	Percent	A value close to 100% can indicate a potential problem situation where applications executing on the host may not be able to write data to the disk partition(s) with very high usage. The detailed diagnosis of this measure reveals the top-10 files consuming maximum space in the datastore. **The detailed diagnosis capability will not be available for the 'Total' descriptor**.
Datastore availability:	Indicates whether/not this VMFS disk partition/datastore is available.	Percent	While the value 0 indicates that the datastore is not available, the value 100 indicates that the datastore is currently available. If a datastore becomes unavailable, then VMs that are currently using that datastore could be rendered inaccessible to users. To know which VMs would be / have been impacted by the non-availability of a datastore, use the detailed diagnosis capability of this measure (if enabled). **This measure is not available for the 'Total' descriptor**.
Virtual disks:	Indicates the total size of the virtual disk files in this datastore. For the Total descriptor, this is the total	MB	The detailed diagnosis of this measure, if enabled, lists the disk files on this datastore and the size of each disk file. **The detailed diagnosis capability**

Measurement	Description	Measurement Unit	Interpretation
	size of all virtual disk files in all datastores.		**will not be available for the 'Total' descriptor**.
Snapshot files:	Indicates the total size of the snapshot files in this datastore. For the Total descriptor, this is the total size of all snapshot files in all datastores.	MB	The detailed diagnosis of this measure, if enabled, lists the snapshot files on this datastore and the size of each snapshot file. **The detailed diagnosis capability will not be available for the 'Total' descriptor.**
Swap files:	Indicates the total size of the swap files in this datastore. For the Total descriptor, this is the total size of all swap files in all datastores.	MB	The detailed diagnosis of this measure, if enabled, lists the swap files on this datastore and the size of each swap file. **The detailed diagnosis capability will not be available for the 'Total' descriptor**.
Other VM files:	Indicates the total size of the other VM files in this datastore. For the Total descriptor, this is the total size of all other VM files in all datastores.	MB	The value of this measure reports the space that is being used by files such as configuration, suspend data, NVRAM, screenshots and others. The detailed diagnosis of this measure, if enabled, lists these files and the size of each such file on this datastore. **The detailed diagnosis capability will not be available for the 'Total' descriptor**. By comparing the values reported by the Disk files, Snapshot files, Swap files, and Other VM files measures, you can accurately identify the type of files that are occupying too much space on a datastore.
Allocated space:	Indicates the amount of physical space provisioned by an administrator for this datastore. For the Total descriptor, this is the total	MB	This measure reports the storage size up to which files on this datastore (or all datastores) can be stacked.

Measurement	Description	Measurement Unit	Interpretation
	physical space provisioned for all datastores used by this host.		
Number of LUNs for the physical storage:	Indicates the number of LUNs through which the host communicates with this datastore.	Number	**This measure is not available for the 'Total' descriptor.**
Average write requests rate:	Indicates the average number of write commands issued per second to this datastore.	Commands/Sec	These measures are good indicators of the load on a datastore. Compare the values of these measures across datastores to accurately identify th overloaded datastores. **Both these measures are not available for the 'Total' descriptor.**
Average read requests rate:	Indicates the average number of read commands issued per second to this datastore.	Commands/Sec	
Write latency:	Indicates the average amount of time taken to write to this datastore.	Secs	Very high values for these measures are indicative of processing bottlenecks in a datastore. To know which datastore is experiencing the maximum latency, compare the values of these measures across datastores. **Both these measures are not available for the 'Total' descriptor.**
Read latency:	Indicates the average amount of time taken to read from this datastore.	Secs	
Storage I/O control normalized latency:	Indicates the device latency that the host observes while communicating with this datastore; this latency is a normalized average across virtual machines	Secs	Storage I/O Control allows cluster-wide storage I/O prioritization, which allows better workload consolidation and helps reduce extra costs associated with over provisioning. When you enable Storage I/O Control on a datastore, ESX/ESXi begins to monitor the device latency that hosts observe when communicating with that datastore. When device latency exceeds a threshold, the datastore is

Measurement	Description	Measurement Unit	Interpretation
			considered to be congested and each virtual machine that accesses that datastore is allocated I/O resources in proportion to their shares. You set shares per virtual machine. You can adjust the number for each based on need. **This measure is not available for the 'Total' descriptor.**
Storage I/O control aggregated IOPS:	Indicates the aggregate number of I/O operations that are occuring on this datastore.	Number	**This measure is not available for the 'Total' descriptor.**
Read rate:	Indicates the rate at which data is read from this datastore.	MB/Sec	**This measure is not available for the 'Total' descriptor.**
Write rate:	Indicates the rate at which data is written to this datastore.	MB/Sec	**This measure is not available for the 'Total' descriptor.**
Storage I/O Control active time:	Indicates whether storage I/O control (SIOC) is active on this datastore or not.		Storage I/O Control (SIOC) is used to provide I/O prioritization of virtual machines running on a group of VMware vSphere hosts that have access to a shared storage pool. It extends the familiar constructs of shares and limits, which exist for CPU and memory, to address storage utilization through a dynamic allocation of I/O capacity across a cluster of vSphere hosts. If SIOC is active, the value of this measure will be 100%. If it is not active, the value of this measure will be 0.
Total iops:	Indicates the total number of	Commands/Sec	Compare the value of this measure

Measurement	Description	Measurement Unit	Interpretation
	read and write commands issued per second to this datastore.		across datastores to identify the busiest datastore in terms of the rate at which read/write commands are issued on it. A consistent increase in the value of this measure for a particular datastore could indicate a potential I/O overload.
Total latency:	Indicates the total amount of time taken to read and write to this datastore.	Secs	Ideally, the value of this measure should be low. A high value could indicate an I/O processing bottleneck. By comparing the value of this measure across datastore, you can quickly identify which datastore is the most latent. Such datastores are ideal candidates for enabling SIOC. Enabling IOC on a datastore triggers the the monitoring of device latency that hosts observe when communicating with that datastore. When latency exceeds a set threshold the feature engages automatically as the datastore is experiencing congestion. Each virtual machine that accesses that datastore is then allocated I/O resources in proportion to their shares.
Throughput:	Indicates the rate at which the data is read and written to this datastore.	MB/Sec	Ideally, the value of this measure should be high. A steady decrease in the value of this measure could indicate that the datastore is experiencing a congestion. You may want to enable SIOC on such datastores, so that when latency exceeds a set threshold the SIOC feature engages automatically and then allocates I/O resources to each VM that acceses that datastore in proportion to their shares.

The detailed diagnosis of the *Datastore availability* measure reveals the type of the datastore.

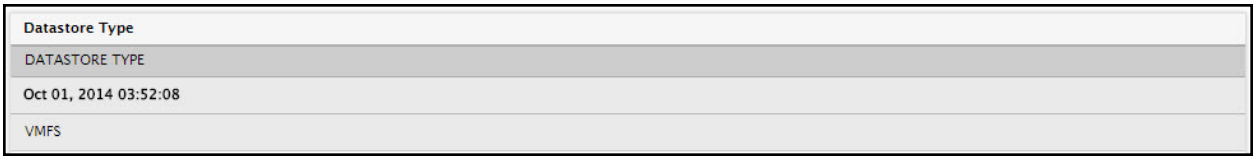


Figure 3.5: The detailed diagnosis of the Datastore availability measure

The detailed diagnosis of the *Number of LUNs for physical storage* measure reveals the names of the LUNs in the datastore.

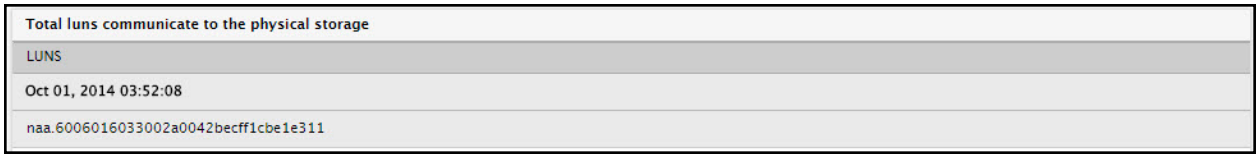


Figure 3.6: The detailed diagnosis of the Number of LUNs for physical storage measure

3.2.2 Subsystems – ESX Test

This test reports real-time CPU utilization statistics pertaining to the sub-systems of an ESX server, and helps identify those sub-systems that consume too much CPU resources.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every sub-system executing on the ESX server host

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the

test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6**.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing

sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically

displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **DESCRIPTOR LIMIT** –This test reports a set of metrics for every sub-system that is auto-discovered by the eG agent. Typically, many levels of sub-systems operate on a vSphere erver. If all these sub-systems are monitored, hundreds of records related to sub-systems will end up being stored in the eG database, leaving the database strained. To reduce the stress on the eG database, you may want to restrict the monitoring scope of this test to a few sub-system levels alone. By default, the value of this parameter is set to 2. This means that the test, by default, will monitor only 2 levels of sub-systems.In the sub-system name, the character '/' separates one level from the other. For instance, the sub-systemssystem and vim, which contain no '/' characters in their name,are said to be sub-systems at the first level. The sub-systems *system/driversand vim/tmp*, which contain a single '/' are sub-systems at the second level. Likewise, *system/drivers/kernel* is a sub-system at the third level, and so on. In the above example, the default value 2 for this parameter will ensure that the test reports metrics for system, vim, *system/drivers,andvim/tmponly*, as these are the sub-systems at the first 2 levels. This test therefore will report system, vim, system/drivers,andvim/tmpas its descriptors. If you want the test to monitor the *system/drivers/kernelsub-system* as well, then you will have to change the value of this parameter to 3.
10. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability

- Both the normal and abnormalfrequencies configured for the detailed diagnosis measures should not be 0.

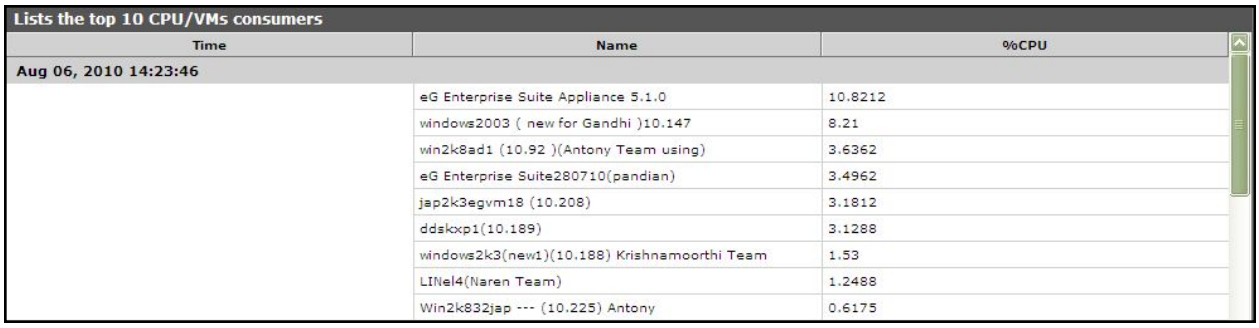
Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Physical CPU usage:	Indicates the percentage of physical CPU used by this subsystem.	Percent	A very high value for this measure indicates excessive CPU utilization by the component. The CPU utilization may be high because a few processes are consuming a lot of CPU, or because there are too many processes contending for a limited resource. If the ESX server being monitored is of version 3 or 3.5, then, an additional **vim/console** descriptor will appear for this test, which would report the CPU usage of the processes executing on the service console. A high value for the vim/console descriptor indicates that one/more service console processes are consuming CPU resources excessively. The VMs descriptor on the other hand, reports the overall physical CPU usage of all VMs executing on the target vSphere/ESX server. In the event of high physical CPU usage by VMs, you can use the detailed diagnosis available for this descriptor to know which VMs are resource-intensive.
Resource memory overhead:	Indicates the total of all overhead metrics for powered- on virtual machines, plus the overhead of running	MB	vSphere/ESXi virtual machines can incur two kinds of memory overhead: • The additional time to access memory within a virtual machine.

Measurement	Description	Measurement Unit	Interpretation
	vSphere services on this sub-system.		• The extra space needed by the ESX/ESXi host for its own code and data structures, beyond the memory allocated to each virtual machine. vSphere/ESXi memory virtualization adds little time overhead to memory accesses. Because the processor's paging hardware uses page tables (shadow page tables for software-based approach or nested page tables for hardware-assisted approach) directly, most memory accesses in the virtual machine can execute without address translation overhead. The memory space overhead has two components: • A fixed, system-wide overhead for the VMkernel and (for ESX only) the service console • Additional overhead for each virtual machine Overhead memory includes space reserved for the virtual machine frame buffer and various virtualization data structures, such as shadow page tables. Overhead memory depends on the number of virtual CPUs and the configured memory for the guest operating system. vSphere/ESXi also provides optimizations such as memory sharing to reduce the amount of physical memory used on the underlying server. These optimizations can save more memory than is taken up by the overhead.
Resource memory swapped:	Indicates the amount of memory that is swapped	MB	

Measurement	Description	Measurement Unit	Interpretation
	out of this sub- system group which includes kernels, VMs and console.		
Resource memory zero:	Indicates the amount of memory that is zeroed out of this sub-system.	MB	The "Memory Zero" amount will fluctuate as memory is over allocated. vSphere will zero out the VM's memory to use with other VMs.
Resource memory mapped:	Indicates the amount of memory that is mapped by this sub-system.	MB	
Resource memory share saved:	Indicates the amount of memory saved from the memory share that is allocated for this sub-system.	MB	
Resource memory shared:	Indicates the current amount of memory that is shared by this sub-system.	MB	vShere/ESX can share common memory pages across VMs. This includes pages from VMs running the same virtual machine OS and applications.
Resource memory touched:	Indicates the amount of memory that has been touched i.e., read from or written to this sub-system in the past X minutes, where X refers to the time period for which this test is executed.	MB	

The **VMs** descriptor on the other hand, reports the overall physical CPU usage of all VMs executing on the target vSphere/ESX server. In the event of high physical CPU usage by VMs, you can use the detailed diagnosis available for this descriptor to know which VMs are resource-intensive.



Lists the top 10 CPU/VMs consumers

Time	Name	%CPU
Aug 06, 2010 14:23:46		
	eG Enterprise Suite Appliance 5.1.0	10.8212
	windows2003 (new for Gandhi)10.147	8.21
	win2k8ad1 (10.92)(Antony Team using)	3.6362
	eG Enterprise Suite280710(pandian)	3.4962
	jap2k3egvm18 (10.208)	3.1812
	ddskxp1(10.189)	3.1288
	windows2k3(new1)(10.188) Krishnamoorthi Team	1.53
	LINel4(Naren Team)	1.2488
	Win2k832jap --- (10.225) Antony	0.6175

Figure 3.7: The detailed diagnosis of the VMs descriptor of the Subsystems - Esx Test

3.2.3 Processors - ESX Test

This test reports the CPU usage of each of the processors supported by the ESX host.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each processor supported by the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in 2.5 .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX

user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the **SSL** flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen

on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag.

In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the Other option from the VIRTUAL CENTER list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least '*Read-only*' privileges to the target ESX server.

9. **REPORT SUMMARY** - By default, this flag is set to **No** indicating that this test will not report the **Summary** of CPU usage across processors, by default. If you want to ensure that the test reports the CPU usage summary too, set this flag to **Yes**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Physical CPU used:	Indicates the percentage of physical CPU used by this	Percent	A very high value for this measure indicates excessive CPU utilization by

Measurement	Description	Measurement Unit	Interpretation
	processor.		the processor. The CPU utilization may be high because a few processes are consuming a lot of CPU, or because there are too many processes contending for a limited resource.

3.2.4 CPU - ESX Test

This test alerts administrators to issues with the overall CPU usage of the ESX host across processors.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called

the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for

the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer toSection **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Overall physical CPU utilization:	Indicates the percentage of physical CPU used by the host.	Percent	A very high value for this measure indicates excessive CPU utilization by the host. The CPU utilization may be high because a few processes are consuming a lot of CPU, or because there are too many processes contending for a limited resource.
Overall physical CPU throttled:	Indicates the percentage of scheduling limit over a past minute.	Percent	
Overall physical CPU used:	Indicates the CPU usage in MHz of the VMware ESX host.	Mhz	
Reserved capacity:	Indicates the total CPU capacity that is reserved by the VMs.	Mhz	
Time in the COSTOP state:	Indicates the percentage of time the virtual host was ready to execute commands but was waiting for the availability of multiple pCPUs as the virtual host was configured to use multiple vCPUs.	Percent	If the virtual host is unresponsive and the value of this measure is high, it may indicate that the vSphere host has limited CPU resources to simultaneously co-schedule all pCPUs. If this value is low, then any performance problems should be

Measurement	Description	Measurement Unit	Interpretation
			attributed to other issues and not to the co-scheduling of the pCPU.

3.2.5 Storage LUNs – ESX Test

LUN is a **Logical Unit Number**. It can be used to refer to an entire physical disk, or a subset of a larger physical disk or disk volume. The physical disk or disk volume could be an entire single disk drive, a partition (subset) of a single disk drive, or disk volume from a RAID controller comprising multiple disk drives aggregated together for larger capacity and redundancy.

The **Storage LUNs – ESX** test reports critical usage statistics pertaining to every LUN on the ESX server host. In the VMware environment, a LUN is typically referred to using its *HBA path*, where HBA stands for the Host Bus Adapter. The *HBA path* on the other hand is expressed using the notation: *vmhba<HBA ID:Target ID:LUN ID>*. For example, if the HBA path is *vmhba0:0:0*, then the first *0* indicates the HBA ID, the second zero represents the Target ID, and the final 0 the LUN ID.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each LUN on the ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the

"Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the

credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text

boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Physical disk reads:	Indicates the rate at which read commands were issued.	Commands/Sec	
Physical disk writes:	Indicates the rate at which write commands were issued.	Commands/Sec	
Issued commands:	Indicates the number of commands issued per second.	Commands/Sec	
Physical disk commands aborted:	Indicates the number of commands aborted per second.	Aborts/Sec	
Data writes to physical disk:	Indicates the rate at which data was written.	MB/Sec	

Measurement	Description	Measurement Unit	Interpretation
Data reads from physical disk:	Indicates the rate at which data was read.	MB/Sec	
Bus resets:	Indicates the number of SCSI bus resets.	Number	The VMs use the SCSI protocol to communicate to disks, even over Fibre Channel to SAN Luns. SCSI Bus Resets are issued to release resources. These SCSI Bus Resets are in effect the SCSI subsystem timing out, commands being canceled, and retrying. This happens when the HBA device is overloaded, or its q-depth is exhausted. The first thing to know is which vmhba controller ©, target/path (T), and LUN (L) experienced these problems. The more VMs sharing a single Lun the more likely that resets will occur. A rule of thumb is no more than 10 VMs sharing a Lun.
Guest read latency:	Indicates the average amount of time taken for a read from the perspective of a guest operating system.	Secs	This is the sum of kernel latency and physical device read latency. High latency is a cause for concern, as it is an indicator of contention for storage resources. If the value of this measure is high, then check the values reported by the Kernel disk read latency and Physical device read latency measures for this storage adapter. Doing so enables you to quickly determine why exactly the guest OS is experiencing latencies while reading from the disk - is it because of latencies in the VM kernel? Or is it owing to a slowdown in the physical device?
Guest write latency:	Indicates the average amount of time taken for a write from the perspective of a guest operating	Secs	This is the sum of kernel latency and physical device write latency. High latency is always a cause for concern, as it is an indicator of contention

Measurement	Description	Measurement Unit	Interpretation
	system.		for storage resources. If the value of this measure is high, then check the values reported by the Kernel disk write latency and Physical device write latency measures for this storage adapter. Doing so enables you to quickly determine why exactly the guest OS is experiencing latencies while writing to this disk - is it because of latencies in the VM kernel? Or is it owing to a slowdown in the physical device?
Disk command latency:	Indicates the average amount of time taken for a command to execute, from the perspective of a guest operating system.	Secs	This is the sum of kernel latency and physical device command latency. High latency is always a cause for concern, as it is an indicator of contention for storage resources. If the value of this measure is high, then check the values reported by the Kernel disk command latency and Physical device command latency measures for this storage adapter. Doing so enables you to quickly determine why exactly the guest OS is experiencing latencies while executing commands on this disk - is it because of latencies in the VM kernel? Or is it owing to a slowdown in the physical device?
Kernel disk read latency:	Indicates the average time spent in the ESX server's VM kernel per read.	Secs	A high value for this measure is a cause for concern. You might hence want to investigate the reasons for the same.
Kernel disk write latency:	Indicates the average time spent in ESX server VM kernel per write.	Secs	
Kernel disk	Indicates the average time	Secs	

Measurement	Description	Measurement Unit	Interpretation
command latency:	spent in the ESX server VM kernel per command.		
Queue read latency:	Indicates the average time spent in the ESX server VM kernel queue per read.	Secs	A high value of this measure indicates that the VMkernel is unable to process queued read requests quickly. If the problem persists, then the queue size could increase considerably. To avoid this, swiftly determine the reasons for a slowdown at the VMkernel, and fix it.
Queue write latency:	Indicates the average time spent in the ESX server VM kernel queue per write.	Secs	A high value of this measure indicates that the VMkernel is unable to process queued write requests quickly. If the problem persists, then the queue size could increase considerably. To avoid this, swiftly determine the reasons for a slowdown at the VMkernel, and fix it.
Queue command latency:	Indicates the average time spent in the ESX server VM kernel queue per command.	Secs	A high value of this measure indicates that the VMkernel is unable to process queued commands quickly. If the problem persists, then the queue size could increase considerably. To avoid this, swiftly determine the reasons for a slowdown at the VMkernel, and fix it.
Physical device read latency:	Indicates the average time taken to complete a read from the physical device.	Secs	Shortage of the physical storage resources can adversely impact the performance of the ESX host and the VMs configured on it. Therefore, if the value of this measure is very high or is steadily increasing, then quickly figure out what is pulling down the performance of the physical disk, and

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td rowspan="3">attend to it.</td></tr>
<tr><td>Physical device write latency:</td><td>Indicates the average time taken to complete a write to the physical device.</td><td>Secs</td></tr>
<tr><td>Physical device command latency:</td><td>Indicates the average time taken to complete a command from the physical device.</td><td>Secs</td></tr>
<tr><td>Status:</td><td>Indicates the current state of this LUN.</td><td></td><td>A LUN can be in any one of the following states:
Ok
error
Off
unknownState
lostCommunication
degraded
quiesced
The numeric values that correspond to each of the states discussed above are listed in the table below:<table><tr><th>State</th><th>Value</th></tr><tr><td>UnknownState</td><td>0</td></tr><tr><td>Ok</td><td>1</td></tr><tr><td>Off</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr><tr><td>lostCommunication</td><td>4</td></tr><tr><td>degraded</td><td>5</td></tr><tr><td>quiesced</td><td>6</td></tr></table>Note:
By default, this measure reports the States listed in the table above to indicate the status of a LUN. The graph of this</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			measure however, represents the status of a LUN using the numeric equivalents - 0 to 6.
	Number of multi paths available to a lun: Indicates the number of storage paths through which the host communicates with this LUN.	Number	To maintain a constant connection between an ESX/ESXi host and its storage, ESX/ESXi supports multipathing. Multipathing is a technique that lets you use more than one physical path that transfers data between the host and an external storage device. In case of a failure of any element in the SAN network, such as an adapter, switch, or cable, ESX/ESXi can switch to another physical path, which does not use the failed component. This process of path switching to avoid failed components is known as path failover. In addition to path failover, multipathing provides load balancing. Load balancing is the process of distributing I/O loads across multiple physical paths. Load balancing reduces or removes potential bottlenecks.
Queue depth:	Indicates the number of outstanding I/O requests to this LUN for which a response has not been received from the LUN.	Number	A low value is desired for this measure. A high value is indicative of a large number of pending requests for the LUN and hints at a potential processing bottleneck on the LUN.
Throughput:	Indicates the rate at which data was read and written to this LUN.	MB/Sec	A high value is desired for this measure. A consistent decrease in the value of this measure can signal a potential slowdown of the LUN.
Capacity:	Indicates the total capacity of this LUN.	MB	

Measurement	Description	Measurement Unit	Interpretation
Capacity used:	Indicates the amount of space used in this LUN.	MB	Compare the value of this measure across LUNs to know which LUN is consuming space excessively.
Capacity reserved:	Indicates the amount of space reserved for this LUN.	MB	
Free capacity:	Indicates the amount of free space currently available for this LUN.	MB	Compare the value of this measure across LUNs to know which LUN is rapidly running out of space.
Capacity usage:	Indicates the percentage of space in this LUN that is in use.	Percent	A low value is desired for this measure. A value close to 100% is a cause for concern, as it indicates a potential space crunch on the LUN. You may want to compare the value of this measure across LUNs to know which LUN is utilizing disk space exc. essively.
Is SSD?	Indicates whether/not this LUN is SSD or non-SSD.		Solid State Disks (SSD) offer a much higher throughput and much lower latency than traditional magnetic hard disks, since they are based on flash memory. vSphere hosts can use locally attached SSDs as a host swap cache, as virtual flash, as a vSAN, or as a regular datastore. If the LUN being monitored is SSD-based, then the value of this measure wwill be Yes. If not, then the value of this measure will be No. The numeric values that correspond to each of the measure values listed above are as follows: **Measure value** / **Numeric Value** Yes / 1 No / 0

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>Note:
By default, this measure reports the Measure Values listed in the table above. In the graph of this measure however, the SSD status of the LUN is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>LUN health:</td><td>Indicates the current health status of this LUN.</td><td></td><td>The values that this measure can report and their corresponding numeric values are discussed in the table above:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Healthy</td><td>0</td></tr><tr><td>Failed</td><td>4</td></tr><tr><td>Offline</td><td>5</td></tr><tr><td>Decommissioned</td><td>6</td></tr><tr><td>Permanent disk failure</td><td>16</td></tr></table>Note:
By default, this measure reports the Measure Values listed in the table above. In the graph of this measure however, the health status of the LUN is represented using the corresponding numeric equivalents only.</td></tr>
</table>

3.2.6 Storage Activity – ESX Test

To determine the overall I/O activity across all the storage adapters/disks on the ESX server host, use the **Storage Activity - Esx** test.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the ESX server that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be

created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Physical disk activity:	Indicates the overall disk I/O activity in the monitored	MB/Sec	

Measurement	Description	Measurement Unit	Interpretation
	ESX server.		
Data writes to physical disk:	Indicates the rate at which data was written.	MB/Sec	
Data reads from physical disk:	Indicates the rate at which data was read.	MB/Sec	

3.2.7 Memory – ESX Test

This test reports statistics pertaining to the machine memory of the VMware vSphere/ESXi server.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every ESX server monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it

will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all

vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total physical memory:	Indicates the total amount of machine memory.	MB	
Used physical memory:	Indicates the amount of physical memory that is in use.	MB	Ideally, the value of this measure should be low.
Free physical memory:	Indicates the amount of machine memory that is free.	MB	A high value is typically desired for this measure.
Percent physical memory free:	Indicates the percentage of machine memory that is available for use.	Percent	A very low value for this measure indicates a shortage of memory resources. If more machine memory is not made available soon, then this could significantly degrade the performance of the guest OS'.
Memory state:	Describes the contention for memory.	Number	This measure takes one of the following values: • 0 - high (lot of memory available) • 1 - soft • 2 - hard

Measurement	Description	Measurement Unit	Interpretation
			• 3 - low (memory is overcommitted) The higher the number, the memory state is more constrained.
Memory granted:	Indicates the amount of memory that the VMkernel has allocated to all virtual machines running on the server.	MB	
Memory unreserved:	Indicates the current amount of unreserved swap space.	MB	
Shared memory:	Indicates the current amount of shared guest operating system memory.	MB	VMware ESX can share common memory pages across VMs. This includes pages from VMs running the same virtual machine OS and applications.
Memory shared common:	Indicates the amount of memory required for a single copy of shared pages in running VMs.	MB	
Balloon memory:	Indicates the total amount of physical memory currently reclaimed by the ESX server using the vmmemctl modules.	MB	The vmmectl driver that is installed on a virtual machine, emulates an increase or decrease inmemory pressure on the guest operating system; this way, it forces the guest OS to place memory pages into its local swap file. This driver differs from the VMware swap file method as it forces the operating system to determine what memory it wishes to page. Once the memory is paged locally on the guest operating system, the free physical pages of memory may be reallocated to other guests. As the ESX hosts sees that memory demand has

Measurement	Description	Measurement Unit	Interpretation
			been reduced, it will instruct vmmemctl to “deflate” the balloon and reduce pressure on the guest OS to page memory. The maximum amount of memory that can be reclaimed from a guest may be configured by modifying the “sched.mem.maxmemctl” advanced option. If the memory reclaimed from a guest (i.e., the value of this measure) is very low, it indicates excessive memory usage by the guest. Under such circumstances, you might want to consider allocating more memory to the guest.
Percent balloon memory:	Indicates the percentage of balloon memory.	Percent	
Current swap used:	Indicates the total amount of swap space used.	MB	This counter reflects the total amount of VMkernel swap usage on the host. In almost any scenario, this counter should be at or close to zero as VMkernel memory swapping is used as a last resort. Significant or consistent memory swapping indicates that ESX host memory is severely overcommitted and that performance degradation is imminent or actively occurring.
Zero memory:	Indicates the amount of memory that is zeroed out.	MB	The “Memory Zero” amount will fluctuate as memory is over allocated. ESX will zero out the VM’s memory to use with other VM’s.
Memory reserved capacity:	Indicates the amount of memory currently utilized to satisfy minimum memory values set for all VMs.	MB	

Measurement	Description	Measurement Unit	Interpretation
Active memory:	Indicates the amount of memory that is actively used.	MB	
Kernel memory:	Indicates the amount of machine memory being used by the ESX Server VMKernel.	MB	
Swap in rate:	Indicates the rate at which memory is swapped from disk into active memory.	Mbps	A high rate of swap ins and swap outs could be indicative of a memory contention on the host.
Swap out rate:	Indicates the rate at which memory is swapped from active memory to disk.	Mbps	
Memory overhead:	Indicates the total of all overhead metrics for powered- on virtual machines, plus the overhead of running vSphere services on the host.	MB	vSphere/ESXi virtual machines can incur two kinds of memory overhead: • The additional time to access memory within a virtual machine. • The extra space needed by the ESX/ESXi host for its own code and data structures, beyond the memory allocated to each virtual machine. vSphere/ESXi memory virtualization adds little time overhead to memory accesses. Because the processor's paging hardware uses page tables (shadow page tables for software-based approach or nested page tables for hardware- assisted approach) directly, most memory accesses in the virtual machine can execute without address translation overhead. The memory space overhead has two components: • A fixed, system-wide overhead for the

Measurement	Description	Measurement Unit	Interpretation
			VMkernel • Additional overhead for each virtual machine Overhead memory includes space reserved for the virtual machine frame buffer and various virtualization data structures, such as shadow page tables. Overhead memory depends on the number of virtual CPUs and the configured memory for the guest operating system. vSphere/ESXi also provides optimizations such as memory sharing to reduce the amount of physical memory used on the underlying server. These optimizations can save more memory than is taken up by the overhead.
Is memory overcommitted?	Indicates whether memory is over-committed or not.		Host memory is over-committed when the total memory space allocated (memory granted) to powered-on VMs, plus host memory overhead, is greater than the amount of total physical memory available to the host. However, note that it is unwise to run a virtual machine with a working set that is larger than the host memory. If this is the case, the hypervisor has to reclaim the virtual machine's active memory through ballooning or hypervisor swapping, which will lead to potentially serious virtual machine performance degradation. If the host memory is overcommited, then this measure will report the value *Yes*. If not, then this measure will report *No*. The numeric values that correspond to the measure values discussed above are listed in the table below:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td><table><tr><th>Numeric Value</th><th>Measure Value</th></tr><tr><td>1</td><td>Yes</td></tr><tr><td>0</td><td>No</td></tr></table>Note:
By default, this measure reports the values Yes or No only to indicate whether the host memory is overcommitted or not. The graph of this measure however, represents the host memory state using the numeric equivalents - 0 or 1.</td></tr>
<tr><td>Memory overcommitted:</td><td>Indicates how much percentage of memory is over-committed.</td><td>Percent</td><td>Host memory is over-committed when the total memory space allocated (memory granted) to powered-on VMs, plus host memory overhead, is greater than the amount of total physical memory available to the host.
A very high value for this measure indicates a shortage of memory resources in the host.</td></tr>
<tr><td>Usage:</td><td>Indicates the memory usage as a percentage of the total configured or available memory.</td><td>Percent</td><td>A consistent increase in this value could be indicative of a slow, but steady erosion of the host physical memory. If the trend continues, it could significantly degrade the performance of the host and the guest OS'.</td></tr>
<tr><td>Service console memory:</td><td>Indicates the amount of memory that is currently reserved for the service console.</td><td>MB</td><td></td></tr>
<tr><td>Machine memory saving:</td><td>Indicates the amount of memory saved due to sharing of memory.</td><td>MB</td><td>The value of this measure is the difference between the value of the Memory shared common measure and the Shared memory measure.</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			The amount of memory saved by memory sharing depends on workload characteristics. A workload of many nearly identical virtual machines might free up more than thirty percent of memory, while a more diverse workload might result in savings of less than five percent of memory.

3.2.8 System - Console Test

The Section **3.2.4** reports the CPU usage of each of the core components of an ESX server. When executed on an ESX server 3/3.5/4.0, this test reports the CPU usage of the service console component as well. However, if one/more processes executing on the console contend for limited CPU resources, the knowledge of the overall CPU usage of the console might alone not suffice; to troubleshoot better, administrators may additionally require the names of those processes that are currently executing on the console and the CPU usage of each. Similarly, while the Section **3.2.7** can alert administrators to a memory contention at the console-level (if an ESX sever 3/3.5/4.0 is monitored), it cannot provide pointers to which processes on the console are contributing to the abnormal memory consumption.

Therefore, to help administrators of ESX 3/3.5/4.0 servers to accurately diagnose the root-cause of CPU or memory contentions experienced by the service console, and to facilitate effective analysis of its impact on other parameters such as run queue length and swap memory usage, eG Enterprise offers the specialized **System - Console** test.

Target of the test : An ESX server 3/3.5/4.0

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **CONSOLE USER** - Provide the name of a valid user to the ESX server being monitored.
4. **CONSOLE PASSWORD** - Specify the password of the **CONSOLE USER** here.
5. **CONFIRM PASSWORD** - Confirm the **CONSOLE PASSWORD** by retyping it here.
6. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run

detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

- The eG manager license should allow the detailed diagnosis capability
- Both the normal and abnormalfrequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Virtual CPU utilization of console:	This measurement indicates the percentage of utilization of the CPU time of the service console.	Percent	A high value could signify a CPU bottleneck. The CPU utilization may be high because a few processes are consuming a lot of CPU, or because there are too many processes contending for a limited resource. Check the currently running processes to see the exact cause of the problem. The detailed diagnosis of this test reveals the top- 10 CPU- intensive processes on the console.
System usage of CPU by console:	Indicates the percentage of CPU time spent for system-level processing.	Percent	An unusually high value indicates a problem and may be due to too many system- level tasks executing simultaneously.
Run queue length of console:	Indicates the instantaneous length of the queue in which threads are waiting for the processor cycle. This length does not include the threads that are currently being executed.	Number	A value consistently greater than 2 indicates that many processes could be simultaneously contending for the processor.
Blocked processes on console:	Indicates the number of processes blocked for I/O, paging, etc.	Number	A high value could indicate an I/O problem on the console (e.g., a slow disk).

Measurement	Description	Measurement Unit	Interpretation
Swap memory of console:	Denotes the committed amount of virtual memory. This corresponds to the space reserved for virtual memory on disk paging file (s).	MB	An unusually high value for the swap usage can indicate a memory bottleneck. Check the memory utilization of individual processes to figure out the process (es) that has (have) maximum memory consumption and look to tune their memory usages and allocations accordingly.
Free memory of console OS:	Indicates the free memory available.	MB	A very low value of free memory is also an indication of high memory utilization on a host. The detailed diagnosis of this measure lists the top 10 processes responsible for maximum memory consumption on the service console.

The detailed diagnosis of the *Virtual CPU utilization of console* measure reveals the top-10 CPU-intensive processes on the console.

Lists the top 10 CPU processes

Time	PID	%CPU	ARGS
Nov 27, 2009 12:27:52			
	2880	5.5	/usr/lib/vmware/bin/vmware-hostd /etc/vmware/hostd/config.xml -u
	3451	0.3	/sbin/sfcbd ++min=0,max=75,swap,group=sfcb -d
	4663	0.1	/opt/vmware/vpxa/vpx/vpxa
	21998	0.1	/opt/egurkha/jre/bin/java -Xrs -Dsun.net.inetaddr.ttl=900 EgMainAgent -manager 192.168.10.39 -port 7077 -dir /opt/egurkha -ssl false -highSecurity false

Figure 3.8: The detailed diagnosis of the Virtual CPU utilization of console measure

The detailed diagnosis of the *Free memory of console OS* measure lists the top 10 processes responsible for maximum memory consumption on the service console.

Lists the top 10 memory processes

Time	PID	%MEM	ARGS
Nov 27, 2009 12:27:52			
	21998	29.6	/opt/egurkha/jre/bin/java -Xrs -Dsun.net.inetaddr.ttl=900 EgMainAgent -manager 192.168.10.39 -port 7077 -dir /opt/egurkha -ssl false -highSecurity false
	2880	19	/usr/lib/vmware/bin/vmware-hostd /etc/vmware/hostd/config.xml -u
	4663	7	/opt/vmware/vpxa/vpx/vpxa
	2780	5	/usr/lib/vmware/webAccess/java/jre1.5.0_15/bin/webAccess -server -Xincgc -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djava.endorsed.dirs=/usr/lib/vmware/webAccess/tomcat/apache-tomcat-6.0.16/common/endorsed -classpath /usr/lib/vmware/webAccess/tomcat/apache-tomcat-6.0.16/bin/bootstrap.jar:/usr/lib/vmware/webAccess/tomcat/apache-tomcat-6.0.16/bin/commons-logging-api.jar -Dcatalina.base=/usr/lib/vmware/webAccess/tomcat/apache-tomcat-6.0.16 -Dcatalina.home=/usr/lib/vmware/webAccess/tomcat/apache-tomcat-6.0.16 -Djava.io.tmpdir=/usr/lib/vmware/webAccess/tomcat/apache-tomcat-6.0.16/temp org.apache.catalina.startup.Bootstrap start
	3441	1.8	/sbin/sfcbd ++min=0,max=75,swap,group=sfcb -d
	3451	1.5	/sbin/sfcbd ++min=0,max=75,swap,group=sfcb -d
	3179	1.1	/sbin/sfcbd ++min=0,max=75,swap,group=sfcb -d
	28422	1	sshd: egurkha [priv]
	3157	0.8	/bin/cimslp ++min=0,max=75,swap,group=sfcb
	3436	0.5	-d. -P1:2:50 -S/etc/opt/vmware/aam/vmware-sites

Figure 3.9: The detailed diagnosis of the Free memory of console OS measure

3.2.9 Disk Space - Console Test

This test auto-discovers the disk partitions supported by the service console, and reports the space usage of each partition; this way, space-hungry partitions can be isolated.

Target of the test : An ESX server 3/3.5/4.0

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **CONSOLE USER** - Provide the name of a valid user to the ESX server being monitored.
4. **CONSOLE PASSWORD** - Specify the password of the **CONSOLE USER** here.
5. **CONFIRM PASSWORD** - Confirm the **CONSOLE PASSWORD** by retyping it here.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total capacity:	Indicates the total capacity of a disk partition; for the	MB	

Measurement	Description	Measurement Unit	Interpretation
	Total descriptor however, this measure indicates the total capacity across all disk partitions.		
Used space:	Indicates the amount of space used in a disk partition; for the Total descriptor however, this measure indicates the sum of space used across all disk partitions of the service console.	MB	Ideally, this value should be low.
Free space:	Indicates the current free space available for each disk partition; for the Total descriptor however, this measure indicates the sum of free space available across all disk partitions.	MB	Ideally, this value should be high.
Percent usage:	Indicates the percentage of space usage on each disk partition; for the Total descriptor however, this measure indicates the percentage of space utilized across all disk partitions.	Percent	A value close to 100% can indicate a potential problem situation where applications executing on the guest may not be able to write data to the disk partition(s) with very high usage.

3.2.10 Disk Activity - Console Test

This test reports statistics pertaining to the input/output utilization of each physical disk on the service console.

Target of the test : An ESX server 3/3.5/4.0

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **CONSOLE USER** - Provide the name of a valid user to the ESX server being monitored.
4. **CONSOLE PASSWORD** - Specify the password of the **CONSOLE USER** here.
5. **CONFIRM PASSWORD** - Confirm the **CONSOLE PASSWORD** by retyping it here.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Percent disk busy:	Indicates the percentage of elapsed time during which the disk is busy processing requests (i.e., reads or writes).	Percent	Comparing the percentage of time that the different disks are busy, an administrator can determine whether load is properly balanced across the different disks.
Disk read time:	Indicates the average time (in seconds) taken by the disk to read data.	Secs	Ideally, the value of this measure should be low.
Disk write time:	Indicates the average time (in seconds) taken by the disk to write data.	Secs	Ideally, the value of this measure should be low.
Reads from disk :	Indicates the number of reads happening on a logical disk per second.	Reads/Sec	A dramatic increase in this value may be indicative of an I/O bottleneck on the guest.
Data reads from disk:	Indicates the rate at which bytes are transferred from the disk during read operations.	Kbytes/Sec	A very high value indicates an I/O bottleneck.
Writes to disk:	Indicates the number of writes happening on a disk per second.	Writes/Sec	A dramatic increase in this value may be indicative of an I/O bottleneck.
Data writes to disk:	Indicates the rate at which	Kbytes/Sec	A very high value indicates an I/O

Measurement	Description	Measurement Unit	Interpretation
	bytes are transferred from the disk during write operations.		bottleneck.
Avg queue for disk:	Indicates the average number of both read and write requests that were queued for this disk during the last measurement period.	Number	A consistent and significant increase in the value of this measure for a particular disk could indicate an I/O overload that could be choking the disk, rendering it unable to process I/O requests quickly.
Current queue for disk:	Indicates the number of requests currently outstanding on the disk.	Number	This measure includes requests in service at the time of the snapshot. This is an instantaneous length, not an average over the time interval. Multi-spindle disk devices can have multiple requests active at one time, but other concurrent requests are awaiting service. This measure might reflect a transitory high or low queue length, but if there is a sustained load on the disk drive, it is likely that this will be consistently high. Requests experience delays proportional to the length of this queue minus the number of spindles on the disks. This difference should average less than two for good performance.
Percent reads from disk:	Indicates the percentage of elapsed time that disk is busy servicing read requests.	Percent	
Percent writes to disk:	Indicates the percentage of elapsed time that disk is busy servicing write requests.	Percent	

3.2.11 ManagementAgent - ESX Test

The management agents synchronize VMware components and let you access the ESXi host through the vSphere Client or vCenter Server. They are installed with the vSphere software. Typically, these agents include hostd, ntpd, sfcbd, slpd, wsman, and vobd.

This test tracks the memory usage of each of management agents and accurately points you to memory-hungry agents.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every management agent on the vSphere host

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** of this document.

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** of this document.

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status

information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** of this document.

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** of this document. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the

test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** of this document.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section

3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Memory used:	Indicates the amount of memory that is utilized by this management agent.	MB	Ideally, the value of this measure should be low. Comparing the memory usage across the management agents will enable you to isolate the most memory-intensive agent.
Memory swap in:	Indicates the rate at which memory is swapped in by this management agent.	Mbps	A high rate of swap ins and swap outs could be indicative of a memory contention on the host. In the event of a contention, you can compare the swap in/out rates across the management agents to identify the memory-hungry agent.
Memory swap out:	Indicates the rate at which memory is swapped out by this management agent.	Mbps	
Memory swap used:	Indicates the total amount of swap space used.	MB	Significant or consistent memory swapping indicates that the vSphere host memory is severely overcommitted and that performance degradation is imminent or actively occurring.

3.2.12 Storage Paths - ESX Test

To maintain a constant connection between an ESX/ESXi host and its storage, ESX/ESXi supports multipathing. Multipathing is a technique that lets you use more than one physical path that transfers data between the host and an external storage device. In case of a failure of any element in the SAN network, such as an adapter, switch, or cable, ESX/ESXi can switch to another physical path, which does not use the failed component. This process of path switching to avoid failed components is known as path failover.

In addition to path failover, multipathing provides load balancing. Load balancing is the process of distributing I/O loads across multiple physical paths. Load balancing reduces or removes potential bottlenecks.

This test serves as a good indicator of the effectiveness of the multipathing technique. Using this test, you can monitor the current state and load on each path to a vSphere host's storage, and can thus quickly and accurately isolate failed/overloaded paths.

This test is disabled by default. To enable this test, select the **ENABLE/DISABLE** option from the **Tests** menu of the **Agents** tile (in the eG admin interface). Then, select *VMware vSphere ESX* as the **Component type** and select this test from the **DISABLED TESTS** list. Finally, click the **<** button to enable the selected test and click the **Update** button.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every storage path

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section

3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average write requests rate:	Indicates the average number of write commands issued per second on this storage path during the collection interval.	Commands/Sec	This is a good indicator of the workload on each storage path. Comparing the value of these measures across paths will reveal the overloaded storage paths, thus shedding light on irregularities (if any) in load balancing.
Average read requests rate:	Indicates the average number of read commands issued per second on this storage path during the collection interval.	Commands/Sec	
Write latency:	Indicates the average amount of time taken for a write on the storage path.	Secs	High latencies are often indicative of a transfer bottleneck on the storage path. Excessive loading of a path could also cause read/write slowdowns. Compare the value of these measures across paths to determine which path is experiencing the maximum latency.
Read latency:	Indicates the average amount of time taken for a read on the storage path.	Secs	
Read rate:	Indicates the rate at which commands are read from the storage path.	MB/Sec	This is a good indicator of the workload on each storage path. Comparing the value of these measures across paths will reveal the overloaded storage paths, thus shedding light on irregularities (if any) in load balancing.
Write rate:	Indicates the rate at which commands are written to the storage path	MB/Sec	

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Average commands issued:</td><td>Indicates the average number of commands issued per second on storage path during the collection interval.</td><td>Commands/Sec</td><td></td></tr>
<tr><td>State:</td><td>Indicates the current state of this storage path.</td><td></td><td>This measure can report any of the following values to indicate the state of a storage path:
Active
Dead
Disabled
Standby
Unknown
The numeric values that correspond to the above- mentioned states are as follows:<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Active</td><td>0</td></tr><tr><td>Dead</td><td>1</td></tr><tr><td>Disabled</td><td>2</td></tr><tr><td>Standby</td><td>3</td></tr><tr><td>Unknown</td><td>4</td></tr></table>Note:
By default, this measure reports the above-mentioned States while indicating the state of the storage path. However, the graph of this measure will represent path states using the corresponding numeric equivalents mentioned in the table above.</td></tr>
<tr><td>Total latency:</td><td>Indicates the total amount of time taken to read from and write into this storage</td><td>Secs</td><td>A high value or a consistent rise in the value of this measure is a cause for concern, as it indicates a</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
	path from the perspective of guest operating system.		current/potential I/O processing bottleneck with the path.
Throughput:	Indicates the rate at which the data is read from or written into this storage path.	MB/Sec	Ideally, the value of this measure should be high. A sudden or steady drop in this value could indicate a gradual deterioration in the performance of the storage path, probably owing to a processing bottleneck.

3.2.13 Storage Adapters - ESX Test

Using this test, you can monitor the current state and load on each storage adapter supported by the vSphere host, and can thus quickly and accurately isolate failed/overloaded adapters.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every storage adapter

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6**.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to

connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average write requests rate:	Indicates the average number of write commands issued per second on this storage adapter during the collection interval.	Commands/Sec	This is a good indicator of the workload on each storage adapter. Comparing the value of these measures across adapters will reveal the overloaded storage adapter. Excessive loading on a storage path can cause adapter overloads. Using the detailed diagnosis of these measures, you can compare the rate at which read requests and write requests were issued to all the storage paths via which the vSphere host is communicating with the adapter. Overloaded storage paths can thus be isolated.
Average read requests rate:	Indicates the average number of read commands issued per second on this storage adapter during the collection interval.	Commands/Sec	
Write latency:	Indicates the average amount of time taken for a write on this storage adapter.	Secs	High latencies are often indicative of a processing bottleneck on the storage adapter. Excessive loading of the adapter, or of the path using which the host communicates with the adapter could also cause read/write slowdowns.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td rowspan="2">Compare the value of these measures across adapters to determine which adapter is experiencing the maximum latency.</td></tr>
<tr><td>Read latency:</td><td>Indicates the average amount of time taken for a read on this storage adapter.</td><td>Secs</td></tr>
<tr><td>Read rate:</td><td>Indicates the rate at which commands are read from this storage adapter.</td><td>MB/Sec</td><td rowspan="3">This is a good indicator of the workload on each storage adapter. Comparing the value of these measures across adapter will reveal the overloaded storage adapter, thus shedding light on irregularities (if any) in load balancing.</td></tr>
<tr><td>Write rate:</td><td>Indicates the rate at which commands are written to this storage adapter.</td><td>MB/Sec</td></tr>
<tr><td>Average commands issued:</td><td>Indicates the average number of commands issued per second on this storage adapter during the collection interval.</td><td>Commands/Sec</td></tr>
<tr><td>Status:</td><td>Indicates the current state of this storage adapter.</td><td></td><td>This measure can report any of the following values to indicate the state of a storage adapter:
Online
Offline
Fault
Unknown
The numeric values that correspond to the above-mentioned states are as follows:<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Online</td><td>0</td></tr><tr><td>Offline</td><td>1</td></tr><tr><td>Fault</td><td>2</td></tr><tr><td>Unknown</td><td>3</td></tr></table></td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			Note: By default, this measure reports the above- mentioned States while indicating the state of the storage adapter. However, the graph of this measure will represent adapter states using the corresponding numeric equivalents mentioned in the table above.
Number of LUNs mapped to the storage adapter:	Indicates the number of LUNs that are communicating with this storage adapter.	Number	
Number of targets for the storage adapter:	Indicates the number of targets that are communicating with this storage adapter.	Number	
Number of paths to the storage adapter:	Indicates the number of paths using which the vSphere host is communicating with this storage adapter.	Number	
Total latency:	Indicates the total amount of time taken to read from and write into this storage adapter from the perspective of guest operating system.	Secs	A high value or a consistent rise in the value of this measure is a cause for concern, as it indicates a current/potential I/O processing bottleneck with the storage adapter.
Throughput:	Indicates the rate at which the data is read from or written into the storage adapter.	MB/Sec	Ideally, the value of this measure should be high. A sudden or steady drop in this value could indicate a gradual deterioration in the performance of the storage adapter, probably owing to a processing bottleneck.

3.2.14 Uptime - ESX Test

In most virtualized environments, it is essential to monitor the uptime of critical vSphere/ESX servers in the infrastructure. By tracking the uptime of each of the servers, administrators can determine what percentage of time a server has been up. Comparing this value with service level targets, administrators can determine the most trouble-prone areas of the infrastructure.

In some environments, administrators may schedule periodic reboots of their servers. By knowing that a specific server has been up for an unusually long time, an administrator may come to know that the scheduled reboot task is not working on a server.

The **Uptime – ESX** test included in the eG agent monitors the uptime of critical vSphere/ESX servers in a virtualized infrastructure.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for vSphere/ESX server host that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **REPORTMANAGERTIME** - By default, this flag is set to **Yes**, indicating that, by default, the detailed diagnosis of this test, if enabled, will report the shutdown and reboot times of the vSphere/ESX host in the manager's time zone. If this flag is set to **No**, then the shutdown and reboot times are shown in the time zone of the system where the agent is running(i.e., the system being managed for agent-based monitoring, and the system on which the remote agent is running - for agentless monitoring).
5. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30

mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have

been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

6. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
7. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
8. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
9. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server

via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

10. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormalfrequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Has the ESX server been rebooted?:	Indicates whether the server has been rebooted during the last measurement period or not.	Boolean	If this measure shows 1, it means that the server was rebooted during the last measurement period. By checking the time periods when this metric changes from 0 to 1, an administrator can determine the times when this server was rebooted. The detailed diagnosis of this measure, if enabled, will provide you with the details of the last reboot of the ESX host. Such details will include the

Measurement	Description	Measurement Unit	Interpretation
			shutdown date/time, reboot date/time, the shutdown duration (in minutes), and whether the host has been configured for maintenance or not.
Uptime of the ESX server during the last measure period:	Indicates the time period that the system has been up since the last time this test ran. .	Secs	If the server has not been rebooted during the last measurement period and the agent has been running continuously, this value will be equal to the measurement period. If the server was rebooted during the last measurement period, this value will be less than the measurement period of the test. For example, if the measurement period is 300 secs, and if the server was rebooted 120 secs back, this metric will report a value of 120 seconds. The accuracy of this metric is dependent on the measurement period - the smaller the measurement period, greater the accuracy.
Total uptime of the ESX server:	Indicates the total time that the server has been up since its last reboot.		This measure displays the number of years, months, days, hours, minutes and seconds since the last reboot. Administrators may wish to be alerted if a server has been running without a reboot for a very long period. Setting a threshold for this metric allows administrators to determine such conditions.

3.2.15 Grid GPUs – ESX Test

GPU-accelerated computing is the use of a graphics processing unit (GPU) together with a CPU to accelerate scientific, analytics, engineering, consumer, and enterprise applications. GPU-accelerated computing enhances application performance by offloading compute-intensive portions of the application to the GPU, while the remainder of the code still runs on the CPU. Architecturally, while a CPU has only few cores and handles few hundred threads at a time, a GPU is composed of hundreds of cores that can handle thousands of threads simultaneously and render a flawless rich graphics experience.

Now, imagine if you could access your GPU-accelerated applications, even those requiring intensive graphics power, anywhere on any device. **NVIDIA GRID** makes this possible. With NVIDIA GRID, a virtualized GPU designed specifically for virtualized server environments, data center managers can bring true PC graphics-rich experiences to users.

The NVIDIA GRID GPUs will be hosted in enterprise data centers and allow users to run virtual desktops or virtual applications on multiple devices connected to the internet and across multiple operating systems, including PCs, notebooks, tablets and even smartphones. Users can utilize their online-connected devices to enjoy the GPU power remotely.

In VDI/virtualized server environments, the NVIDIA GRID delivers GPU resources to virtual desktops/VMs. This way, graphics can be rendered on a virtual machine's (VM's) host server rather than on a physical end-point device. This technology now makes it possible to use virtual desktop technology to support users accessing graphics intensive workloads. There are two modes of making GPU resources available to virtual desktops:

- **Dedicated GPU or GPU Pass-through Technology:** NVIDIA GPU pass-through technology lets you create a virtual workstation that gives users all the benefits of a dedicated graphics processor at their desk. By directly connecting a dedicated GPU to a virtual machine through the hypervisor, you can now allocate the full GPU and graphics memory capability to a single virtual machine without any resource compromise.

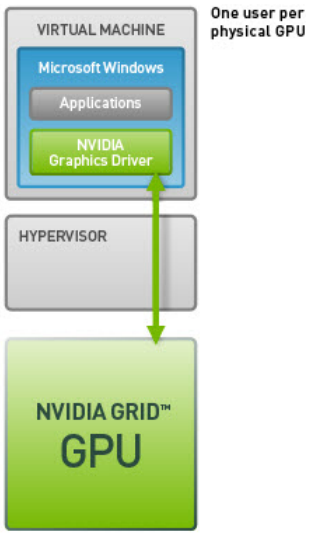


Figure 3.10: Dedicated GPU Technology

- **Shared GPU or Virtual GPU (vGPU) Technology:** GRID vGPU is the industry's most advanced technology for sharing true GPU hardware acceleration between multiple virtual desktops—without compromising the graphics experience. With GRID vGPU technology, the graphics commands of each virtual machine are passed directly to the GPU, without translation by the hypervisor. This allows the GPU hardware to be time-sliced to deliver improved shared virtualized graphics performance. The GRID vGPU manager allows for management of user profiles. IT managers can assign the optimal amount of graphics memory and deliver a customized graphics profile to meet the specific needs of each user.

Every virtual desktop has dedicated graphics memory, just like they would at their desk, so they always have the resources they need to launch and run their applications.

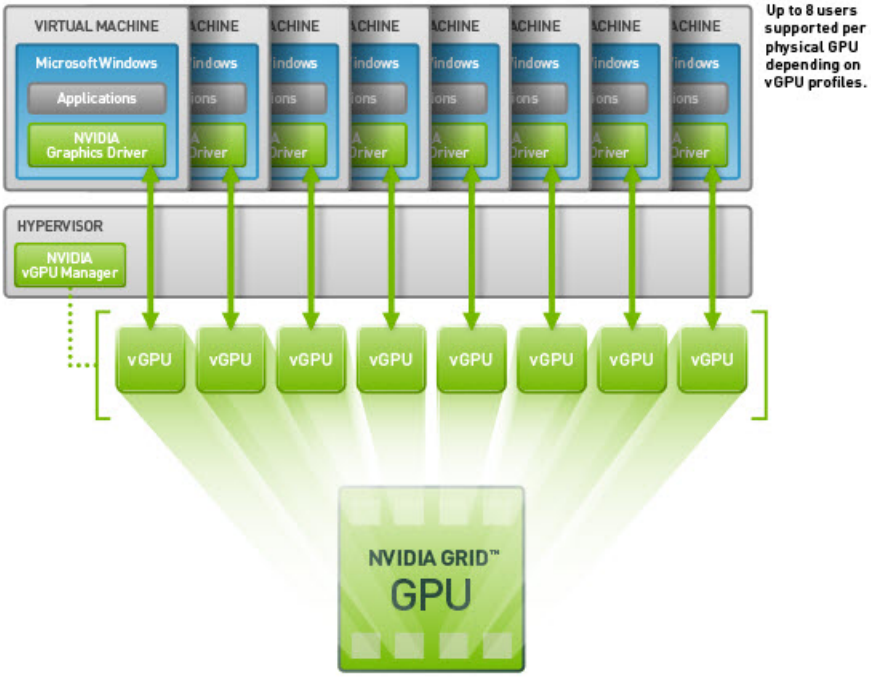


Figure 3.11: Shared vGPU Technology

In GPU-enabled VMware vSphere environments, if users to VMs/virtual desktops complain of slowness when accessing graphic applications, administrators must be able to instantly figure out what is causing the slowness – is it because adequate GPU resources are not available to the host? Or is it because of excessive utilization of GPU memory and processing resources by a few VMs/virtual desktops on the host? Accurate answers to these questions can help administrators determine whether/not:

- The host is sized with sufficient GPU resources;
- The GPUs are configured with enough graphics memory;
- The GPU technology in use – i.e., the GPU Pass-through technology or the Shared vGPU technology – is ideal for the graphics processing requirements of the environment;

Measures to right-size the host and fine-tune its GPU configuration can be initiated based on the results of this analysis. This is exactly what the **Grid GPUs - ESX** test helps administrators achieve!

This test supports GPU monitoring for NVIDIA K1 and K2 Grids installed on a VMware vSpere server. Using the test, administrators can monitor each physical GPU card installed on the server and can determine how actively memory on that card is utilized, thus revealing the card on which memory is used consistently. In addition, the test also indicates how busy each GPU card is, and in the process pinpoints those GPU cards that are being over-utilized by the VMs/virtual desktops on the host. The adequacy of the physical GPU resources is thus revealed. Moreover, detailed diagnostics provided by the test also lead you to those VMs/virtual desktops that are using each card. In addition, the power consumption and temperature of each GPU card is also monitored, so that its current temperature and power usage can be ascertained;

administrators are thus alerted to abnormal power usage of the GPU and unexpected fluctuations in its temperature. The power limit set and the clock frequencies configured are also revealed, so that administrators can figure out whether the GPU is rightly configured for optimal processing or is any fine-tuning required.

Note:

- NVIDIA WMI (NVWMI) is a graphics and display management and control technology that interfaces to Microsoft's Windows Management Instrumentation infrastructure, specific to NVIDIA graphics processing units (GPUs). This allows scripts and programs to be created that configure specific GPU related settings, perform automated tasks, retrieve and display a range of information related to the GPU as well as many other administrative tasks and functions.

  For this test to run and report metrics, the NVWMI should be installed on the vSphere host. To know how, refer to the *Configuring the eG Agent to Monitor NVIDIA Graphics Processing Units (GPUs)* section of the *Monitoring Citrix XenServers* document.

- This test will run and report GPU usage metrics only if the **Shared GPU or Virtual GPU (vGPU) Technology** is used to deliver GPU resources to virtual desktops/VMs.

Target of the test : A VMware vSphere host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each GRID physical GPU assigned to the XenServer host being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **REPORTMANAGERTIME** - By default, this flag is set to **Yes**, indicating that, by default, the detailed diagnosis of this test, if enabled, will report the shutdown and reboot times of the vSphere/ESX host in the manager's time zone. If this flag is set to **No**, then the shutdown and reboot times are shown in the time zone of the system where the agent is running(i.e., the system being managed for agent-based monitoring, and the system on which the remote agent is running - for agentless monitoring).
5. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the

test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing

sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

6. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
7. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
8. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
9. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically

displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

10. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormalfrequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
GPU memory utilization:	Indicates the proportion of time over the past sample period during which global (device) memory was being read or written on this GPU.	Percent	A value close to 100% is a cause for concern as it indicates that graphics memory on a GPU is almost always in use. In a Shared vGPU environment on the other hand, memory may be consumed all the time, if one/more VMs/virtual

Measurement	Description	Measurement Unit	Interpretation
			desktops on the host utilize the graphics memory excessively and constantly. If you find that only a single VM/virtual desktop has been consistently hogging the graphic memory resources, you may want to switch to the Dedicated GPU mode, so that excessive memory usage by that VM/virtual desktop has no impact on the performance of other VMs/virtual desktops on that host. If the value of this measure is high almost all the time for most of the GPUs, it could mean that the host is not sized with adequate graphics memory.
Used frame buffer memory:	Indicates the amount of frame buffer memory on-board this GPU that is being used by the host.	MB	Frame buffer memory refers to the memory used to hold pixel properties such as color, alpha, depth, stencil, mask, etc. Properties like the screen resolution, color level, and refresh speed of the frame buffer can impact graphics performance. Also, if Error-correcting code (ECC) is enabled on a host, the available frame buffer memory may be decreased by several percent. This is because, ECC uses up memory to detect and correct the most common kinds of internal data corruption. Moreover, the driver may also reserve a small amount of memory for internal use, even without active work on the GPU; this too may impact frame buffer memory. For optimal graphics performance therefore, adequate frame buffer memory should be allocated to the host.

Measurement	Description	Measurement Unit	Interpretation
Free frame buffer memory:	Indicates the amount of frame buffer memory on-board this GPU that is yet to be used by the host.	MB	
Frame buffer memory utilization:	Indicates the percentage of frame buffer memory on-board this GPU that is being utilized by the host.	Percent	A value close to 100% is indicative of excessive frame buffer memory usage. Properties like the screen resolution, color level, and refresh speed of the frame buffer can impact graphics performance. Also, if Error-correcting code (ECC) is enabled on a host, the available frame buffer memory may be decreased by several percent. This is because, ECC uses up memory to detect and correct the most common kinds of internal data corruption. Moreover, the driver may also reserve a small amount of memory for internal use, even without active work on the GPU; this too may impact frame buffer memory. For optimal graphics performance therefore, adequate frame buffer memory should be allocated to the host.
GPU compute utilization:	Indicates the proportion of time over the past sample period during which one or more kernels was executing on this GPU.	Percent	A value close to 100% indicates that the GPU is busy processing graphic requests almost all the time. In a Shared vGPU environment on the other hand, a GPU may be in use almost all the time, if one/more VMs/virtual desktops on the host are running highly graphic-intensive applications. If you find that only a single VM/virtual desktop has been consistently hogging the GPU resources, you may want to switch to the Dedicated GPU mode, so that excessive

Measurement	Description	Measurement Unit	Interpretation
			GPU usage by that VM/virtual desktop has no impact on the performance of other VMs/virtual desktops on that host. If all GPUs are found to be busy most of the time, you may want to consider augmenting the GPU resources of the host. Compare the value of this measure across physical GPUs to know which GPU is being used more than the rest.
Power consumption:	Indicates the current power usage of this GPU.	Watts	A very high value is indicative of excessive power usage by the GPU. In such cases, you may want to enable Power management so that the GPU limits power draw under load to fit within a predefined power envelope by manipulating the current performance state.
Core GPU temperature:	Indicates the current temperature of this GPU.	Celsius	Ideally, the value of this measure should be low. A very high value is indicative of abnormal GPU temperature.
Total framebuffer memory:	Indicates the total size of frame buffer memory of this GPU.	MB	Frame buffer memory refers to the memory used to hold pixel properties such as color, alpha, depth, stencil, mask, etc.
Total BAR1 memory:	Indicates the total size of the BAR1 memory of this GPU.	MB	BAR1 is used to map the frame buffer (device memory) so that it can be directly accessed by the CPU or by 3rd party devices (peer-to-peer on the PCIe bus).
Used BAR1 memory:	Indicates the amount of BAR1 memory on this GPU that is being used by to the host.	MB	For better user experience with graphic applications, enough BAR1 memory should be available to the host.

Measurement	Description	Measurement Unit	Interpretation
Free BAR1 memory:	Indicates the total size of BAR1 memory of this GPU that is still not used by the host.	MB	
BAR1 memory utilization:	Indicates the percentage of the total BAR1 memory on this GPU that is currently being utilized by the host.	Percent	A value close to 100% is indicative of excessive BAR1 memory usage by the host. For best graphics performance, sufficient BAR1 memory resources should be available to the host.
Power management:	Indicates whether/not power management is enabled for this GPU.		Many NVIDIA graphics cards support multiple performance levels so that the server can save power when full graphics performance is not required. The default Power Management Mode of the graphics card is Adaptive. In this mode, the graphics card monitors GPU usage and seamlessly switches between modes based on the performance demands of the application. This allows the GPU to always use the minimum amount of power required to run a given application. This mode is recommended by NVIDIA for best overall balance of power and performance. If the power management mode is set to Adaptive, the value of this measure will be Supported. Alternatively, you can set the Power Management Mode to Maximum Performance. This mode allows users to maintain the card at its maximum performance level when 3D applications are running regardless of GPU usage. If the power management mode of a GPU is Maximum Performance, then the value of this measure will be Maximum.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>The numeric values that correspond to these measure values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Supported</td><td>1</td></tr><tr><td>Maximum</td><td>0</td></tr></table>
Note:
By default, this measure will report the Measure Values listed in the table above to indicate the power management status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Power limit:</td><td>Indicates the power limit configured for this GPU.</td><td>Watts</td><td>This measure will report a value only if the value of the ‘Power management’ measure is ‘Supported’.
The power limit setting controls how much voltage a GPU can use when under load. Its not advisable to set the power limit at its maximum – i.e., the value of this measure should not be the same as the value of the Max power limit measure - as it can cause the GPU to behave strangely under duress.</td></tr>
<tr><td></td><td>Default power limit:
Indicates the default power management algorithm’s power ceiling for this GPU.</td><td>Watts</td><td>This measure will report a value only if the value of the ‘Power management’ measure is ‘Supported’.</td></tr>
<tr><td>Enforced power limit:</td><td>Indicates the power management algorithm’s power ceiling for this GPU.</td><td>Watts</td><td>This measure will report a value only if the value of the ‘Power management’ measure is</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			'Supported'. The total board power draw is manipulated by the power management algorithm such that it stays under the value reported by this measure.
Min power limit:	The minimum value that the power limit of this GPU can be set to.	Watts	**This measure will report a value only if the value of the 'Power management' measure is 'Supported'**.
Max power limit:	The maximum value that the power limit of this GPU can be set to.	Watts	**This measure will report a value only if the value of the 'Power management' measure is 'Supported'**. If the value of this measure is the same as that of the Power limit measure, then the GPU may behave strangely.
Graphics clock:	Indicates the current frequency of the graphics clock of this GPU.	MHz	GPU has many more cores than your average CPU but these cores are much simpler and much smaller so that many more actually fit on a small piece of silicon. These smaller, simpler cores go by different names depending upon the tasks they perform. Stream processors are the cores that perform a single thread at a slow rate. But since GPUs contain numerous stream processors, they make overall computation high. The streaming multiprocessor clock refers to how fast the stream processors run. The Graphics clock is the speed at which the GPU operates. The memory clock is how fast the memory on the card runs.

Measurement	Description	Measurement Unit	Interpretation
			By correlating the frequencies of these clocks (i.e., the value of these measures) with the memory usage, power usage, and overall performance of the GPU, you can figure out if overclocking is required or not. Overclocking is the process of forcing a GPU core/memory to run faster than its manufactured frequency. Overclocking can have both positive and negative effects on GPU performance. For instance, memory overclocking helps on cards with low memory bandwidth, and with games with a lot of post-processing/textures/filters like AA that are VRAM intensive. On the other hand, speeding up the operation frequency of a shader/streaming processor/memory, without properly analyzing its need and its effects, may increase its thermal output in a linear fashion. At the same time, boosting voltages will cause the generated heat to sky rocket. If improperly managed, these increases in temperature can cause permanent physical damage to the core/memory or even “heat death”. Putting an adequate cooling system into place, adjusting the power provided to the GPU, monitoring your results with the right tools and doing the necessary research are all critical steps on the path to safe and successful overclocking.
Streaming multiprocessor clock:	Indicates the current frequency of the streaming multiprocessor clock of this GPU.	MHz	
Memory clock:	Indicates the current frequency of the memory clock of this GPU.	MHz	
Fan speed:	Indicates the percent of maximum speed that this GPU’s fan is currently intended to run at.	Percent	The value of this measure could range from 0 to 100%. An abnormally high value for this measure could indicate a problem condition – eg., a sudden surge in the

Measurement	Description	Measurement Unit	Interpretation
			temperature of the GPU that could cause the fan to spin faster. Note that the reported speed is only the intended fan speed. If the fan is physically blocked and unable to spin, this output will not match the actual fan speed. Many parts do not report fan speeds because they rely on cooling via fans in the surrounding enclosure. By default the fan speed is increased or decreased automatically in response to changes in temperature.
Compute processes:	Indicates the number of processes having compute context on this GPU.	Number	Use the detailed diagnosis of this measure to know which processes are currently using the GPU. The process details provided as part of the detailed diagnosis include, the PID of the process, the process name, and the GPU memory used by the process. Note that the GPU memory usage of the processes will not be available in the detailed diagnosis, if the Windows platform on which XenApp operates is running in the WDDM mode. In this mode, the Windows KMD manages all the memory, and not the NVIDIA driver. Therefore, the NVIDIA SMI commands that the test uses to collect metrics will not be able to capture the GPU memory usage of the processes.
Volatile single bit errors:	Indicates the number of volatile single bit errors in this GPU.	Number	Volatile error counters track the number of errors detected since the last driver load. Single bit ECC errors are automatically corrected by the hardware and do not result in data corruption. Ideally, the value of this measure should be 0.

Measurement	Description	Measurement Unit	Interpretation
Volatile double bit errors:	Indicates the total number of volatile double bit errors in this GPU.	Number	Volatile error counters track the number of errors detected since the last driver load. Double bit errors are detected but not corrected. Ideally, the value of this measure should be 0.
Aggregate single bit errors:	Indicates the total number of aggregate single bit errors in this GPU.	Number	Aggregate error counts persist indefinitely and thus act as a lifetime counter. Single bit ECC errors are automatically corrected by the hardware and do not result in data corruption. Ideally, the value of this measure should be 0.
Aggregate double bit errors:	Indicates the total number of aggregate double bit errors in this GPU.	Number	Aggregate error counts persist indefinitely and thus act as a lifetime counter. Double bit errors are detected but not corrected. Ideally, the value of this measure should be 0.

3.3 The Network Layer

This layer tracks the network connectivity to the ESX server host, and also monitors the traffic to and from each of the network interfaces of the ESX host.



Figure 3.12: The tests mapped to the Network layer

The **Network** test depicted by Figure 3.12 has already been discussed in the *Monitoring Unix and Windows Servers* document. The following section will hence discuss the **Network - Esx** test only.

3.3.1 Network - ESX Test

This test reports key statistics pertaining to the network traffic to and from every network interface supported by the ESX server host.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every network interface supported by the ESX server host monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status

information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the

test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7**.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP

address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **REPORTACTIVEONLY** - By default, this test reports metrics for only those network interfaces that are connected to a vSwitch. Accordingly, the **REPORTACTIVEONLY** flag is set to **Yes** by default. To ensure that the test reports metrics for all network interfaces, regardless of whether/not they are connected to a vSwitch, set this flag to **No**.
10. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Network packets transmitted:	Indicates the rate at which packets were transmitted by this network interface.	Packets/Sec	An increase in traffic from the server can indicate an increase in accesses from the server (from users on the ESX server, or from the virtual machines on the server). Use the detailed diagnosis of the Network packets transmitted measure to know the rate at which packets were transmitted by each VM on the target virtual host via this NIC. Likewise, you can use the detailed diagnosis of the Network data transmitted measure to know how much data was transmitted per second by every VM on this hypervisor via this NIC.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td rowspan="2">From these detailed metrics, you can precisely pinpoint the VM that is imposing the maximum load on this NIC.</td></tr>
<tr><td>Network data transmitted:</td><td>Indicates the rate at which data was transmitted by this network interface.</td><td>Mbps</td></tr>
<tr><td>Network packets received:</td><td>Indicates the rate at which data packets were received by this network interface.</td><td>Packets/Sec</td><td rowspan="2">An increase in traffic to the server can indicate an increase in accesses to the server (from users or from other applications external to the server) or that the server is under an attack of some form.

Use the detailed diagnosis of the Network packets received measure to know the rate at which packets were received by each VM on the target virtual host via this NIC. Likewise, you can use the detailed diagnosis of the Network data received measure to know how much data was received per second by every VM on this hypervisor via this NIC.

From these detailed metrics, you can precisely pinpoint the VM that is imposing the maximum load on this NIC.</td></tr>
<tr><td>Network data received:</td><td>Indicates the rate at which data was received by this network interface.</td><td>Mbps</td></tr>
<tr><td>Status:</td><td>Indicates whether the network interface card is available or not.</td><td></td><td>If this measure reports the value Up, it indicates that the NIC is accessible. Whereas, if the status is Down, it indicates that the NIC cannot be accessed.

The numeric values that correspond to each of the states discussed above are listed in the table below:
<table><tr><th>State</th><th>Value</th></tr><tr><td>Up</td><td>100</td></tr><tr><td>Down</td><td>0</td></tr></table></td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			Note: By default, this measure reports the value Up or Down to indicate the availability of the NIC. The graph of this measure however, represents the status of an NIC using the numeric equivalents - 100 or 0.
Speed:	Indicates the current speed of this network interface card.	Mbps	Compare the value of this measure across NICs to identify the slowest NIC and to assess network usage across NICs. You can also use the value of this measure to predict the maximum throughput that can be reached.
Packets dropped during transmission:	Indicates the rate at which packet were dropped by this interface during transmissions.	Packets/Sec	A low value is typically desired for this measure.
During transmission dropped packets:	Indicates the percentage of packets that were dropped by this network interface during transmissions.	Percent	A very high value of this measure is a cause for concern, as it could indicate a network congestion.
Packets dropped during reception:	Indicates the rate at which packets were dropped by this interface during reception.	Packets/Sec	A low value is typically desired for this measure.
During reception dropped packets:	Indicates the percentage of packets that were dropped by this network interface during reception.	Percent	A very high value of this measure is a cause for concern, as it could indicate a network congestion.
Network IOPS:	Indicates the rate at which packets were received and transmitted by this network	Packets/Sec	A high value is desired for this measure. A consistent drop in the value of this measure is a cause for concern, as it

Measurement	Description	Measurement Unit	Interpretation
	interface.		indicates a strain on the network interface, which is causing it to slowdown.
Network throughput:	Indicates the rate at which data was received and transmitted by this network interface	MB/Sec	A high value is desired for this measure. A consistent drop in the value of this measure is a cause for concern, as it indicates a strain on the network interface, which is causing it to slowdown.

To know the network health across all network interfaces supported by the ESX server, just click on the descriptor with the name of the vSphere/ESXi server being monitored. For instance, in the case of Figure 3.12, click on the descriptor **esx3i.chn.egurkha.com** to view the following measure:

Measurement	Description	Measurement Unit	Interpretation
Network usage:	Indicates the rate at which data transmitted and received for all the NIC instances of the host.	Mbps	Use the detailed diagnosis of this measure to know the data usage per VM on the monitored hypervisor. From this, you can quickly identify the VM that is making maximum use of the network resources of the vSphere/ESX server.

3.4 The Virtual Network Layer

A virtual network is a network of virtual machines running on a single physical machine that are connected logically to each other so that they can send data to and receive data from each other. Each virtual network is serviced by a single virtual switch (vSwitch). Typically, every vSwitch is associated with one/more port groups; a port group specifies port configuration options such as bandwidth limitations and VLAN tagging policies for each logical port on the vSwitch, and thus defines how a connection to the network is made via the vSwitch.

The **Virtual Network** layer provides insights into the composition of the virtual networks on a vSphere/ESX server and the traffic to and from each of the virtual networks, so that administrators can promptly detect overloads, accurately identify the vSwitch-port group combination which handled the heavy traffic, and instantly isolate the VM in the virtual network to/from which the data was transmitted.

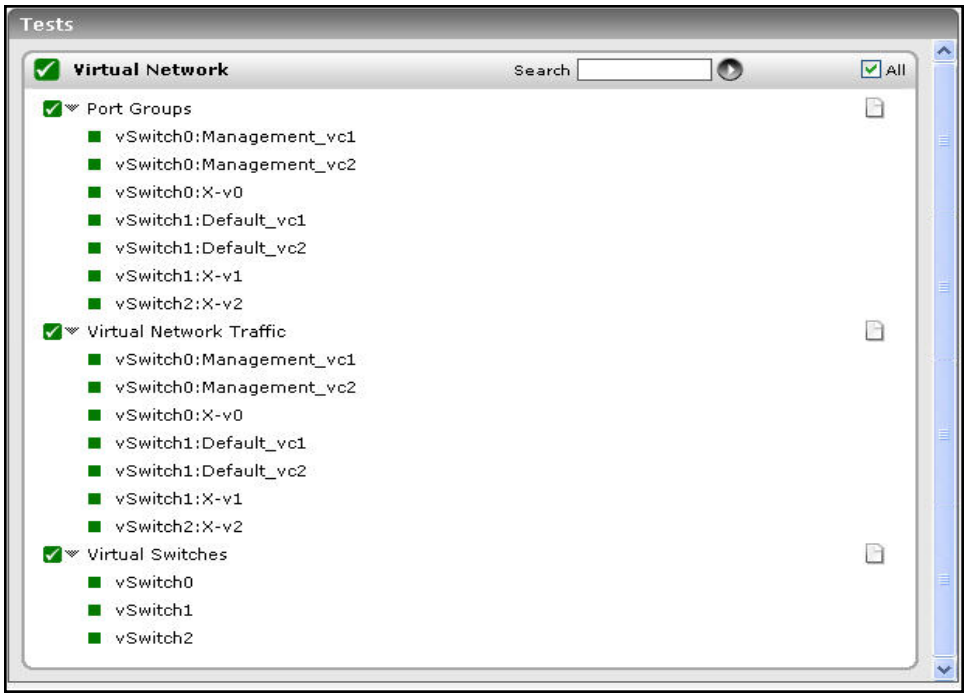


Figure 3.13: The Virtual Network Layer

3.4.1 Virtual Network Traffic Test

Each virtual network is serviced by a single virtual switch (vSwitch). Typically, every vSwitch is associated with one/more port groups; a port group specifies port configuration options such as bandwidth limitations and VLAN tagging policies for each logical port on the vSwitch, and thus defines how a connection to the network is made via the vSwitch.

This test monitors the traffic to and from each of the port groups associated with every vSwitch that services the virtual networks on an ESX server. Besides alerting administrators to abnormally high traffic on a vSwitch, this test also leads them to the exact VM in the virtual network that is responsible for this abnormality.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each port group associated with every virtual switch that services the virtual networks on the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user

credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

- **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

  Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-

enabled vCenter, the eG agent connects to port 80 of vCenter.

Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

   The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

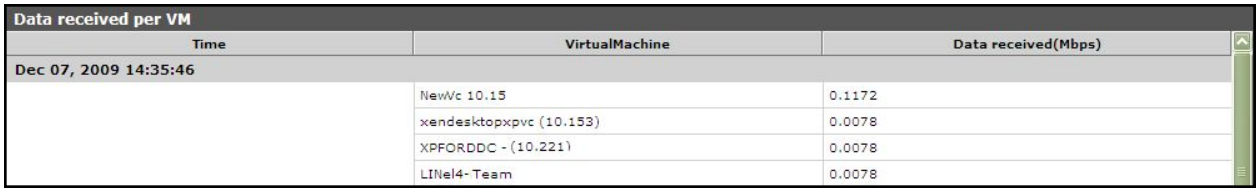
   - The eG manager license should allow the detailed diagnosis capability
   - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Data received:	Indicates the rate at which data is received by this port group mapped to this vSwitch.	Mbps	An increase in traffic to the server can indicate an increase in accesses to the server (from users or from other applications external to the server) or that the server is under an attack of some form. The detailed diagnosis of this measure lists the top-10 VMs in the virtual network that are receiving data at a high rate. Using this information, you can accurately identify the VM that is experiencing the most traffic.
Packets received:	Indicates the rate at which data packets were received by this port group mapped to this vSwitch.	Packets/Sec	An increase in traffic to the server can indicate an increase in accesses to the server (from users or from other applications external to the server) or that the server is under an attack of some form. The detailed diagnosis of this measure lists the top-10 VMs in the virtual network that are receiving packets at a high rate. Using this information, you can accurately identify the VM that is experiencing the most traffic.
Data transmitted:	Indicates the rate at which data was transmitted by this port group mapped to this vSwitch.	Mbps	An increase in traffic from the server can indicate an increase in accesses from the server (from users on the ESX server, or from the virtual machines on the server). The detailed diagnosis of this measure lists the top-10 VMs in the virtual network that are sending data at a high rate. Using this information, you can accurately identify the VM that is experiencing the most traffic.

Measurement	Description	Measurement Unit	Interpretation
Packets transmitted:	Indicates the rate at which packets were transmitted by this port group mapped to this vSwitch.	Packets/Sec	An increase in traffic from the server can indicate an increase in accesses from the server (from users on the ESX server, or from the virtual machines on the server). The detailed diagnosis of this measure lists the top-10 VMs in the virtual network that are sending packets at a high rate. Using this information, you can accurately identify the VM that is experiencing the most traffic.

The detailed diagnosis of the *Data received* measure lists the top-10 VMs in the virtual network that are receiving data at a high rate. Using this information, you can accurately identify the VM that is experiencing the most traffic.

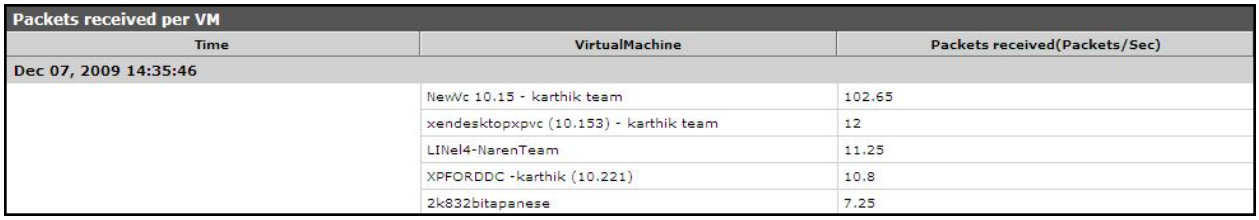


Data received per VM

Time	VirtualMachine	Data received(Mbps)
Dec 07, 2009 14:35:46		
	NewVc 10.15	0.1172
	xendesktopxpvc (10.153)	0.0078
	XPFORDDC - (10.221)	0.0078
	LINel4-Team	0.0078

Figure 3.14: The detailed diagnosis of the Data received measure

The detailed diagnosis of the *Packets received* measure lists the top-10 VMs in the virtual network that are receiving packets at a high rate. Using this information, you can accurately identify the VM that is experiencing the most traffic.

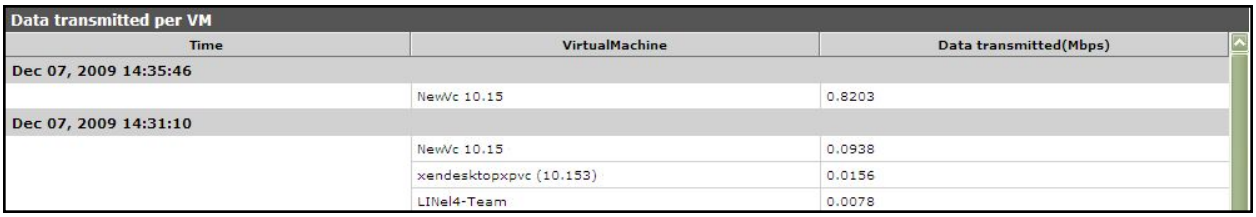


Packets received per VM

Time	VirtualMachine	Packets received(Packets/Sec)
Dec 07, 2009 14:35:46		
	NewVc 10.15 - karthik team	102.65
	xendesktopxpvc (10.153) - karthik team	12
	LINel4-NarenTeam	11.25
	XPFORDDC -karthik (10.221)	10.8
	2k832bitapanese	7.25

Figure 3.15: The detailed diagnosis of the Packets received measure

The detailed diagnosis of the *Data transmitted* measure lists the top-10 VMs in the virtual network that are sending data at a high rate. Using this information, you can accurately identify the VM that is experiencing the most traffic.

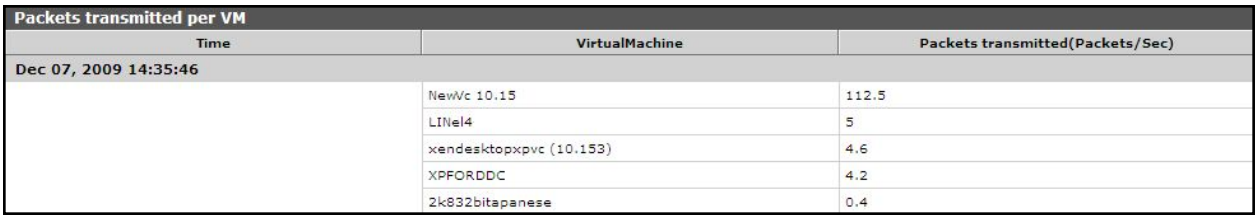


Data transmitted per VM

Time	VirtualMachine	Data transmitted(Mbps)
Dec 07, 2009 14:35:46		
	NewVc 10.15	0.8203
Dec 07, 2009 14:31:10		
	NewVc 10.15	0.0938
	xendesktopxpvc (10.153)	0.0156
	LINel4-Team	0.0078

Figure 3.16: The detailed diagnosis of the Data transmitted measure

The detailed diagnosis of the *Packets transmitted* measure lists the top-10 VMs in the virtual network that are sending packets at a high rate. Using this information, you can accurately identify the VM that is experiencing the most traffic.



Packets transmitted per VM

Time	VirtualMachine	Packets transmitted(Packets/Sec)
Dec 07, 2009 14:35:46		
	NewVc 10.15	112.5
	LINel4	5
	xendesktopxpvc (10.153)	4.6
	XPFORDDC	4.2
	2k832bitapanese	0.4

Figure 3.17: The detailed diagnosis of the Packets transmitted measure

3.4.2 Port Groups – ESX Test

Port groups aggregate multiple ports under a common configuration and provide a stable anchor point for virtual machines connecting to labeled networks. Each port group is identified by a network label, which is unique to the current host. All members of a port group share common attributes like a VLAN tag. All members of a port group can see all of the packets sent by other members of this port group. A port group is always connected to a single vSwitch.

This test auto-discovers all the port groups connecting to each vSwitch on an ESX server, and reports the number and powered-on state of the VMs in the virtual network managed by every port group. Powered-off VMs in the virtual network can thus be quickly identified.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each port group associated with every virtual switch that services the virtual networks on the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target

ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

- **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6**.

  Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7**.

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the

eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

   The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

   - The eG manager license should allow the detailed diagnosis capability
   - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total VMs:	Indicates the number of VMs in this port group.	Number	The detailed diagnosis of this measure reveals the names of the VMs in this port group.
VMs powered off:	Indicates the number of powered off VMs in this port group.	Number	The detailed diagnosis of this measure reveals the names of the powered off VMs.

The detailed diagnosis of the *Total VMs* in port group measure reveals the names of the VMs in this port group.

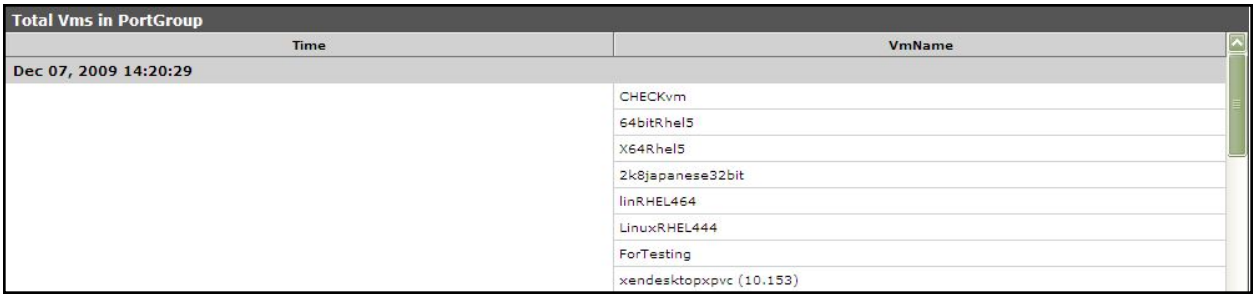


Figure 3.18: The detailed diagnosis of the Total VMs in port group measure

The detailed diagnosis of the *PoweredOff VMs in PortGroup* measure reveals the names of the powered off VMs.

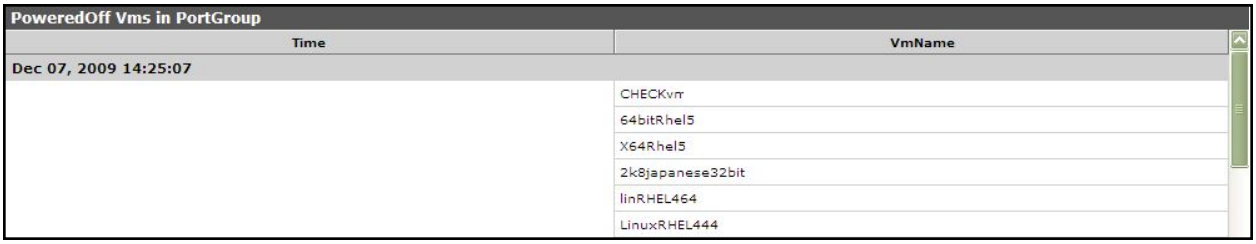


Figure 3.19: The detailed diagnosis of the PoweredOff VMs in PortGroup measure

3.4.3 Virtual Switches Test

The Virtual Switch is nothing more than a logical switching fabric built into your VMware infrastructure (ESX) so that you can network your VMs however you need them. While each virtual network (network of VMs) on an ESX server will be connecting to a single vSwitch only, a vSwitch may provide network support to more than one virtual network on an ESX server.

This test monitors the virtual networks connecting to each vSwitch on an ESX server, and reports the number of VMs, port groups, and physical adapters associated with these virtual networks. In addition, the test also reveals those adapters that are currently switched off.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each virtual switch that services the virtual networks on the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into

it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then

refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who

has at least 'Read-only' privileges to the target ESX server.

9. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

   The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

   - The eG manager license should allow the detailed diagnosis capability
   - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Portgroups in vSwitch:	Indicates the number of port groups associated with this vSwitch.	Number	Use the detailed diagnosis of this measure to determine the names of the port groups connecting to this vSwitch.
VMs in vSwitch:	Indicates the total number of VMs connected to this vSwitch.	Number	The detailed diagnosis of this measure reveals the names of the VMs serviced by the vSwitch and the port groups with which each VM is associated.
Physical NICs in vSwitch:	Indicates the number of physical network adapters associated with this vSwitch.	Number	A virtual network can be connected to a physical network by associating one or more physical Ethernet adapters, also referred to as uplink adapters, with the virtual network's virtual switch. If no uplink adapters are associated with the virtual switch, all traffic on the virtual network is confined within the physical host machine. If one or more uplink adapters are associated with the virtual switch, virtual machines connected to that virtual network are also able to access the physical networks connected to the uplink adapters. To know which adapters are linked with this vSwitch, use the detailed diagnosis

Measurement	Description	Measurement Unit	Interpretation
			of this measure.
Physical NICs in vSwitch that are up:	Indicates the number of physical network adapters that are up.	Number	Use the detailed diagnosis of this measure to figure out which physical adapters are up.
Physical NICs in vSwitch that are down:	Indicates the number of physical network adapters that are down.	Number	Use the detailed diagnosis of this measure to figure out which physical adapters are down.
Allocated NICs down:	Indicates the percentage of physical network adapters that are down.	Percent	

Use the detailed diagnosis of the *Portgroups in vSwitch* measure to determine the names of the port groups connecting to a particular vSwitch.

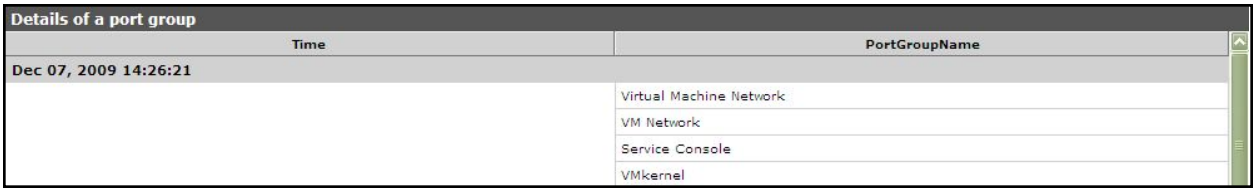
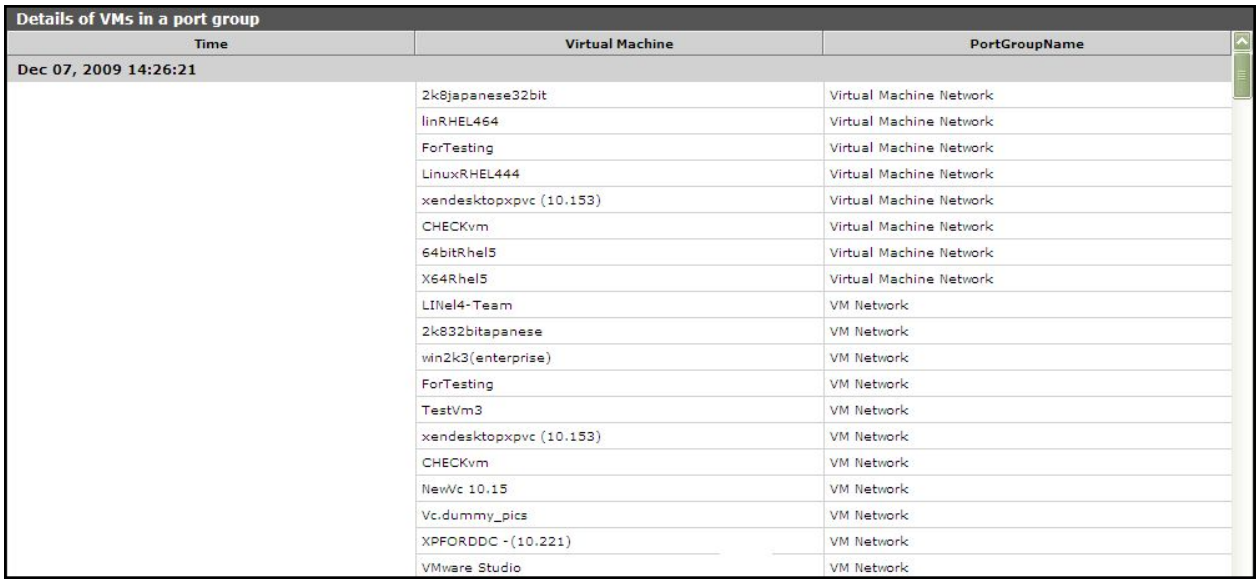


Figure 3.20: The detailed diagnosis of the Portgroups in vSwitch measure

The detailed diagnosis of the *Virtual machine in vSwitch* measure reveals the names of the VMs serviced by the vSwitch and the port groups with which each VM is associated.

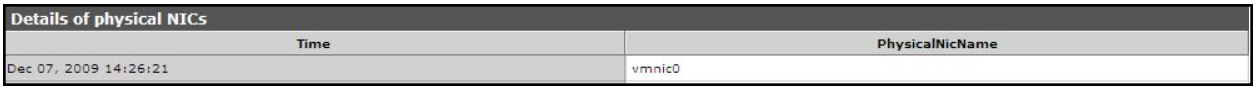


Details of VMs in a port group

Time	Virtual Machine	PortGroupName
Dec 07, 2009 14:26:21		
	2k8japanese32bit	Virtual Machine Network
	linRHEL464	Virtual Machine Network
	ForTesting	Virtual Machine Network
	LinuxRHEL444	Virtual Machine Network
	xendesktopxpvc (10.153)	Virtual Machine Network
	CHECKvm	Virtual Machine Network
	64bitRhel5	Virtual Machine Network
	X64Rhel5	Virtual Machine Network
	LINel4-Team	VM Network
	2k832bitapanese	VM Network
	win2k3(enterprise)	VM Network
	ForTesting	VM Network
	TestVm3	VM Network
	xendesktopxpvc (10.153)	VM Network
	CHECKvm	VM Network
	NewVc 10.15	VM Network
	Vc.dummy_pics	VM Network
	XPFORDDC -(10.221)	VM Network
	VMware Studio	VM Network

Figure 3.21: The detailed diagnosis of the Virtual Machine in vSwitch measure

The detailed diagnosis of the *Physical NICs in vSwitch* measure displays the names of the physical adapters associated with a particular vSwitch.

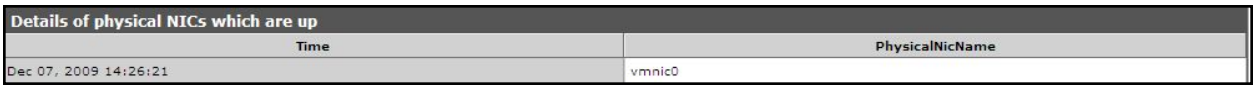


Details of physical NICs

Time	PhysicalNicName
Dec 07, 2009 14:26:21	vmnic0

Figure 3.22: The detailed diagnosis of the Physical NICs in vSwitch measure

Use the detailed diagnosis of the *Physical NICs up in vSwitch* measure to figure out which physical adapters are up.

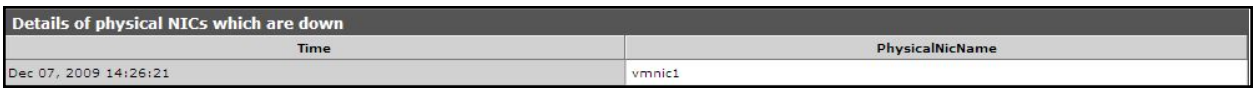


Details of physical NICs which are up

Time	PhysicalNicName
Dec 07, 2009 14:26:21	vmnic0

Figure 3.23: The detailed diagnosis of the Physicals NICs up in vSwitch measure

Use the detailed diagnosis of the *Physical NICs down in vSwitch* measure to figure out which physical adapters are down.



Details of physical NICs which are down

Time	PhysicalNicName
Dec 07, 2009 14:26:21	vmnic1

Figure 3.24: The detailed diagnosis of the Physicals NICs down in vSwitch measure

3.4.4 Distributed Port Groups Test

Distributed Virtual Port Groups (DV Port Groups) are port groups associated with a vSphere Distributed Switch (vDS) and specify port configuration options for each member port. DV Port Groups define how a connection is made through the vDS to the network. Configuration parameters are similar to those available

with Port Groups on Standard Switches. The VLAN ID, traffic shaping parameters, port security, teaming and load balancing configuration, and other settings are configured here. Each VDS supports up to 5000 static port groups.

This test auto-discovers all the distributed port groups connecting to each distributed switch on an ESX server, and reports the number and powered-on state of the VMs in the virtual network managed by every port group. Powered-off VMs in the virtual network and the distributed port groups they are associated with them can thus be quickly identified.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each distributed port group that services the virtual networks on the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4**.

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

     Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status

information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the

test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP

address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
10. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total VMs:	Indicates the number of VMs managed by this distributed port group.	Number	Use the detailed diagnosis of this measure to determine the names of the VMs managed by a particular distributed port group.
VMs powered off:	Indicates the number of powered off VMs managed by this distributed port group.	Number	The detailed diagnosis of this measure reveals the names of the powered off VMs.
VMs powered on:	Indicates the number of powered on VMs in this distributed port group.	Number	The detailed diagnosis of this measure reveals the names of the powered on VMs.

3.4.5 Distributed Switches Test

The vSphere Distributed Switch (VDS) extends the features and capabilities and features of virtual networks while simplifying provisioning and the ongoing process of configuration, monitoring, and management.

With ESX 3.5 and prior releases, virtual networks were constructed using virtual switches. Each ESX host would use one or more virtual switches to connect the VMs with the server NICs and the outside physical network. With the release of vSphere 4.0, VMware introduced an additional choice for VMware virtual networking with the vSphere Distributed Switch. VDS eases the management burden of per host virtual switch configuration by treating the network as an aggregated resource. Individual host-level virtual switches are abstracted into a single large vSphere Distributed Switch that spans multiple hosts at the Datacenter level. Management plane and Proxy switch are two main components of VDS. Management plane is part of the vCenter server and helps manage virtual network infrastructure at scale through a single pane of glass. Proxy switch on the other hand manages VM traffic flows on the local host. Each vCenter can support up to 32 VDSs and each VDS can manage up to 350 hosts. Many of the concepts involved in configuring and managing a Standard Switch are carried forward with the VDS. Port Groups become Distributed Virtual Port Groups (DV Port Groups) that span multiple hosts and ensure configuration consistency for VMs and virtual ports necessary for such functions as VMotion.

This test monitors the virtual networks connecting to each distributed Switch on an ESX server, and reports the number of VMs, port groups, and physical adapters associated with these virtual networks. In addition, the test also reveals those adapters that are currently switched off.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each distributed switch on the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the

"Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the

credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text

boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
10. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Port groups in vSwitch:	Indicates the number of port groups associated with this	Number	Use the detailed diagnosis of this measure to determine the names of the

Measurement	Description	Measurement Unit	Interpretation
	distributed switch.		distributed port groups connecting to this distributed switch.
VMs in vSwitch:	Indicates the total number of VMs connected to this distributed switch.	Number	The detailed diagnosis of this measure reveals the names of the VMs serviced by the distributed switch and the distributed port groups with which each VM is associated
Physical NICs in vSwitch:	Indicates the number of physical network adapters associated with this distributed switch.	Number	
Physical NICs in vSwitch that are up:	Indicates the number of physical network adapters that are up.	Number	Use the detailed diagnosis of this measure to figure out which physical adapters are up.
Physical NICs in vSwitch that are down:	Indicates the number of physical network adapters that are down.	Number	Use the detailed diagnosis of this measure to figure out which physical adapters are down.
Allocated NICs down:	Indicated the percentage of physical network adapters that are down	Percentage	

3.5 The VM Processes Layer

The tests associated with this layer monitor the critical log files of the ESX server for errors/warnings/general information messages. In addition, the layer also indicates the availability and responsiveness of the ESX server, from an external perspective.

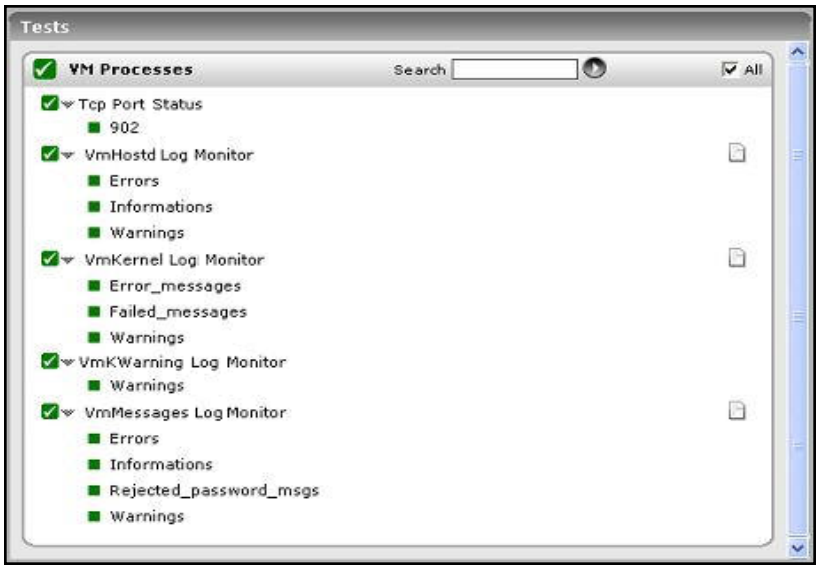


Figure 3.25: The tests mapped to the VM Processes layer

3.5.1 TCP Port Status Test

This test indicates the health of TCP connections to the ESX server by periodically verifying the availability and responsiveness of configured ESX server ports.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every configured port name

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - Host name of the server for which the test is to be configured
3. **PORT** - Enter the port to which the specified **HOST** listens
4. **TARGETPORTS** - Specify either a comma-separated list of port numbers that are to be tested (eg., 80,7077,1521), or a comma-separated list of *port name:port number* pairs that are to be tested (eg., smtp:25,mssql:1433). In the latter case, the port name will be displayed in the monitor interface. Alternatively, this parameter can take a comma-separated list of *port name:IP address:port number* pairs that are to be tested, so as to enable the test to try and connect to Tcp ports on multiple IP addresses. For example, *mysql:192.168.0.102:1433,egwebsite:209.15.165.127:80*.
5. **TIMEOUT** - Here, specify the maximum duration (in seconds) for which the test will wait for a response from the server. The default **TIMEOUT** period is 60 seconds.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Availability:	Whether the TCP connection is available	Percent	An availability problem can be caused by different factors - e.g., the server process may not be up, a network problem may exist, or there could be a configuration problem with the DNS server.
Response time:	Time taken (in seconds) by the server to respond to a request.	Secs	An increase in response time can be caused by several factors such as a server bottleneck, a configuration problem with the DNS server, a network problem, etc.

3.5.2 VmHostd Log Monitor Test

The VmHostdLogTest monitors the ESX Server host agent log file to which the activities of the agent that configures and manages the ESX server host and its virtual machines are logged. The test scans the log file for entries matching configured patterns, and alerts administrators whenever such entries are found in the log file.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every log file pattern configured for monitoring

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - To enable this test to read the log file and report metrics, the test should be configured with an **ESX USER** and **ESX PASSWORD**.

   If the metrics are to be collected by directly connecting to the ESX server in question (i.e., if the **VIRTUAL CENTER** parameter is set to none), then a user with Administrator privileges to the target ESX server can be set as the **ESX USER**. However, if, owing to security constraints, you cannot expose the credentials of the Administrator, then you can create a special role for this purpose in the ESX server, and grant the Diagnostics permission to this role. You can then add a new user to the ESX server, and

assign the newly created role to that user. The detailed procedure for accomplishing this task is available inSection **3.5.2.1** .

On the other hand, if the test needs to connect to vCenter for collecting the log-file related metrics (i.e., if the IP address of vCenter is specified against the **VIRTUAL CENTER** parameter), then a user with Administrator privileges to vCenter must be set as the **ESX USER**. Here again, if you cannot expose the credentials of the Administrator due to security restrictions imposed by your environment, then, you can create a special role in vCenter, grant Diagnostics permission to this role, and then assign this role to a local/domain user to vCenter. The credentials of this user can then be passed to the ESX USER and ESX PASSWORD parameters. The steps for creating such a role in vCenter and assigning the role to a local/domain user are documented in Section **3.5.2.1** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via **HTTPS** by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section

3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **HOSTD PATTERNS** - In the **HOSTD PATTERNS** text box, specify the pattern to search for in the log file in the following format: <*PatternName*>:<*Pattern*>, where <*PatternName*> is the pattern name that will be displayed in the monitor interface and <*Pattern*> is an expression of the form - *expr or expr or expr or expr*, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.

   For example, say you specify **VM:VM-*** in the **HOSTD PATTERNS** text box. This indicates that "**VM**"is the pattern name that will be displayed as a descriptor of this test in the monitor interface. "**VM-***" indicates that the test will monitor only those lines in the log file which start with the term "**VM-**". Similarly, if your pattern specification reads: **offline:*offline**, then it means that the pattern name is **offline** and that the test will monitor those lines in the log which end with the term **offline**.

   Multiple search patterns can be specified as a comma-separated list. For example: **VM:VM-*,offline:*offline*,online:*online**. If you want to monitor all the entries in the log file, your specification would be all:all.

10. **IGNORE HOSTD LINES** - In the **IGNORE HOSTD LINES** text box, specify the pattern of entries that is to be excluded from monitoring. For example, if you want the test to disregard those entries which begin with the term **VM**, then specify **VM*** in the **IGNORE HOSTD LINES** text box. Multiple search patterns can be specified as a comma-separated list. For example, **VM*,*VM**. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. If no lines are to be ignored, then enter *none* here. If you want to ignore all the lines in the log file, your specification would be **all:all**.

11. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

12. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New entries:	Indicates the number of new entries in the ESX server host agent log that match the configured patterns.	Number	The detailed diagnosis of this measure, if enabled, provides the details of these entries.

3.5.2.1 Configuring a Special Role and User on the ESX Server for Log Monitoring

If the eG agent has been configured to directly connect to the target ESX server for reading the log files - i.e., if the **VCENTER** parameter of all the log tests is set to *none* - then, to remotely monitor the log files on an ESX server, the eG agent requires **Administrator** privileges to the ESX server in question. In other words, the **ESX USER** and **ESX PASSWORD** parameters of the log test should contain the login credentials of a user with **Administrator** rights to the ESX server. However, if, owing to security constraints, you cannot expose the credentials of the **Administrator**, then you can create a special role for this purpose in the ESX server, and grant the **Diagnostics** permission to this role. You can then add a new user to the ESX server, and assign the newly created role to that user. The procedure for achieving this is detailed below:

1. Login to a system on which the **VMware Infrastructure Client** is installed.
2. Double-click on the **VMware Infrastructure Client** icon on your desktop.
3. Figure 3.26 then appears. To connect to an ESX server, select the **IP address / Name** of the ESX server, and then provide the login information. To grant access permissions to a user, you will have to login to vCenter as an *administrator*. Therefore, provide the **User name** and **Password** of the administrator in Figure 3.26.



Figure 3.26: Connecting to vCenter

4. When the **Virtual Infrastructure Client** opens, click on the **Administration** icon indicated by Figure 3.27.

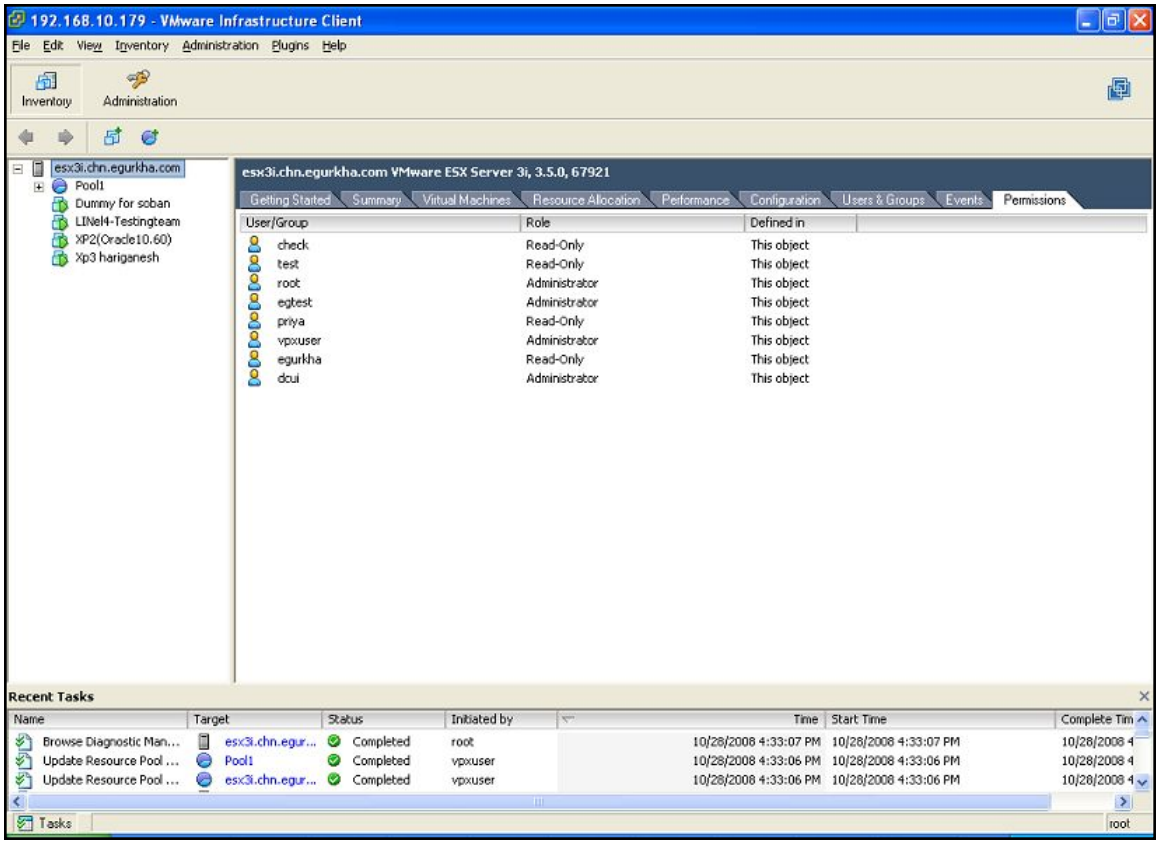


Figure 3.27: Clicking on the Administration icon

5. The **Roles** tab page opens by default, revealing the roles that pre-exist on the ESX server (see Figure 3.28).

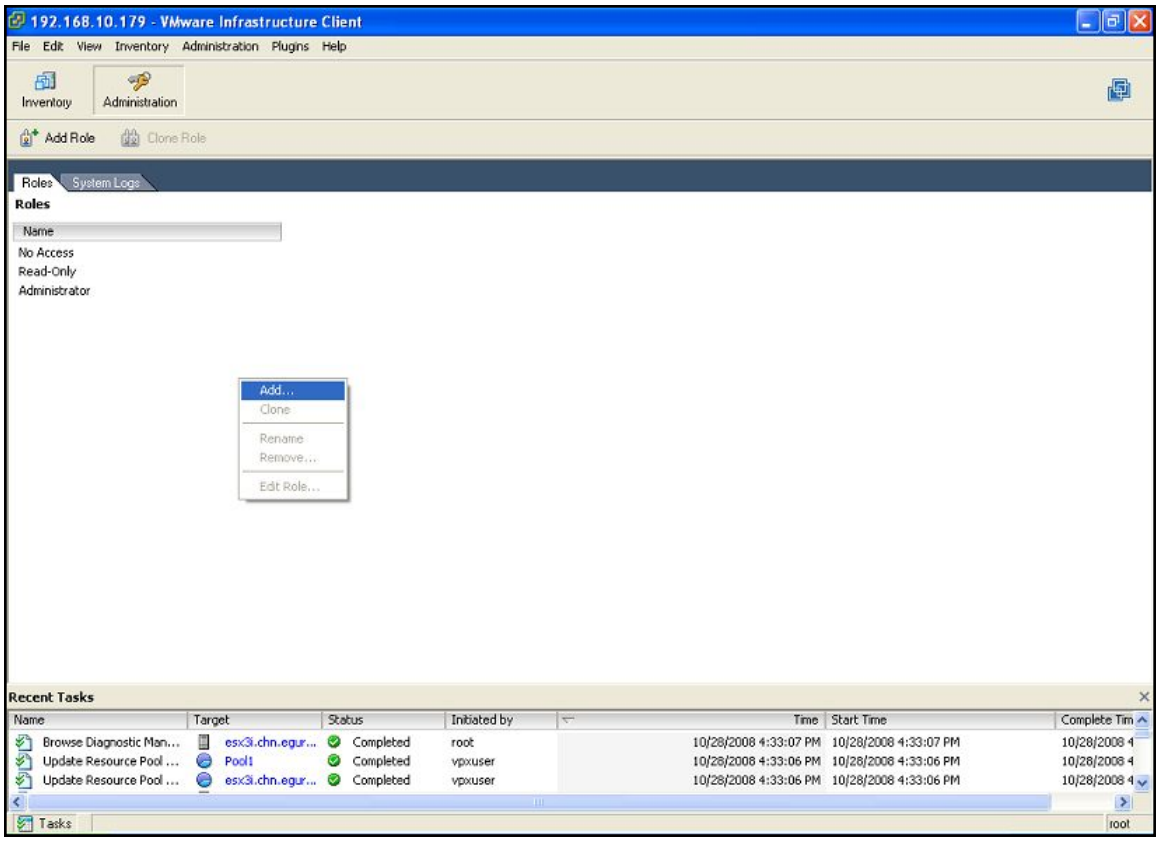


Figure 3.28: Viewing the roles available on the ESX server

6. To create a special role, right-click anywhere within Figure 3.28, and select **Add** from the shortcut menu that appears.
7. Figure 3.29 then appears wherein you need to **Enter Name** of the new role. Since the role requires **Diagnostics** privileges, expand the **Global** node in Figure 3.29, and select the **Diagnostics** check box under it.

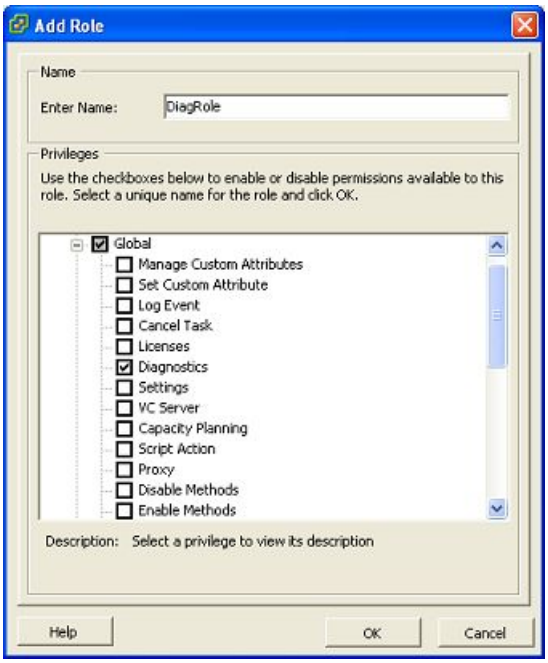


Figure 3.29: Creating a new role and assigning Diagnostics privilege to it

8. Then, click the **OK** button in Figure 3.29 to save the changes.
9. Figure 3.30 then appears listing the newly created role. Once the role is created, proceed to assign the role to a local/domain user to vCenter. The first step towards achieving this is to click on the **Inventory** icon indicated by Figure 3.30.

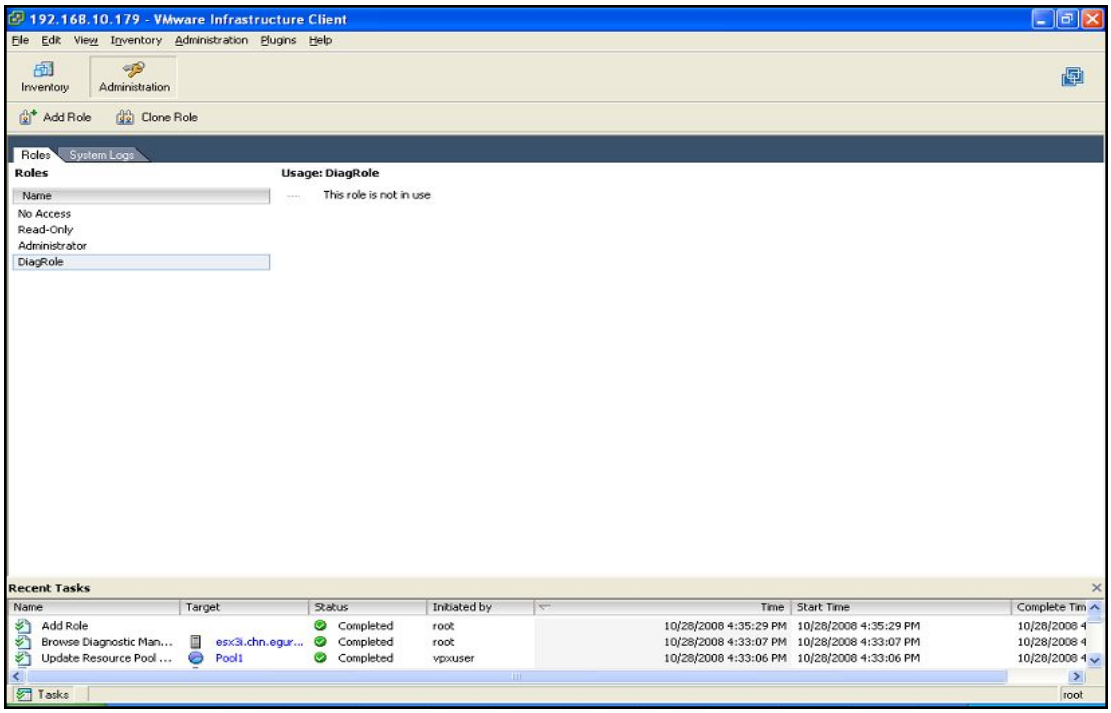


Figure 3.30: The Roles tab page listing the newly created role

10. Figure 3.31 then appears. To create a new user on the ESX server, click on the **Users & Groups** tab page in the right panel of Figure 3.31.

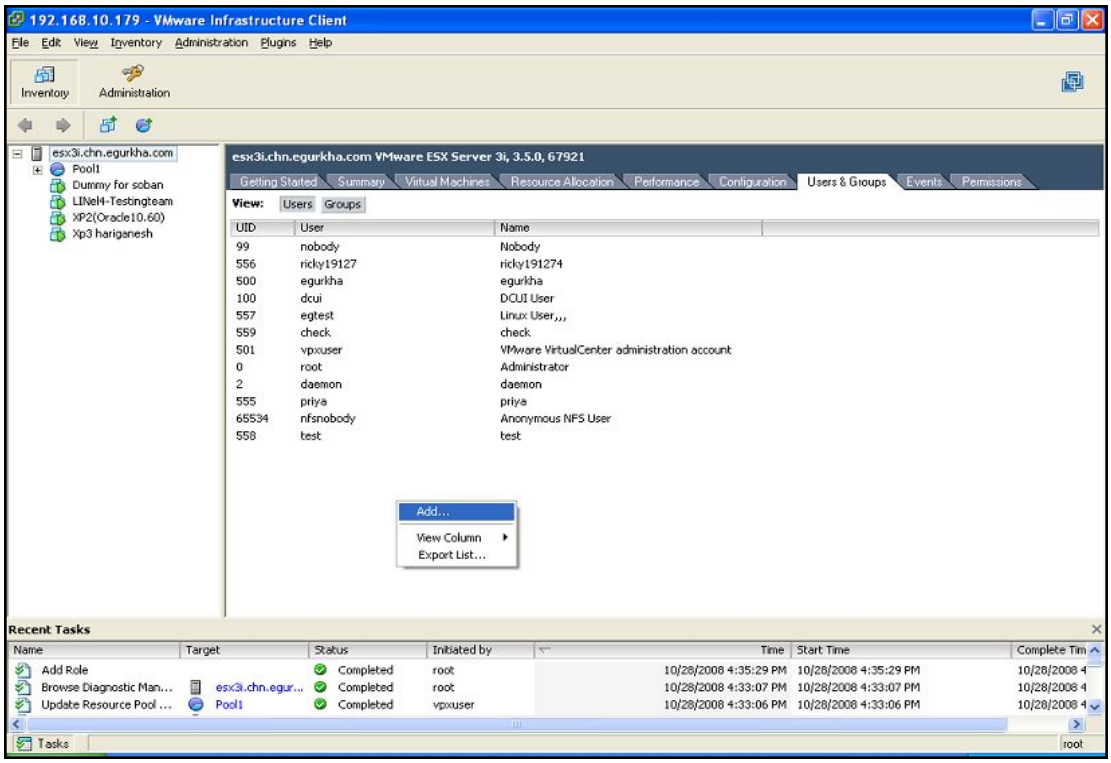


Figure 3.31: Details of users to the ESX server

11. When the **Users & Groups** tab page opens, right-click anywhere within the tab page, and pick the **Add** option from the shortcut menu that appears (see Figure 3.31). Figure 3.32 then opens, using which you can create a new user.

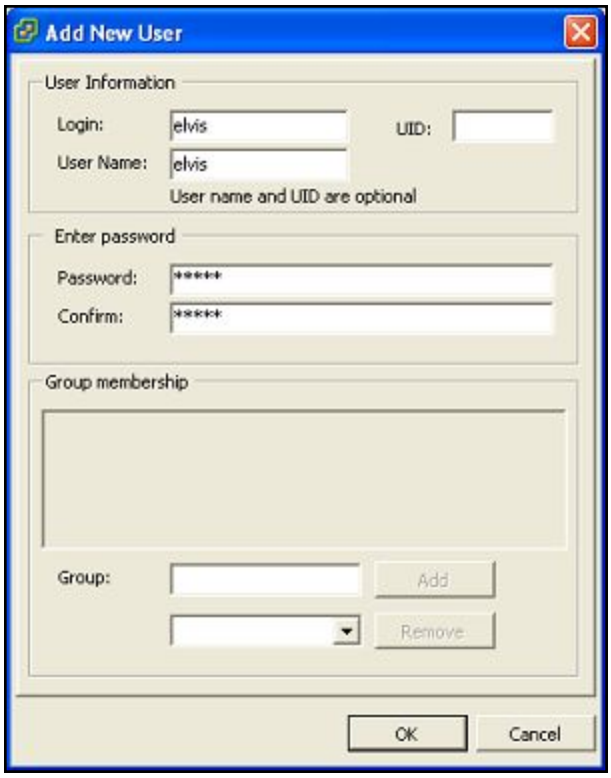


Figure 3.32: Creating a new user

12. To add the new user details, click the **OK** button in Figure 3.32.
13. Next, to assign **Diagnostics** permission to the newly created user, click on the **Permissions** tab page in the right panel of Figure 3.32. Figure 3.33 then appears.

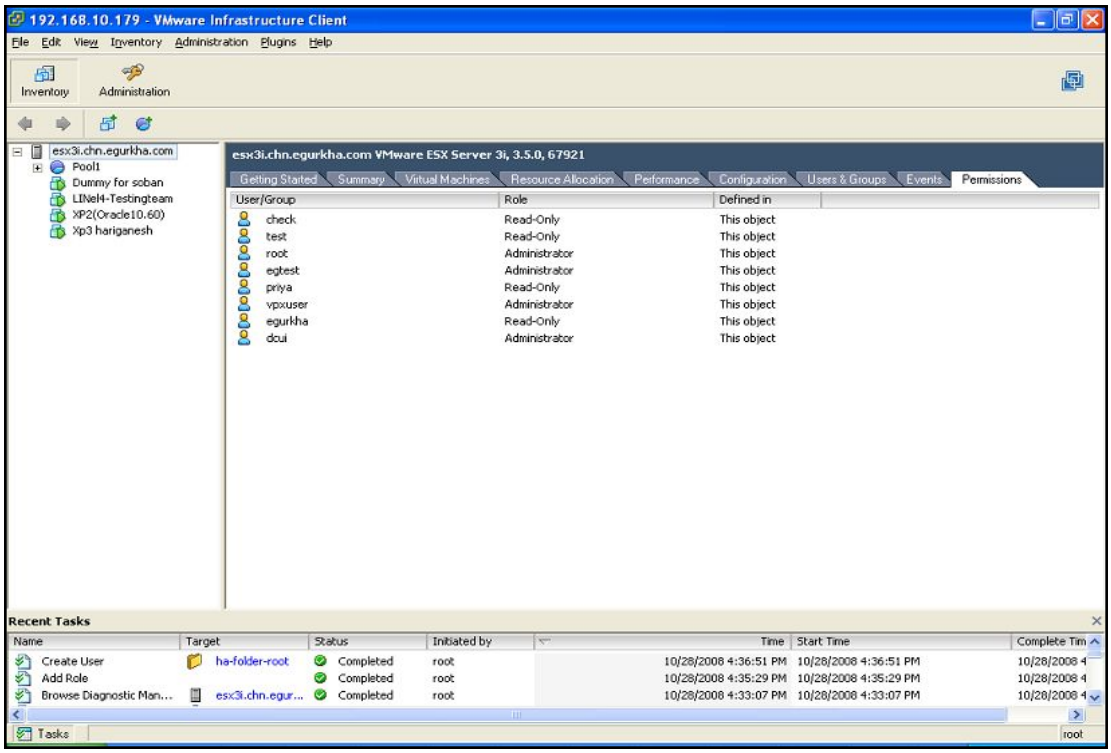


Figure 3.33: The Permissions tab page

14. To map the new role that we created with the new user, first, right-click anywhere within the **Permissions** tab page of Figure 3.33, and then pick the **Add Permission** option from the shortcut menu (see Figure 3.34).

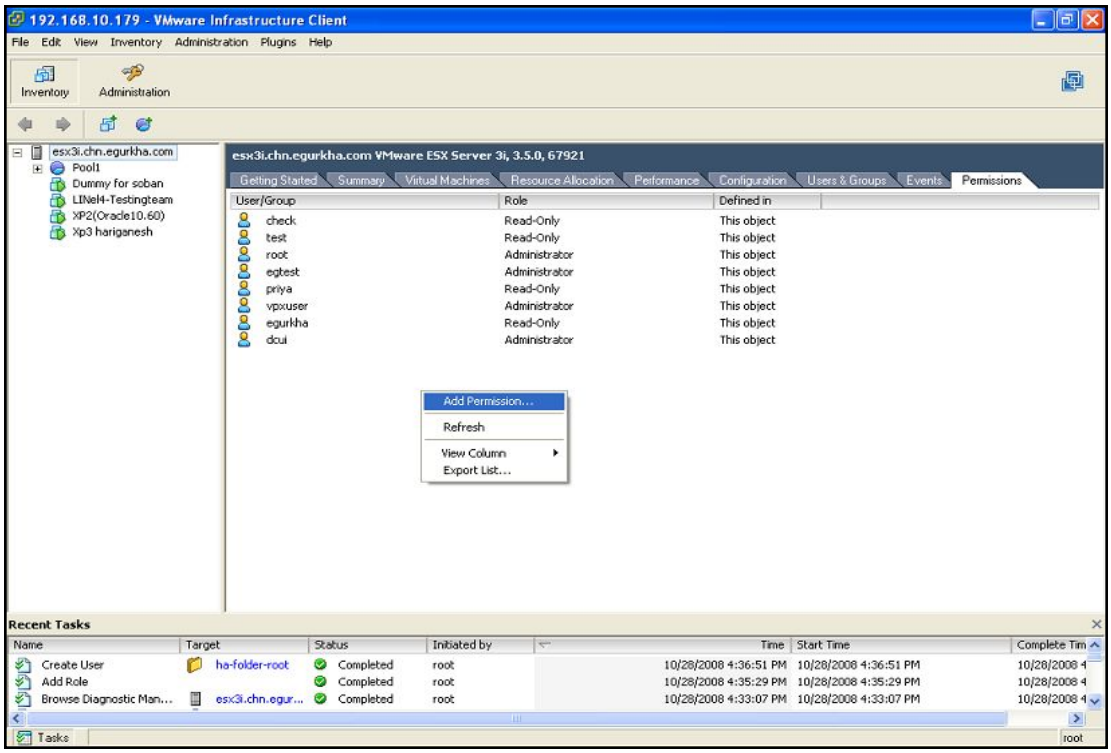


Figure 3.34: Adding a permission

15. From the **Assigned Role** section of Figure 3.35 that appears, select the newly created role. Since no users have been assigned this role yet, the **Users and Groups** section of Figure 3.35 will appear empty. To map this role to the new user, click the **Add** button in Figure 3.35.

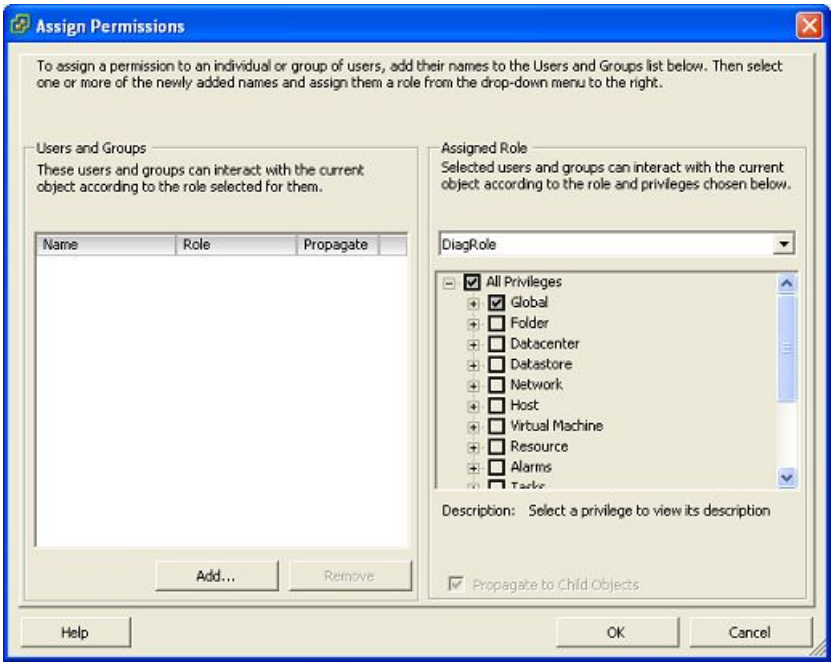


Figure 3.35: Selecting the role to be assigned to the new user

16. Figure 3.36 that then appears allows you to select a local/domain user. Since the user we created is local to the ESX server, just select **<servers>** from the **Domain** list. All valid users to the ESX server will then be listed in the **Users and Groups** section in Figure 3.36. Select the one that you just created from this list (see Figure 3.36).

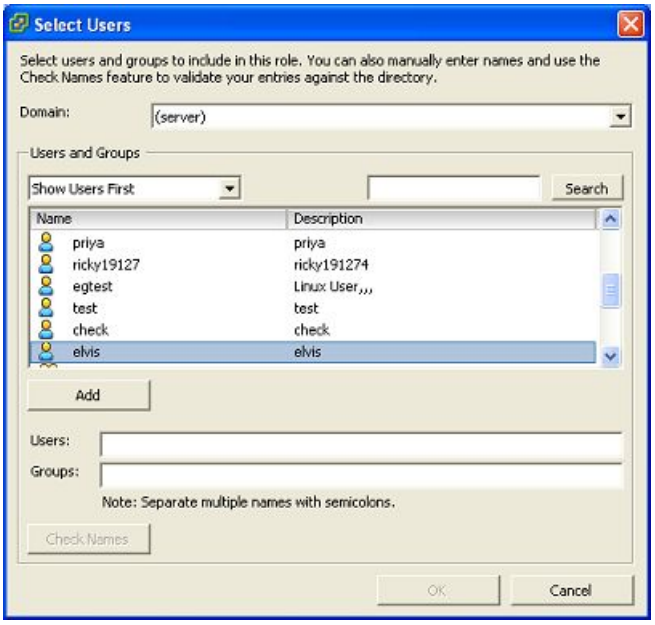


Figure 3.36: Selecting the newly created user

17. Then, click the **Add** button in Figure 3.36. The chosen user will then be added to the **Users** box as depicted by Figure 3.37.

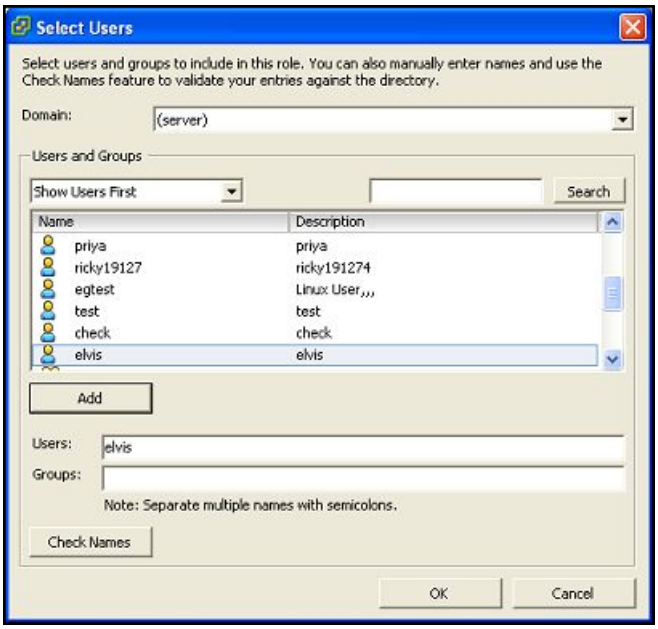


Figure 3.37: Adding the chosen user

18. Next, click the **OK** button in Figure 3.37. Figure 3.38 then appears displaying the user that you added previously, in the **Users and Groups** section.

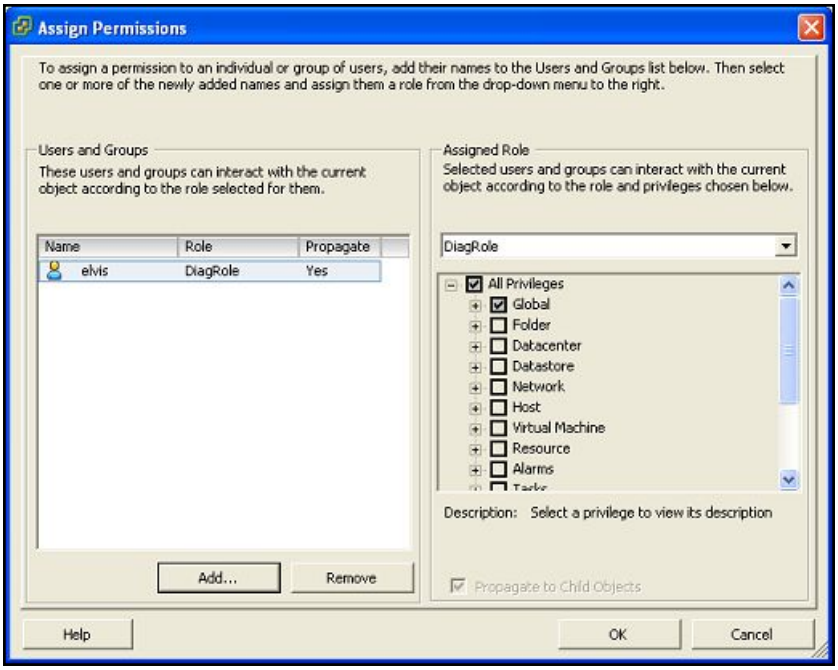


Figure 3.38: Assigning the Diagnostics role to a chosen user

19. Finally, click the **OK** button.

3.5.2.2 Configuring a Special Role on vCenter for Log File Monitoring, and Assigning the Role to a Local/Domain User

If the eG agent has been configured to connect to vCenter for monitoring the ESX log files - i.e., if the **VCENTER** parameter of the log tests is set to the IP address of vCenter - then, to obtain a remote connection to vCenter, the eG agent requires **Administrator** privileges to vCenter. In other words, the **ESX USER** and **ESX PASSWORD** parameters of the log test should contain the login credentials of a user with **Administrator** rights to vCenter. However, if, owing to security constraints, you cannot expose the credentials of the **Administrator**, then you can create a special role for this purpose in vCenter, and grant the **Diagnostics** permission to this role. You can then assign the newly created role to a local/domain user on vCenter. The procedure for achieving this is detailed below:

1. Login to a system on which the **VMware Infrastructure Client** is installed.
2. Double-click on the **VMware Infrastructure Client** icon on your desktop.
3. Figure 3.39 then appears. To connect to the vCenter, select the **IP address / Name** of the vCenter, and then provide the login information. To grant access permissions to a user, you will have to login to vCenter as an *administrator*. Therefore, provide the **User name** and **Password** of the administrator in Figure 3.39.



Figure 3.39: Connecting to vCenter

4. Figure 3.40 then appears. Click on the **Administration** icon indicated by Figure 3.40.

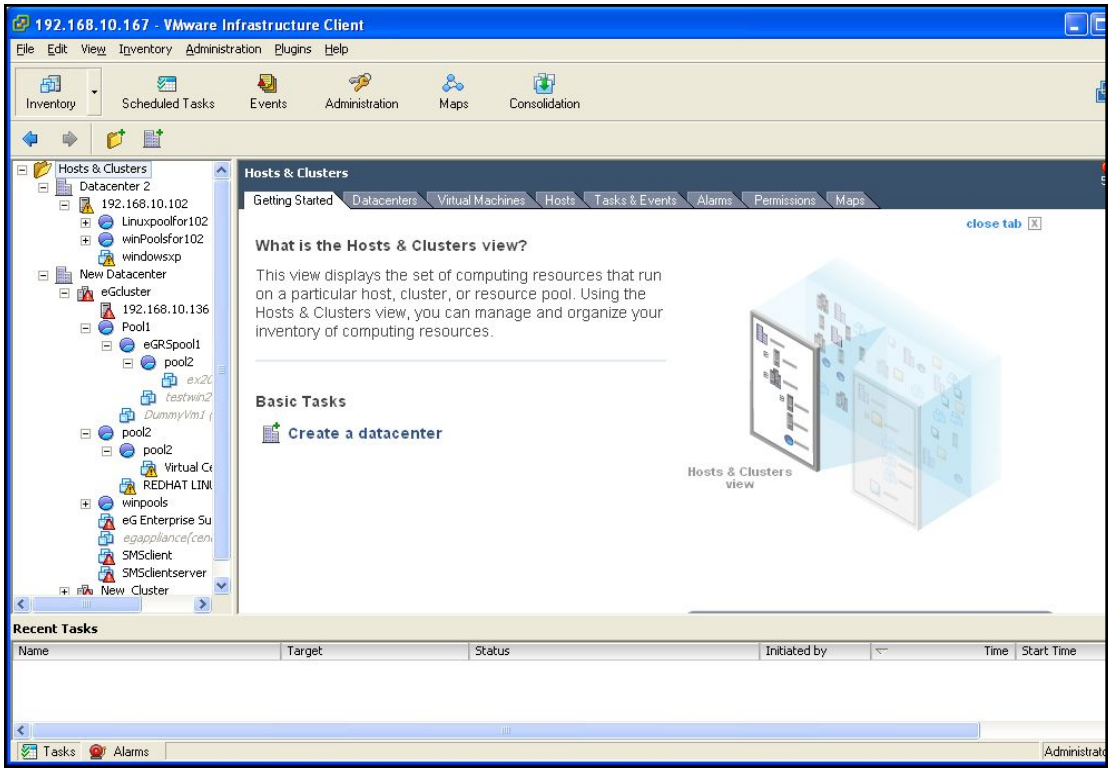


Figure 3.40: The VMware Infrastructure Client console

5. Doing so opens the **Roles** tab page by default. This lists all the existing user roles on vCenter. To create a new role, click on the **Add Role** button indicated by Figure 3.41.

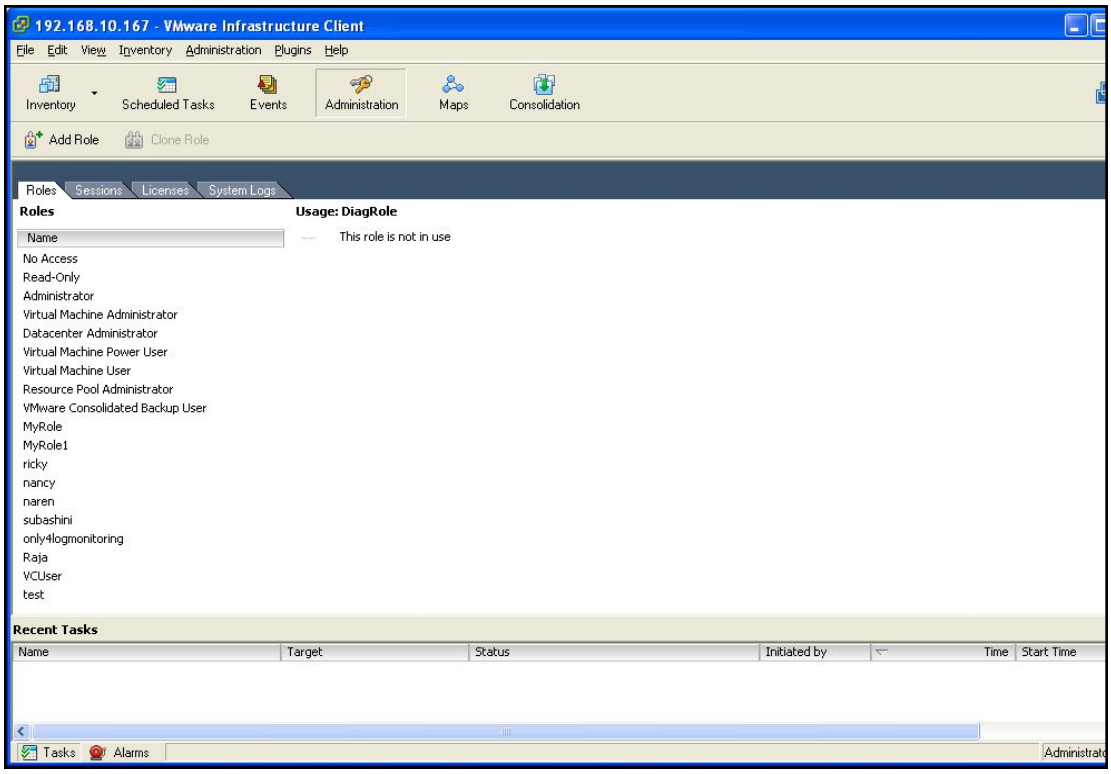


Figure 3.41: Adding a new role

6. In Figure 3.42 that then appears, **Enter Name** of the new role. Then, grant permissions to the new role by first expanding the **Global** node of the tree structure in the **Privileges** section of Figure 3.42. To grant the **Diagnostics** permission to the new role, select the check box corresponding to **Diagnostics** in Figure 3.42. Then, click the **OK** button.

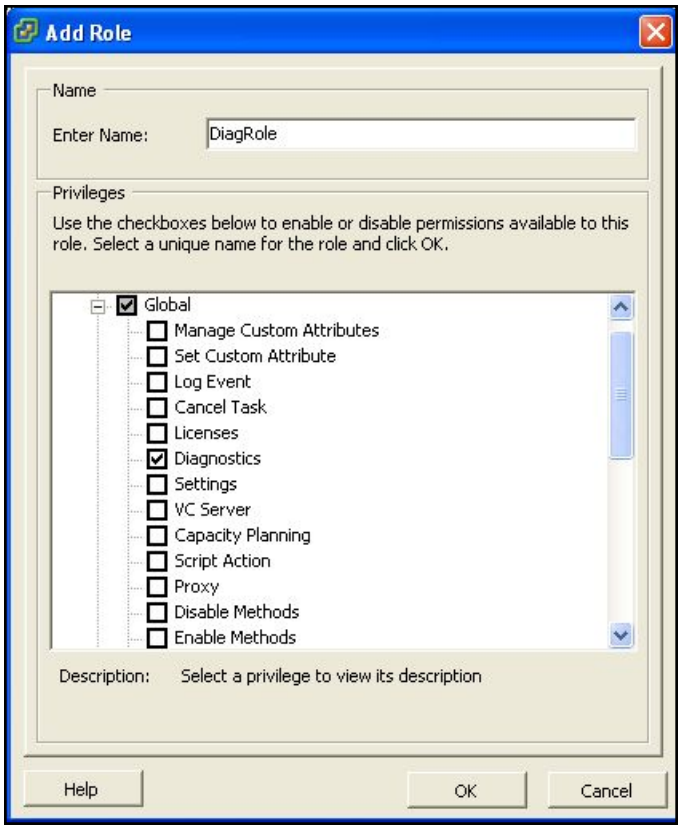


Figure 3.42: Providing the details of the new role

7. You will then return to Figure 3.40. Next, click on the **Inventory** icon indicated by Figure 3.43 to switch to Figure 3.44. Click on the **Hosts & Clusters** node in the tree-structure in the left panel of Figure 3.44, and then click on the **Permissions** tab page in the right panel. This tab page lists the local/domain users to vCenter and the roles currently assigned to them.

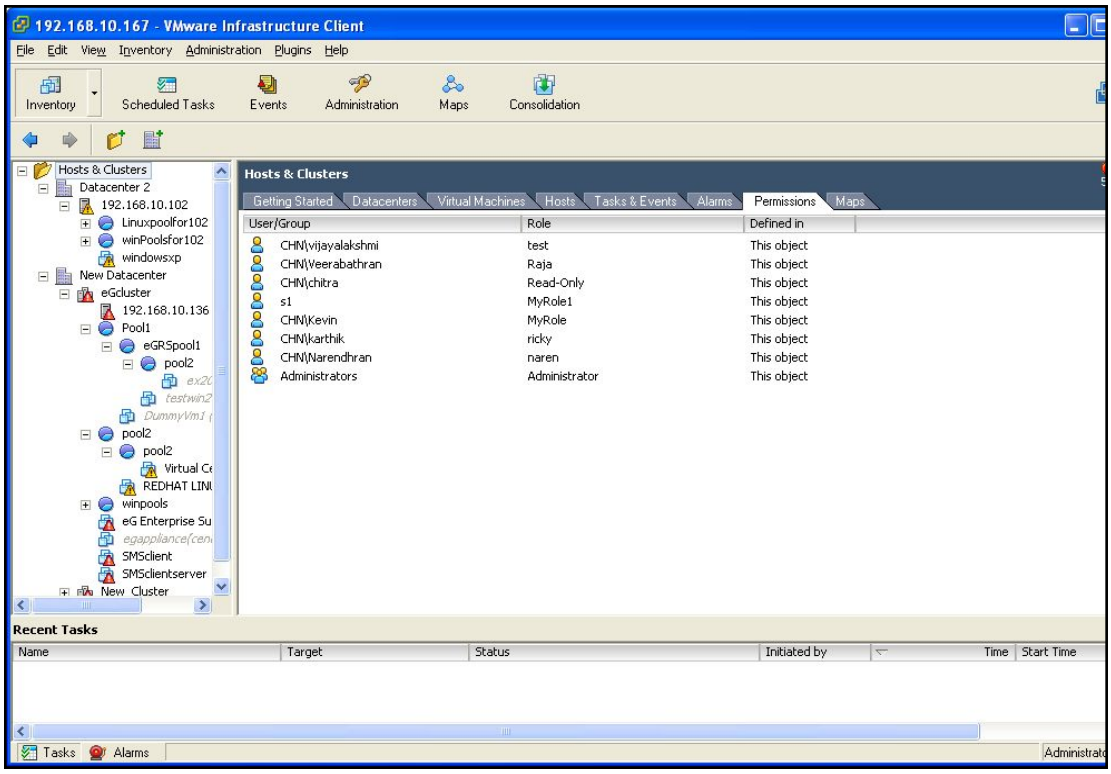


Figure 3.43: The Permissions tab page

8. To assign the newly created role to one of the listed users, right-click anywhere in the empty, white space in the right panel, and select the **Add Permission** option from the shortcut menu that appears.

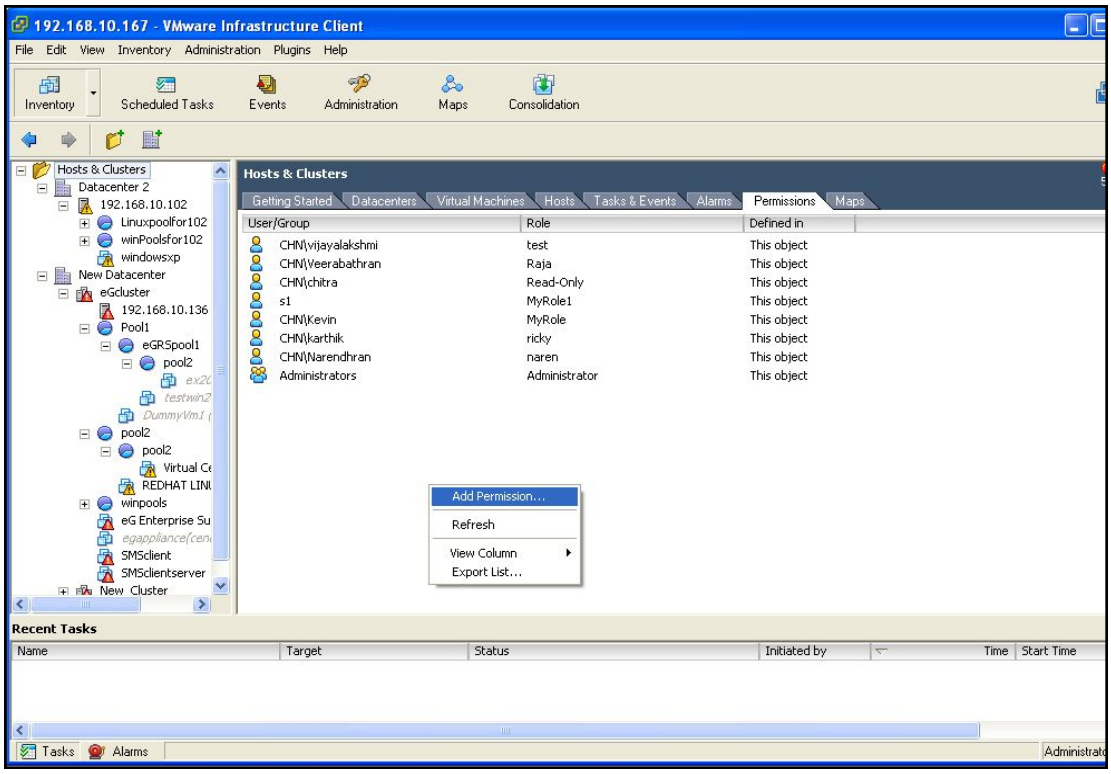


Figure 3.44: Selecting the Add Permission option

9. Figure 3.45 then appears. To choose the user who is to be assigned the new role, click the **Add** button in Figure 3.45.

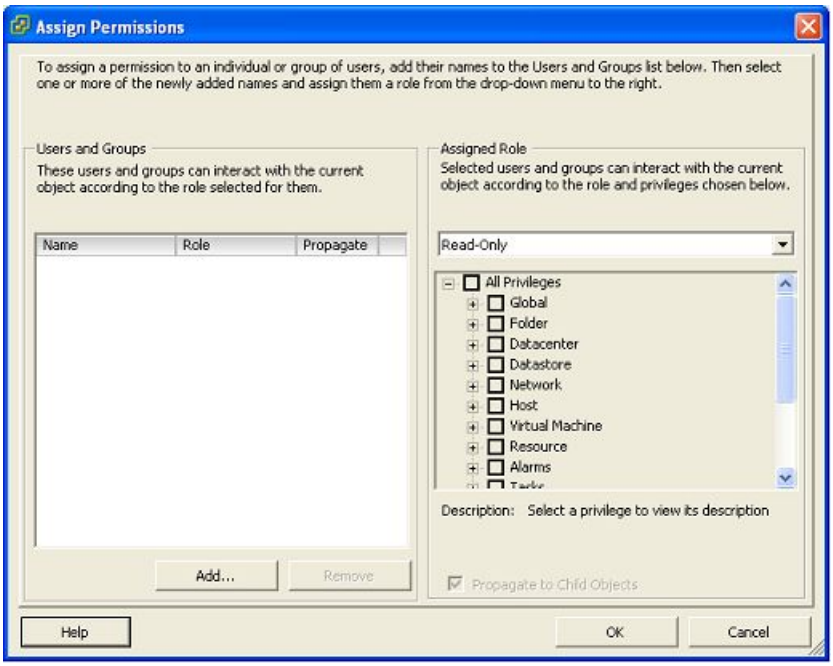


Figure 3.45: Adding a user

10. This will open Figure 3.46. If you want to assign the new role to a domain user, select the **Domain**, and then select the user from the **Users and Groups** list. If you want to assign the new role to a local user, select *(server)* from the **Domain** list, and then select a user from the **Users and Groups** list.
11. Either way, to add the user, click on the **Add** button in Figure 3.46. This will add the chosen user to the **Users** box. Then, click the **OK** button in Figure 3.46.

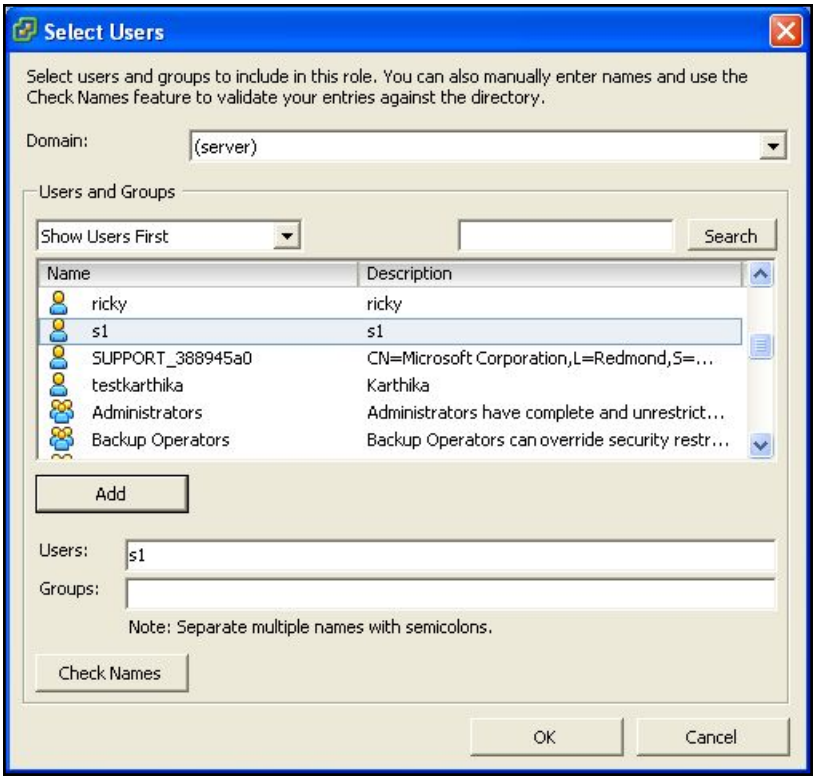


Figure 3.46: Selecting a local/domain user

12. When you return to Figure 3.45, you will find that the local/domain user you selected from Figure 3.46 is displayed in the **Users and Groups** section (see Figure 3.47).

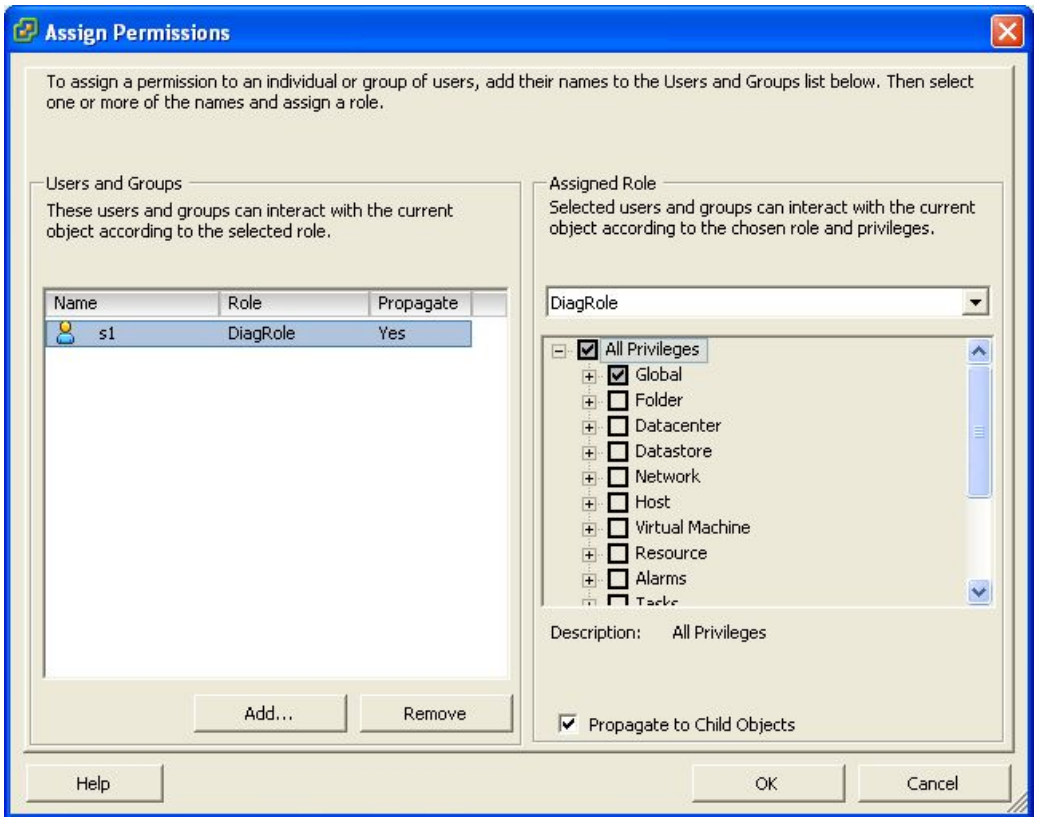


Figure 3.47: Assign the newly created role to the user

13. To assign the newly created role to this user, select the user displayed in the **Users and Groups** section in Figure 3.47, pick the newly created role from the **Assigned Role** section, and then, click the **OK** button in Figure 3.47.
14. Figure 3.48 then appears displaying the user-role mapping

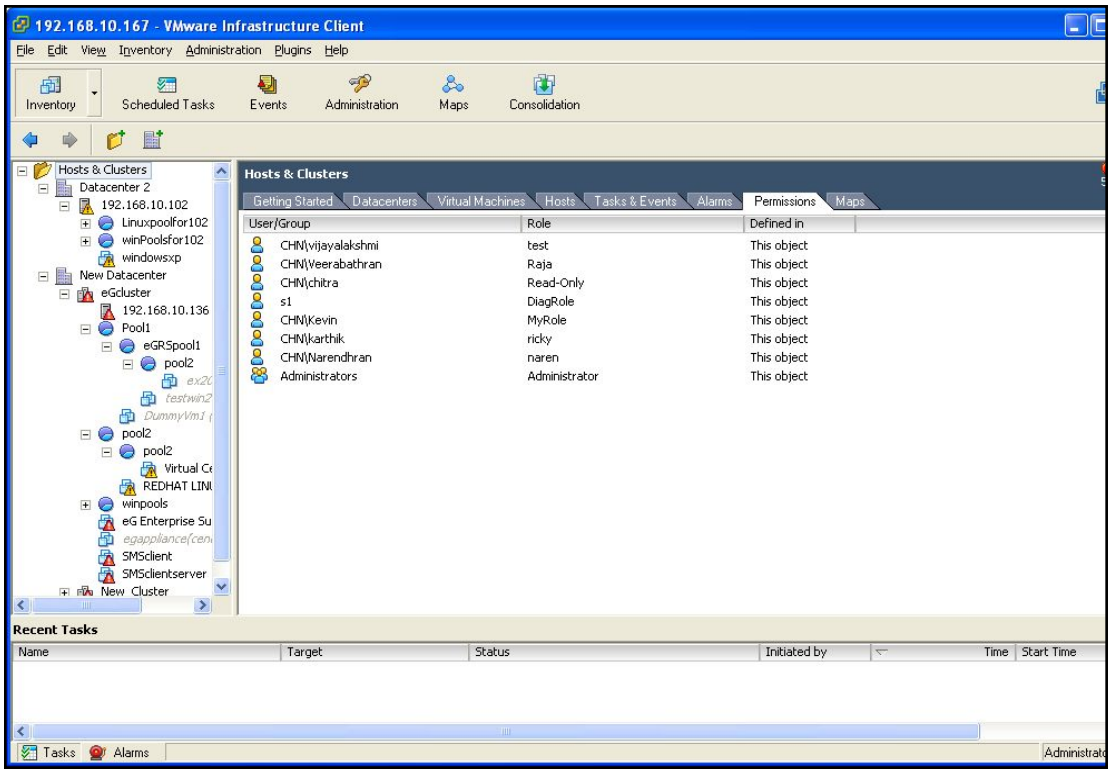


Figure 3.48: The Permissions tab page displaying the chosen user and the role assigned to him/her

3.5.3 VmKernel Log Monitor Test

This test monitors the vmkernel log file of a VMware ESX server to which the activities of the ESX server and its virtual machines are logged. The test scans the log file for entries matching configured patterns, and alerts administrators whenever such entries are logged in the log file.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every log file pattern configured for monitoring

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - To enable this test to read the log file and report metrics, the test should be configured with an **ESX USER** and **ESX PASSWORD**.

   If the metrics are to be collected by directly connecting to the ESX server in question (i.e., if the **VIRTUAL CENTER** parameter is set to none), then a user with Administrator privileges to the target ESX

server can be set as the **ESX USER**. However, if, owing to security constraints, you cannot expose the credentials of the Administrator, then you can create a special role for this purpose in the ESX server, and grant the Diagnostics permission to this role. You can then add a new user to the ESX server, and assign the newly created role to that user. The detailed procedure for accomplishing this task is available inSection **3.5.3** .

On the other hand, if the test needs to connect to vCenter for collecting the log-file related metrics (i.e., if the IP address of vCenter is specified against the VIRTUAL CENTER parameter), then a user with Administrator privileges to vCenter must be set as the ESX USER. Here again, if you cannot expose the credentials of the Administrator due to security restrictions imposed by your environment, then, you can create a special role in vCenter, grant Diagnostics permission to this role, and then assign this role to a local/domain user to vCenter. The credentials of this user can then be passed to the ESX USER and ESX PASSWORD parameters. The steps for creating such a role in vCenter and assigning the role to a local/domain user are documented in Section **3.5.3** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via **HTTPS** by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **VMKERNEL PATTERNS** - In the **VMKERNEL PATTERNS** text box, specify the pattern to search for in the log file in the following format: <*PatternName*>:<*Pattern*>, where <*PatternName*> is the pattern name that will be displayed in the monitor interface and <*Pattern*> is an expression of the form - **expr** or expr or **expr or expr**, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.

   For example, say you specify **VM:VM-*** in the **VMKERNEL PATTERNS** text box. This indicates that "**VM**"is the pattern name that will be displayed as a descriptor of this test in the monitor interface. "**VM-***" indicates that the test will monitor only those lines in the log file which start with the term "**VM-**". Similarly, if your pattern specification reads: **offline:*offline**, then it means that the pattern name is **offline** and that the test will monitor those lines in the log which end with the term **offline**.

   Multiple search patterns can be specified as a comma-separated list. For example: **VM:VM-*,offline:*offline*,online:*online**. If you want to monitor all the entries in the log file, your specification would be *all:all*.

10. **IGNORE VMKERNEL LINES** - In the **IGNORE VMKERNEL LINES** text box, specify the pattern of entries that is to be excluded from monitoring. For example, if you want the test to disregard those entries which begin with the term **VM**, then specify **VM*** in the **IGNORE VMKERNEL LINES** text box. Multiple search patterns can be specified as a comma-separated list. For example, **VM*,*VM**. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. If no lines are to be ignored, then enter *none* here. If you want to ignore all the lines in the log file, your specification would be *all:all*.

11. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

12. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

- The eG manager license should allow the detailed diagnosis capability
- Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New entries:	Indicates the number of new entries in the vmkernel log containing the configured patterns.	Number	The detailed diagnosis of this measure, if enabled, provides the details of these entries.

3.5.4 VmKWarning Log Monitor Test

This test monitors the vmkwarning log file to which the activities related to virtual machines are logged. The test scans the log file for entries matching configured patterns, and alerts administrators whenever such entries are found in the log file.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every log file pattern configured for monitoring

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - To enable this test to read the log file and report metrics, the test should be configured with an **ESX USER** and **ESX PASSWORD**.

   If the metrics are to be collected by directly connecting to the ESX server in question (i.e., if the **VIRTUAL CENTER** parameter is set to none), then a user with Administrator privileges to the target ESX server can be set as the **ESX USER**. However, if, owing to security constraints, you cannot expose the credentials of the Administrator, then you can create a special role for this purpose in the ESX server, and grant the Diagnostics permission to this role. You can then add a new user to the ESX server, and assign the newly created role to that user. The detailed procedure for accomplishing this task is available inSection **3.5.4** .

On the other hand, if the test needs to connect to vCenter for collecting the log-file related metrics (i.e., if the IP address of vCenter is specified against the VIRTUAL CENTER parameter), then a user with Administrator privileges to vCenter must be set as the ESX USER. Here again, if you cannot expose the credentials of the Administrator due to security restrictions imposed by your environment, then, you can create a special role in vCenter, grant Diagnostics permission to this role, and then assign this role to a local/domain user to vCenter. The credentials of this user can then be passed to the ESX USER and ESX PASSWORD parameters. The steps for creating such a role in vCenter and assigning the role to a local/domain user are documented in Section **3.5.4** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via **HTTPS** by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **VMKWARNING PATTERNS** - In the **VMKWARNING PATTERNS** text box, specify the pattern to search for in the log file in the following format: *<PatternName>:<Pattern>*, where *<PatternName>* is the pattern name that will be displayed in the monitor interface and *<Pattern>* is an expression of the form - **expr or expr or expr or expr**, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.

   For example, say you specify **VM:VM-*** in the **VMKWARNING PATTERNS** text box. This indicates that "**VM**"is the pattern name that will be displayed as a descriptor of this test in the monitor interface. "**VM-***" indicates that the test will monitor only those lines in the log file which start with the term "**VM-**". Similarly, if your pattern specification reads: **offline:*offline**, then it means that the pattern name is **offline** and that the test will monitor those lines in the log which end with the term **offline**.

   Multiple search patterns can be specified as a comma-separated list. For example: **VM:VM-*,offline:*offline*,online:*online**. If you want to monitor all the entries in the log file, your specification would be *all:all*.

10. **IGNORE VMKWARNING LINES** - In the **IGNORE VMKWARNING LINES** text box, specify the pattern of entries that is to be excluded from monitoring. For example, if you want the test to disregard those entries which begin with the term **VM**, then specify **VM*** in the **IGNORE VMKWARNING LINES** text box. Multiple search patterns can be specified as a comma-separated list. For example, **VM*,*VM**. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. If no lines are to be ignored, then enter none here. If you want to ignore all the lines in the log file, your specification would be *all:all*.

11. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

12. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New entries:	Indicates the number of new entries in the vmkwarning log that match the configured patterns.	Number	The detailed diagnosis of this measure, if enabled, provides the details of these entries.

3.5.5 VmMessages Log Monitor Test

This test monitors the log file that contains general log messages used to troubleshoot issues with the virtual machines on an ESX server. The test scans the log file for entries matching configured patterns, and alerts administrators whenever such entries are found in the log file.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every log file pattern configured for monitoring

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - To enable this test to read the log file and report metrics, the test should be configured with an **ESX USER** and **ESX PASSWORD**.

   If the metrics are to be collected by directly connecting to the ESX server in question (i.e., if the **VIRTUAL CENTER** parameter is set to none), then a user with Administrator privileges to the target ESX server can be set as the **ESX USER**. However, if, owing to security constraints, you cannot expose the credentials of the Administrator, then you can create a special role for this purpose in the ESX server, and grant the Diagnostics permission to this role. You can then add a new user to the ESX server, and assign the newly created role to that user. The detailed procedure for accomplishing this task is available inSection **3.5.5** .

   On the other hand, if the test needs to connect to vCenter for collecting the log-file related metrics (i.e., if the IP address of vCenter is specified against the **VIRTUAL CENTER** parameter), then a user with Administrator privileges to vCenter must be set as the **ESX USER**. Here again, if you cannot expose the credentials of the Administrator due to security restrictions imposed by your environment, then, you can create a special role in vCenter, grant Diagnostics permission to this role, and then assign this role to a local/domain user to vCenter. The credentials of this user can then be passed to the ESX USER and ESX PASSWORD parameters. The steps for creating such a role in vCenter and assigning the role

to a local/domain user are documented in Section **3.5.5** .

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via **HTTPS** by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.
9. **MESSAGE PATTERNS** - In the **MESSAGE PATTERNS** text box, specify the pattern to search for in the log file in the following format: *<PatternName>:<Pattern>*, where *<PatternName>* is the pattern name that will be displayed in the monitor interface and <Pattern> is an expression of the form **- expr or expr**

or expr or expr, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.

For example, say you specify **VM:VM-*** in the **MESSAGE PATTERNS** text box. This indicates that "VM"is the pattern name that will be displayed as a descriptor of this test in the monitor interface. "**VM**-*" indicates that the test will monitor only those lines in the log file which start with the term "**VM-**". Similarly, if your pattern specification reads: **offline:*offline**, then it means that the pattern name is **offline** and that the test will monitor those lines in the log which end with the term **offline**.

Multiple search patterns can be specified as a comma-separated list. For example: **VM:VM-*,offline:*offline*,online:*online**. If you want to monitor all the entries in the log file, your specification would be *all:all*.

10. **IGNORE VMKWARNING LINES** - In the **IGNORE VMKWARNING LINES** text box, specify the pattern of entries that is to be excluded from monitoring. For example, if you want the test to disregard those entries which begin with the term **VM**, then specify **VM*** in the **IGNORE VMKWARNING LINES** text box. Multiple search patterns can be specified as a comma-separated list. For example, **VM*,*VM**. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. If no lines are to be ignored, then enter none here. If you want to ignore all the lines in the log file, your specification would be *all:all*.
11. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
12. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New entries:	Indicates the new entries in the log file that match the configured patterns.	Number	The detailed diagnosis of this measure, if enabled, provides the details of these entries

3.5.6 ESX Server Status Test

This test periodically checks the availability of the ESX server over the network, and also reports how quickly the server responds to requests. This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick *VMware vSphere ESX* as the **Component type**, set *Performance* as the **Test type**, choose this test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list.

Target of the test : A ESX Server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the ESX server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **REPORTMANAGERTIME** - By default, this flag is set to **Yes**, indicating that, by default, the detailed diagnosis of this test, if enabled, will report the shutdown and reboot times of the vSphere/ESX host in the manager's time zone. If this flag is set to **No**, then the shutdown and reboot times are shown in the time zone of the system where the agent is running(i.e., the system being managed for agent-based monitoring, and the system on which the remote agent is running - for agentless monitoring).
5. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.4** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called

the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section **2.6** .

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vSphere user with the **Change Settings** privilege. For that you need to create a special role on vSphere, assign the **Change Settings** privilege to that role, and then map the role with a new user on vSphere. The procedure for this is detailed in Section **2.7** .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with '*Read-only*' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for

the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

Sometimes, the VMware VI SDK may cache the hardware status metrics it collects and provide the test with the cached results. This may cause the eG agent to receive obsolete hardware status information from the SDK. This is also the reason why, you may at times notice a mismatch between the hardware status reported by the eG agent and by the vSphere client. To ensure that the eG agent always reports the current hardware status, you should configure the eG agent to obtain the hardware metrics from the VMware VI SDK only after the SDK resets the cache to clear its contents, and then refreshes the cache so that the latest hardware status information is fetched into it. To enable the eG agent to make the *reset* and *refresh* SDK calls, the **ESX USER** and **ESX PASSWORD** parameters must be configured with the credentials of a vCenter user with the **Change Settings** privilege. For that you need to create a special role on vCenter, assign the **Change Settings** privilege to that role, and then map the role with a new user on vCenter. The procedure for this is detailed in Section **2.7** .

6. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
7. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
8. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
9. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to **'none'**. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Connection status:	Indicates whether the ESX server is available or not.		If this measure reports the value *Up*, it indicates that the ESX server is available over the network. On the other hand, if the measure reports *Down*, it indicates that the ESX server is not available over the network. The numeric values that correspond to the Up/down states discussed above are available in the table below: **State** / **Value** Up / 100 Down / 0 **Note**: By default, this measure reports the value *Up* or *Down* to indicate ESX server availability. The graph of this measure however, represents the availability of an ESX server over the network using the numeric equivalents - 0 or 100.
Response time:	Indicates the time taken by the ESX server to respond to requests.	Seconds	If the value of this measure consistently increases, it indicates that the performance of the ESX server is gradually deteriorating. Ideally, the value of this measure should be low.

3.6 The Outside View of VMs Layer

This layer provides the host operating system's view of the resource usage levels of each of the virtual guests hosted on it. Using the information reported by this test, administrators can:

- Determine which of the guests is taking up more resources (CPU, memory, network, or disk) than the others. This information can help with load balancing or capacity planning. For example, if one of the guests is receiving a very high rate of requests compared to the others, this guest may be a candidate for migration to another ESX server, so as to minimize the impact it has on the other guests on the current ESX server.
- Determine times when sudden or steady spikes in the physical resource utilization are caused by the guest machines
- Track the overall status of the virtual machines - how many are registered, which ones are powered on, and at what times, etc.
- Understand how resources are shared amongst all available resource pools, and identify resource pools that have been over-utilized.

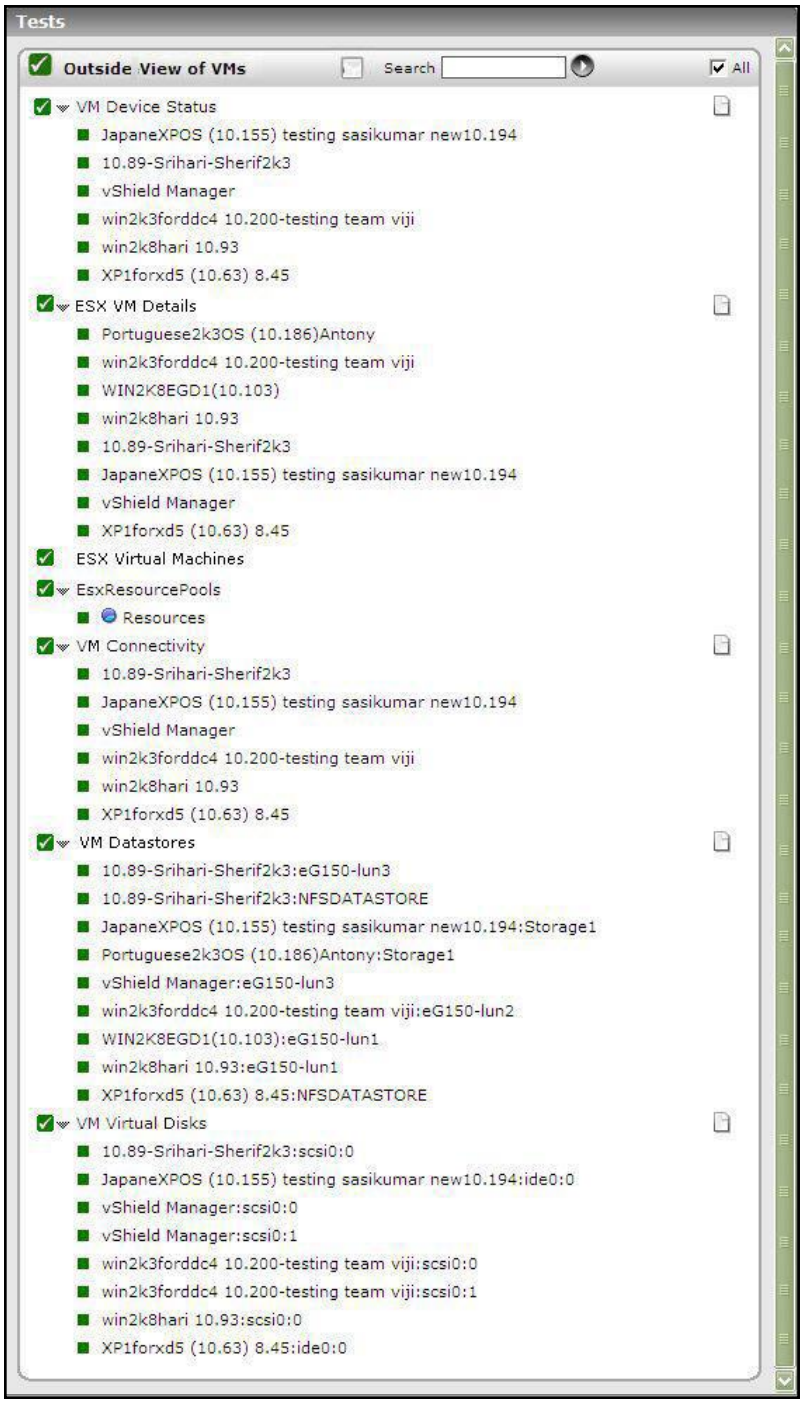


Figure 3.49: The tests associated with the Outside View of VMs layer

3.6.1 VM Details - ESX Test

This test monitors the amount of the physical server's resources that each guest on an vSphere/ESX server is taking up. Using the metrics reported by this test, administrators can determine which virtual guest is taking up most CPU, which guest is generating the most network traffic, which guest is over-utilizing memory, etc.

Note that the amount of resources taken up by a virtual guest will be limited by the resource allocations that have been made by administrators. For example, an administrator could cap the amount of memory that a specific guest may take. Also, virtual guests can be organized into resource pools, and allocation of resources can be made at the resource-pool level. In this case, virtual guests allocated to the same resource pool contend for the resources allocated to the resource pool.

Note:

Template and orphaned VMs also appear as 'descriptors' of this test; these VMs are typically tagged as *(template)* and *(orphaned)*. For such VMs, this test reports only the **Disk Capacity**.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every VM on the ESX server monitored, including template and orphaned VMs; template and orphaned VMs are always tagged as *(template)* and *(orphaned)* in the eG monitoring console.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 of this document.

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the

eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

   Note:

   While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
10. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
11. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

    If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

    On the other hand, if the IP address of the vCenter server of interest to you is not available in the list,

then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD** , and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER FLAG is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **REPORT POWERED ON** - You can set the **REPORT POWERED ON** status to **YES**, so that the test reports an additional measure, *Is VM powered on?*, revealing whether a guest OS is currently running or not. The default status of this flag is set to **Yes** for a VMware ESX server. However, the default status of this flag is **No** in the case of virtual desktop infrastructures (VDI); this is because, in such environments, the virtual desktops will be in the powered-off state most of the time.

17. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run

detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

- The eG manager license should allow the detailed diagnosis capability
- Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
VM power- on state:	Indicates the current state of the virtual machine.		The table below displays the States that can be reported by this measure, and their numeric equivalents: State: Value Off: 0 On: 1 Suspended: 2 Unknown: 3 Standby: 4 Shuttingdown: 5 Resetting: 6 Stuck: 7 **Note:** By default, this measure reports one of the States listed in the table above. The graph of this measure however will represent the VM status using the numeric equivalents - '0' to '7'.
Current sessions:	Indicates the number of sessions that are currently active on the guest.	Number	The value of this measure reveals the guest OS that is currently busy.
	Physical CPU	Percent	Ideally, a VM should use only a small

Measurement	Description	Measurement Unit	Interpretation
	utilization: Indicates the percentage of physical CPU that is used by the VM.		percentage of the physical CPU resources of the base ESX host. High CPU usage by a VM could cause other VMs to contend for limited CPU resources, thereby resulting in adversely impacting the performance of the other VMs and the applications executing on them.
Physical CPU used:	Indicates the CPU used by this guest (in Mhz)	Mhz	The percent CPU usage measure serves as an effective indicator of how resource-intensive a particular VM is on a specific ESX server. However, for performing capacity planning or what-if analysis, the CPU usage of a VM measured in absolute terms would be more useful. For instance, the physical CPU usage of a VM could be 30% on a particular ESX server - this means that the VM is consuming 30% of the total physical CPU capacity of that ESX server. If you are planning to migrate the VM to another ESX server, then it would be unwise to assume that the VM will only consume 30% of CPU on the other ESX server as well, as the percentage will vary depending upon the physical CPU resources that are available to the other ESX server. The absolute measure however, will remain unchanged across ESX servers. Therefore, to decide which ESX server the VM is to be moved to, and to analyze the impact of this movement on the CPU resources of the new ESX host, you would require an absolute measure of CPU usage.
System usage of physical CPU:	Indicates the percentage of time this guest spent in the ESX server VMKernel to process interrupts and to perform other system	Percent	An unusually high value indicates a problem and may be due to too many system-level tasks executing simultaneously.

Measurement	Description	Measurement Unit	Interpretation
	activities.		
VM CPU waits:	Indicates the percentage of time the guest spent in the wait state.	Percent	While the CPU ready metric denotes the time when the world is waiting for CPU to execute, the busy wait state represents the time when the world is waiting for some event to happen before it is ready to execute.
VM CPU extra:	Indicates the percentage of extra CPU used up by the virtual guests.	Percent	Ideally, the value of this measure should be very low. A high value indicates that one/more processes on the VM are consuming more CPU than necessary. Typically, screen savers, animations, and X servers all consume extra CPU, which can affectperformance of other virtual machines and consolidation ratios. The non- trivial consumption of CPU by idling virtual machines can also have an adverse impact on Dynamic Resource Scheduling (DRS) decisions. You might want to disable screen savers and Window animations in the virtual machine to avoid the erosion of CPU resources. On Linux, if using an X server is not required, disable it.
VM CPU ready:	Indicates the percentage of time the guest was ready to run (i.e., it had instructions to execute) but was not able to because of processor contention.	Percentage	This metric should typically be low - generally 5% or less. If a VM waits too long to run, it can significantly affect the responsiveness of the VM.
CPU guaranteed to VM:	Indicates the percentage of CPU guaranteed on the guest.	Percent	In some situations, system administrators want to know that a certain amount of memory for a virtual machine comes directly from the physical resources of the ESX Server machine. Similarly, the administrator

Measurement	Description	Measurement Unit	Interpretation
			might want to guarantee that a certain virtual machine always receives a higher percentage of the physical resources than other virtual machines. To ensure this, you can reserve physical resources of the host for specific virtual machines. Reservation specifies the guaranteed reservation for a virtual machine. The server allows you to power on a virtual machine only if the CPU and memory reservation is available. The server guarantees that amount even when the physical server is heavily loaded.
Physical CPU throttled:	Indicates the percentage of time the ESX host deliberately did not run the VM because that would violate the VM's limit setting.	Percentage	A *Limit* refers to the maximum CPU the host can make available to this VM. Even though the VM is ready to run, if it is prevented from running owing to a probable limit violation, then the value of this measure will not be included in the value of the VM CPU readymeasure. A high value for this metric indicates that the VM has been trying to get additional CPU resources but has not been able to do so because the CPU usage for this VM or its resource pool (if it is a part of a resource pool) has reached the allocated limit.
Memory overhead:	Indicates the memory overhead incurred by the virtual machine.	MB	ESX Server virtual machines can incur two kinds of memory overhead: • The additional time to access memory within a virtual machine. • The extra space needed by the ESX Server host for its own code and data structures, beyond the memory allocated to each virtual machine. The ESX server's memory virtualization ensures that the time overhead to memory accesses is minimal.

Measurement	Description	Measurement Unit	Interpretation
			Addition of more virtual CPUs, allocation of more memory to the guests, and choosing to configure 32-bit guest operating systems instead of 64-bit ones, can go a long way in reducing this overhead. The ESX server also provides optimizations such as memory sharing to save up memory.
Memory swapped out:	Indicates the rate at which memory is being swapped to disk by the ESX Server.	MB	ESX Server hosts use swapping to forcibly reclaim memory from a virtual machine when no vmmemctl driver is available because the vmmemctl driver: • Was never installed • Has been explicitly disabled • Is not running (for example, while the guest operating system is booting) • Is temporarily unable to reclaim memory quickly enough to satisfy current system demands Standard demand-paging techniques swap pages back in when the virtual machine needs them. Swap space must be reserved on disk for any unreserved virtual machine memory. This swap reservation is required to ensure the system is able to preserve virtual machine memory under any circumstances. In practice, only a small fraction of the swap space may actually be used. Typically, swap space usage for each VM should be low. Since access from RAM is much faster than access from physical disk, excessive usage of swap memory will slow down the performance of a VM. Watch for VMs that are seeing higher swap usage and more swap reads and writes.
Current swap memory:	Indicates the amount of memory that is currently swapped.	MB	
Memory swapped in:	Indicates the amount of memory that is being swapped in by the ESX Server from the disk for the guest.	MB	
Swap memory target:Swap memory target:	Indicates the amount of memory that can be swapped.	MB	

Measurement	Description	Measurement Unit	Interpretation
Physical memory consumed:	Indicates the amount of physical memory consumed by the guest operating system.	MB	High memory consumption over long periods can deplete the free memory on the host, causing prolonged delays in the execution of the virtual machines. Comparison of the active memory usage across guests indicates the guest(s) that could be causing a memory bottleneck on the host.
Shared memory:	Indicates the current amount of shared guest operating system memory.	MB	VMware ESX can share common memory pages across VMs. This includes pages from VMs running the same virtual machine OS and applications.
Balloon memory:	Indicates the total amount of physical memory currently reclaimed from this guest using the **vmmemctl** modules.	MB	If the virtual machine's memory usage approaches its memory target, ESX uses ballooning to reduce that virtual machine's memory demands. Using a VMware-supplied **vmmemctl** module installed in the guest operating system as part of VMware Tools suite, ESX can cause the guest to relinquish the memory pages it considers least valuable. The **vmmectl** driver emulates an increase or decrease in memory pressure on the guest operating system; this way, it forces the guest OS to place memory pages into its local swap file. This driver differs from the VMware swap file method as it forces the operating system to determine what memory it wishes to page. Once the memory is paged locally on the guest operating system, the free physical pages of memory may be reallocated to other guests. As the ESX host sees that memory demand has been reduced, it will instruct **vmmemctl** to reduce pressure on the guest OS to page memory. To use ballooning, the guest operating system must be configured with sufficient swap space. The maximum amount of memory that can be

Measurement	Description	Measurement Unit	Interpretation
			reclaimed from a guest may be configured by modifying the "sched.mem.maxmemctl" advanced option.
Balloon memory target:	Indicates the total amount of physical memory the ESX Server host attempts to reclaim using the vmmemctl module.	MB	If you suspect that memory overcommitment is beginning to affect the performance of a virtual machine, you can look at the value of Memory Balloon (Average) in the vSphere Client Performance Chart. An absence of ballooning suggests that ESX is not under heavy memory pressure and thus memory overcommitment is not affecting performance. (Note that some ballooning is quite normal and not indicative of a problem.)
Zero memory:	Indicates the amount of memory that is zeroed out.	MB	This counter is helpful in identifying virtual machines that may have been configured with more RAM than necessary or are not presently using provisioned memory. The "Memory Zero" amount will fluctuate as memory is over allocated. ESX will zero out the VM's memory to use with other VMs.
Memory granted:	Indicates the amount of memory that the Vmkernel has granted to the virtual machine.	MB	
Active memory:	Indicates the amount of memory that is actively used by the guest.	MB	
Configured memory:	Indicates the amount of configured memory.	MB	
Active memory usage:	Indicates the percentage of memory allocated to the guest that is currently in use.	Percent	High memory consumption over long periods can deplete the free memory on the guest, causing prolonged delays in the execution of the application (s) hosted by the guests.

Measurement	Description	Measurement Unit	Interpretation
			Comparison of the memory usage across guests indicates the guest(s) that could be causing a memory bottleneck on the host.
Memory swap in rate:	Indicates the rate at which memory is swapped from disk into active memory.	Mbps	A high rate of swap ins and swap outs could be indicative of a memory contention.
Memory swap out rate:	Indicates the rate at which memory is swapped from disk into active memory.	Mbps	
Disk reads:	Indicates the rate at which read commands were issued to the storage adapters of the guest.	Commands/Sec	
Disk writes:	Indicates the rate at which write commands were issued to the storage adapters of the guest.	Commands/Sec	
Total IOPS:	Indicates the number of commands issued per second.	Commands/Sec	This measure is a good indicator of the level of activity on the storage adapters of the guest.
Disk commands aborted:	Indicates the rate at which commands aborted.	Commands/Sec	A high rate is indicative of a problem condition that requires investigation.
Data writes to disk:	Indicates the rate at which data was written to the disk.	MB/Sec	
Data reads from	Indicates the rate at	MB/Sec	

Measurement	Description	Measurement Unit	Interpretation
disk:	which data was read from the disk.		
Disk capacity:	Indicates the total disk capacity of the virtual machine.	MB	One of the key problems faced by VMware administrators is VM sprawl. Since VMs are easy to create and deploy, many a time an administrator might be faced with scenarios where many VMs are created on an ESX server, but very few are actively used. A VM, whether powered on or off, consumes disk space on an ESX server. When the ESX server hosting the VMs runs low on disk space, administrators might want to know which VM is taking up maximum disk space. This measure reveals the disk capacity of a VM, regardless of its on/off state. A quick comparison of the capacity across VMs can enable administrators to accurately identify the VM that is taking up maximum disk space. **Note:** For template and orphaned VMs, the Esx VM Details test reports only this measure. By comparing the disk capacity of template VMs, you can accurately identify which template VM is occupying too much disk space on the ESX host. Based on these findings, you might want to remove one/more unwanted or unused templates from the ESX host, so as to free space on the host.
Network packets transmitted:	Indicates the number of packets the guest transmitted per second.	Packets/Sec	
Network data transmitted:	Indicates the amount of data the guest transmitted per second.	Mbps	Comparing the data transmitted across all the virtual guests provides an indicator of the guest that is generating most out- bound network traffic.

Measurement	Description	Measurement Unit	Interpretation
Network packets received:	Indicates the number of packets received per second by the guest.	Packets/Sec	
Network data received:	Indicates the rate at which data was received by the guest.	Mbps	Comparing the data received across all the virtual guests provides an indicator of the guest that has the most in-bound network traffic.
Virtual CPU utilization:	Indicates the percentage of allocated CPU resources that this VM is currently using.	Percent	Comparing the value of this measure across VMs will enable you to accurately identify the VMs on which CPU-intensive applications are executing. **Note:** If multiple vCPUs have been allocated to a VM, then the Virtual CPU utilization of that VM will be the aggregate of the usage of all vCPUs associated with that VM.
Memory limit:	Indicates the number of storage paths through which the host communicates with this LUN.	MB	
Is memory configured correctly?	Indicates whether the memory is configured correctly or not for this VM.		If the configured memory is greater than the memory limit reported by the Memory limited measure, then this measure will report the value *No*. If the configured memory is equal to the memory limit reported by the Memory limited measure or, if the memory limit is set to unlimited, then this measure will report the value *Yes*. The numeric values that correspond to the measure values discussed above are listed in the table below:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Numeric Value</th><th>Measure Value</th></tr>
<tr><td>0</td><td>Yes</td></tr>
<tr><td>1</td><td>No</td></tr>
</table>
Note:
By default, this measure reports the values Yes or No only to indicate whether the memory is configured correctly or not. The graph of this measure however, will represent the memory state using the numeric equivalents - 0 or 1.
</td></tr>
<tr><td>Disk throughput:</td><td>Indicates the rate of data reads and writes to the disk of this VM.</td><td>MB/Sec</td><td>This is the sum of the Data reads from disk and Data writes to disk measures. This is a good indicator of the level of disk I/O activity on a VM. A very low value or a consistent decrease in the value of this measure could indicate a current/potential I/O processing bottleneck. You can compare the value of this measure across VMs, to know which VM is the slowest in terms of the speed with which it processes I/O requests.</td></tr>
<tr><td>Total network I/O operations:</td><td>Indicates the rate at which data was received and sent across the network by this VM.</td><td>Mbps</td><td>This is the sum of the Network data transmitted and Network data received measures. This is a good measure of the level of network I/O activity on a VM. A very low value or a consistent decrease in the value of this measure could indicate a current/potential network I/O processing bottleneck. You can compare the value of this measure across VMs, to know which VM is the slowest in terms of the speed with which it processes network I/O requests.</td></tr>
<tr><td>Connection status:</td><td>Indicates the current status of the vCenter server’s connection to this virtual machine.</td><td></td><td>The table below lists the values that this measure can take and briefly describes each value:</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Description</th></tr>
<tr><td>Connected</td><td>The vCenter server has access to the virtual machine.</td></tr>
<tr><td>Disconnected</td><td>The vCenter server is currently disconnected from the virtual machine, since its host is disconnected.</td></tr>
<tr><td>Inaccessible</td><td>One or more of the virtual machine configuration files are inaccessible. For example, this can be due to transient disk failures. In this case, no configuration can be returned for a virtual machine.</td></tr>
<tr><td>Invalid</td><td>The virtual machine configuration format is invalid. Thus, it is accessible on disk, but corrupted in a way that does not allow the server to read the content. In this case, no configuration can be returned for a virtual machine.</td></tr>
<tr><td>Orphaned</td><td>The virtual machine is no longer registered on the host it is associated with. For example, a virtual machine that is unregistered or deleted directly on a host managed by vCenter shows up in this state.</td></tr>
</table>

The numeric values that correspond to these measure values are as follows:
<table>
<tr><th>Numeric Value</th><th>Measure Value</th></tr>
<tr><td>0</td><td>Orphaned</td></tr>
<tr><td>1</td><td>Connected</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Numeric Value</th><th>Measure Value</th></tr>
<tr><td>2</td><td>Disconnected</td></tr>
<tr><td>3</td><td>Inaccessible</td></tr>
<tr><td>4</td><td>Invalid</td></tr>
</table>
Note:
By default, this measure reports the Measure Values to indicate the connection status. In the graph of this measure however, status will be represented using the numeric equivalents - 0 through 4.
</td></tr>
<tr><td>Time in the COSTOP state:</td><td>Indicates the percentage of time that this virtual machine was ready to execute commands but is waiting for the availability of multiple vCPUs as the virtual machine is configured to use multiple vCPUs.</td><td>Percent</td><td>If a virtual machine is unresponsive and the value of this measure is proportionally high when compared to the value of the VM CPU ready measure. it may indicate that the VM has limited CPU resources to simultaneously co- schedule all vCPUs in the virtual machines. If this value is low, then any performance problems should be attributed to other issues and not to the co-scheduling of the vCPU.</td></tr>
<tr><td>VM CPU overlap:</td><td>Indicates the percentage of time the virtual machine was interrupted to perform system services on behalf of that virtual machine or other virtual machines.</td><td>Percent</td><td>If the value of the Virtual CPU utilization measure is unusually high for a VM, you may want to check the value of the VM CPU overlap measure to figure out whether it is owing to system services performed by the virtual machine.</td></tr>
<tr><td>Time in RUN state:</td><td>Indicates the percentage of time the virtual machine is consuming CPU resources.</td><td>Percent</td><td>This value represents the percentage of absolute time the virtual machine was running on the system.
If the value is near ‘zero’, it could mean that the VM is idle, blocked on an operation, or is not scheduled due to resource contention.</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			If it is near 100%, it means that all vCPUs in the virtual machine are busy. This is an indicator that the guest operating system may be stuck in a operational loop.
Compressed memory consumed by VM:	Indicates the amount of compressed memory currently consumed by this VM.	MB	vSphere provides a memory compression cache to improve virtual machine performance when you use memory overcommitment. Memory compression is enabled by default. When a host's memory becomes overcommitted, vSphere compresses virtual pages and stores them in memory. Because accessing compressed memory is faster than accessing memory that is swapped to disk, memory compression in vSphere allows you to overcommit memory without significantly hindering performance. When a virtual page needs to be swapped, vSphere first attempts to compress the page. Pages that can be compressed to 2 KB or smaller are stored in the virtual machine's compression cache, increasing the capacity of the host. The value of this measure indicates the amount of memory consumed by the compressed pages in the compression cache of a VM. Ideally, this value should be much less than the maximum size upto which the compression cache has been configured to grow. If the value of this measure is close or equal to the maximum size configuration of the compression cache, then it means that the compression cache is filling up rapidly, and will soon run out of memory. When this happens, a lot of replaced compressed pages will be decompressed and swapped out to accommodate more compressed pages in the

Measurement	Description	Measurement Unit	Interpretation
			cache. Any following swapins of those pages will hurt VM performance. This can happen if the maximum size of the compression cache is set too low. At the same time, a very large compression cache may also waste VM memory and unnecessarily create host memory pressure, especially when most compressed pages would not be touched in the future. By prudently setting the maximum size of the compression cache, you can ensure optimal usage of the cache and can avoid the eventualities discussed above.
Distributed CPU entitlement:	Indicates the amount of CPU resources, in MHz, that this VM is entitled to, as calculated by DRS (Distributed Resource Scheduler).	MHz	This measure will report a value only for those VMs that are managed by DRS. VMware DRS dynamically balances computing capacity across a collection of hardware resources aggregated into logical resource pools, continuously monitoring utilization across resource pools and intelligently allocating available resources among the virtual machines based on pre-defined rules that reflect business needs and changing priorities. When a virtual machine experiences an increased load, VMware DRS automatically allocates additional resources by redistributing virtual machines among the physical servers in the resource pool. By comparing the value of this measure across VMs, you can identify those VMs that DRS has marked as CPU-intensive, and has hence allocated more CPU resources.
Distributed memory entitlement:	Indicates the amount of memory, in MB, that this VM is entitled to, as calculated by DRS	MB	This measure will report a value only for those VMs that are managed by DRS. VMware DRS dynamically balances

Measurement	Description	Measurement Unit	Interpretation
	(Distributed Resource Scheduler).		computing capacity across a collection a logical resource pools, continuously monitoring utilization across resource pools and intelligently allocating available resources among the virtual machines based on pre-defined rules that reflect business needs and changing priorities. When a virtual machine experiences an increased load, VMware DRS automatically allocates additional resources by redistributing virtual machines among the physical servers in the resource pool. By comparing the value of this measure across VMs, you can identify those VMs that DRS has marked as memory-intensive, and has hence allocated more memory resources.
Overall CPU demand:	Indicates the CPU usage of this VM, while it is running.	MHz	By comparing the value of this measure across VMs, you can identify those VMs that require more CPU resources than the rest. Based on the insights provided by this measure, you can configure DRS to automatically reserve the highest priority in CPU resource allocation for the CPU-hungry VMs.
Private memory:	Indicates the portion of memory that is granted to this VM from non-shared host memory.	MB	Private memory is the amount of memory that is actually stored in the physical memory of the vSphere host, for a virtual machine. In other words, this is the memory that is not shared, but is backed by the host memory. Ideally, the value of this measure should be low. Any increase in the value of this measure for a VM, will automatically increase the value of the Physical memory consumed measure of that VM. This is because: Private memory + Memory overhead = Physical memory consumed

Measurement	Description	Measurement Unit	Interpretation
Static CPU entitlement:	Indicates the static CPU resource entitlement for this VM.	MHz	This value is calculated based on this virtual machine's resource reservations, shares and limit, and doesn't take into account current usage. This is the worst case CPU allocation for this virtual machine – i.e., the amount of CPU resource this virtual machine would receive if all virtual machines running in the cluster went to maximum consumption.
Static memory entitlement:	Indicates the static memory resource entitlement for this VM.	MHz	This value is calculated based on this virtual machine's resource reservations, shares and limit, and doesn't take into account current usage. This is the worst case memory allocation for this virtual machine – i.e., the amount of memory this virtual machine would receive if all virtual machines running in the cluster went to maximum consumption.

3.6.1.1 Configuring Users for VM Monitoring

In order to enable the eG agent to connect to VMs in multiple domains and pull out metrics from them, the eG administrative interface provides a special page using which the different **DOMAIN** names, and their corresponding **ADMIN USER** names and **ADMIN PASSWORDS** can be specified. To access this page, just click on the **Click here** hyperlink in any of the VM test configuration pages.

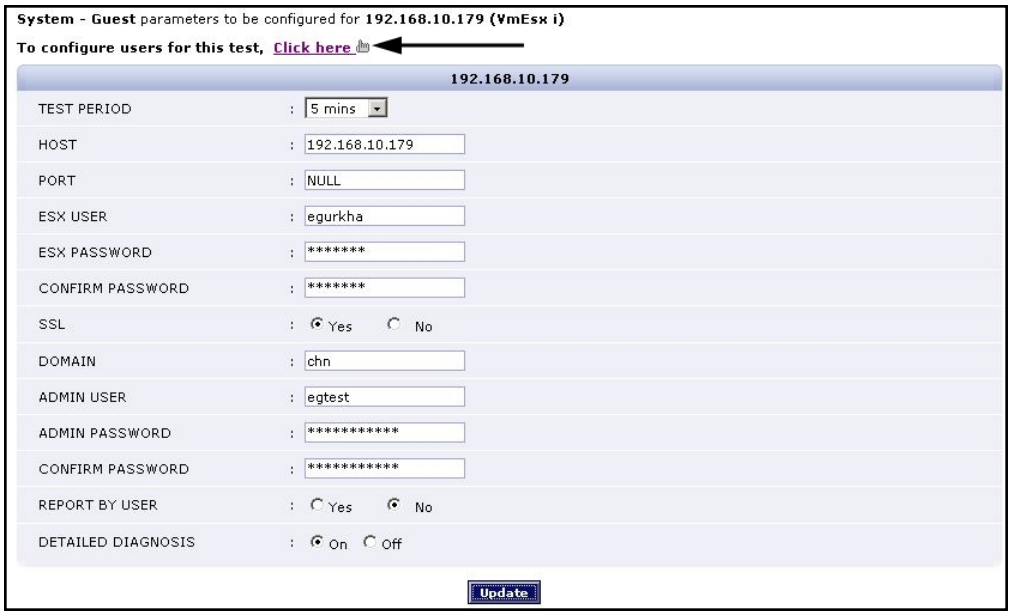


Figure 3.50: Configuring a VM test

Upon clicking, Figure 3.51 will appear, using which the VM user details can be configured.



Figure 3.51: The VM user configuration page

To add a user specification, do the following:

1. First, provide the name of the **Domain** to which the VMs belong (see Figure 3.51). If one/more VMs do not belong to any domain, then, specify *none* here.
2. The eG agent must be configured with user privileges that will allow the agent to communicate with the VMs in a particular domain and extract statistics. If *none* is specified against **Domain**, then a local user account can be provided against **Admin User**. On the other hand, if a valid **Domain** name has been specified, then a domain administrator account can be provided in the **Admin User** text box. If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase*

that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to 5.1.1.

3. The password of the specified **Admin User** should be mentioned in the **Admin Pwd** text box.
4. Confirm the password by retyping it in the **Confirm Pwd** text box.
5. To add more users, click on the ⊕ button in Figure 3.51. This will allow you to add one more user specification as depicted by Figure 3.52.



Figure 3.52: Adding another user

6. In some virtualized environments, the same **Domain** could be accessed using multiple **Admin User** names. For instance, to login to a **Domain** named *egitlab*, the eG agent can use the **Admin User** name *labadmin* or the **Admin User** name *jadmn*. You can configure the eG agent with the credentials of both these users as shown by Figure 3.53.



Figure 3.53: Associating a single domain with different admin users

7. When this is done, then, while attempting to connect to the domain, the eG agent will begin by using the first **Admin User** name of the specification. In the case of Figure 3.53, this will be *labadmin*. If, for some reason, the agent is unable to login using the first **Admin User** name, then it will try to login again, but this time using the second **Admin User** name of the specification - i.e., *jadmin* in our example (see Figure 3.53). If the first login attempt itself is successful, then the agent will ignore the second **Admin User** name.
8. To clear all the user specifications, simply click the **Clear** button in Figure 3.53.
9. To remove the details of a particular user alone, just click the ⊖ button in Figure 3.53.

10. To save the specification, just click on the **Update** button in Figure 3.53. This will lead you back to the test configuration page, where you will find the multiple domain names, user names, and passwords listed against the respective fields (see Figure 3.54).

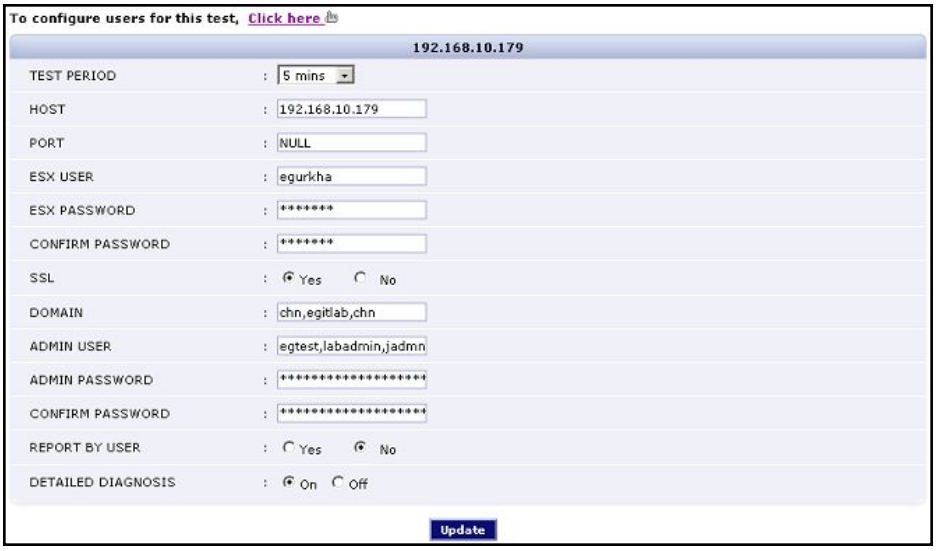


Figure 3.54: The test configuration page displaying multiple domain names, user names, and passwords

3.6.2 Virtual Machines – ESX Test

This test enables administrators to determine how many guests have registered with the ESX server, and how many of these are currently running. In addition, the test also indicates whether any guests have migrated to or from the virtual server. VMware VMotion® technology, unique to VMware, leverages the complete virtualization of servers, storage and networking to migrate an entire running virtual machine instantaneously from a failing/underperforming server to a healthy one, to ensure high availability of the virtual machine. The entire state of a virtual machine is encapsulated by a set of files stored on shared storage, and the VMware VMFS® cluster file system allows both the source and the target ESX Server to access these virtual machine files concurrently. The active memory and precise execution state of a virtual machine can then be rapidly transmitted over a high speed network. Since the network is also virtualized by ESX Server, the virtual machine retains its network identity and connections, ensuring a seamless migration process (see Figure 3.55).

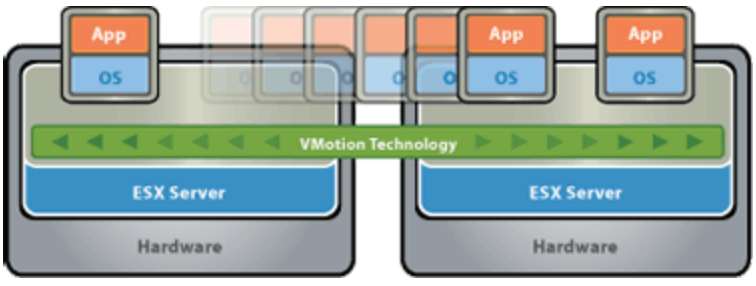


Figure 3.55: How VMotion works?

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the ESX server monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ISVDISERVER** - By default, this flag is set to **NO** indicating that the test is being configured to execute for a *VMware vSphere ESX* model and not a *VMware vSphere VDI* model. This distinction needs to be made, since this test reports 2 additional measures for a *VMware vSphere VDI* server - namely, *Guests with users* and *Guests without users*.
5. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints,

you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

6. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
7. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
8. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
9. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically

displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

10. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the

ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.

- **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1**.

- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.2** .

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

12. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
13. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain**

the 'inside view' of such 'inaccessible' VMs using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

Note:

While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

14. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
15. **DD FOR REGISTERED VMS** – By default, this flag is set to **No**. This means that the test will not report detailed metrics for the *Registered VMs* measure by default. In large virtualized environments characterized by hundreds of registered VMs, this default setting will make sure that there is no unnecessary strain on the eG database. You can, if you so need, enable the test to report detailed metrics for this measure by setting this flag to **Yes**.
16. **DD FOR POWERED ON VMS** – By default, this flag is set to **Yes**. This means that the test will report the complete list of powered on VMs as part of the detailed diagnosis of the *Powered on VMs* measure. To disable this capability for the Powered on VMs measure, set this flag to **No**.
17. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *VMware vSphere ESX* server, this is set to *1:1* by default. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. It is recommended that you do not change the default setting of this parameter. This is because, eG Enterprise can discover the IP addresses of the guest operating systems on an ESX host, only while generating the detailed measures for this test. The automatic discovery of the guest IPs, in turn, enables eG Enterprise to perform **AutoVirtualMapping**.
18. **DETAILED DIAGNOSIS** - It is recommended that this flag be set to **ON** when executing this test on a *VMware vSphere ESX* server.

    The **AutoVirtualMapping** feature will take effect only if this test is running on a *VMware vSphere ESX* server, and is reporting detailed measures.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Registered guests:	Indicates the total number of virtual machines that have been registered with the ESX server.	Number	

Measurement	Description	Measurement Unit	Interpretation
Guests powered on:	Indicates the number of guests that are currently powered on.	Number	To know which are the guests that are powered on, use the detailed diagnosis capability of this measure (if enabled).
Guests with users:	Indicates the number of powered on guests with users logged in.	Number	To know which guests the users have logged into, use the detailed diagnosis capability of this measure (if enabled). Note that this measure will not be available for a VMware vSphere ESX server.
Guests without users:	Indicates the number of powered on guests without any users logged in.	Number	Note that this measure will not be available for a VMware vSphere ESX server.
Added guests:	Indicates the number of guests that were newly added to the ESX server during this measurement period.	Number	The detailed diagnosis of these measures, if enabled, lists the virtual machines that were migrated by VMotion to or from (as the case may be) the ESX server.
Removed guests:	Indicates the number of guests that were newly removed from the ESX server during this measurement period.	Number	
VM templates:	Indicates the number of template VMs currently on the ESX server.	Number	A template is a “golden” copy of a virtual machine (VM) organized by folders and managed with permissions. They are useful because they act as a protected version of a model VM which can be used to create new VMs. As a template is the original and perfect image of a particular VM, it cannot be powered on or run. You can use the detailed diagnosis of this measure to view the names and IP addresses of the template VMs.

Measurement	Description	Measurement Unit	Interpretation
			VM templates occupy disk space on the ESX server. Too many VM templates can sometimes drain the physical disk resources of the ESX host. Hence, if the value of this measure is too high, then, you can determine how much disk space is occupied by each template using the EsxGuestDetails test for the templates, identify the VM templates that are occupying too much space, and take a call on whether to retain such templates or remove them.
Orphaned VMs:	Indicates the number of orphaned VMs on the ESX server, currently.	Number	An orphan virtual machine is one that exists in the vCenter database but is no longer present on the ESX Server host. A virtual machine also shows as orphaned if it exists on a different ESX Server host than the ESX Server host expected by vCenter. A virtual machine can become orphaned, in any of the following situations: • After a VMotion or VMware DRS Migration; • After a VMware HA host failure occurs or after the ESX Server host comes out of maintenance mode; • If you delete a virtual machine outside of vCenter - say, through the VMware Management Interface while VC is down, or through the Virtual Infrastructure (VI) client directly connected to an ESX server host; • If vCenter is restarted while a migration is in progress; this is a temporary situation though; • If you schedule too many virtual

Measurement	Description	Measurement Unit	Interpretation
			machines to be relocated at the same time; To know the names of the orphaned VMs, use the detailed diagnosis of this measure, if enabled.

The detailed diagnosis of the *Guests powered on* measure, provides the complete list of guests that are currently powered on.

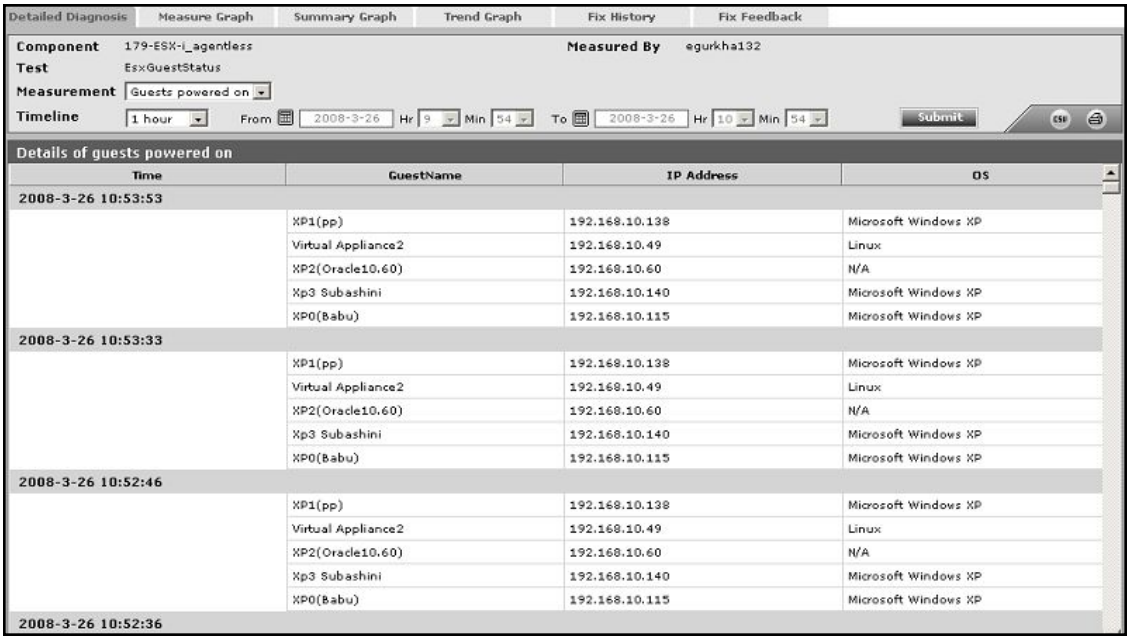


Figure 3.56: The detailed diagnosis of the Guests powered on measure of a VMware vSphere ESX server

To enable you to understand and appreciate how eG Enterprise monitors the VMotion activity on an ESX server, this document takes the help of an illustrated example. This example will attempt to VMotion a VM - *nj_printserver* - from the ESX server - *ESX35_KS* - to the server, *ESX3i*. In the process, the example will demonstrate how eG Enterprise “captures live” the removal of a VM from an ESX server and its addition to another ESX server.

Figure 3.57 depicts the “outside view” of the *nj_printserver* VM on the ESX server, *ESX35_KS*. From the figure, it is evident that the VM *nj_printserver* is currently powered on.

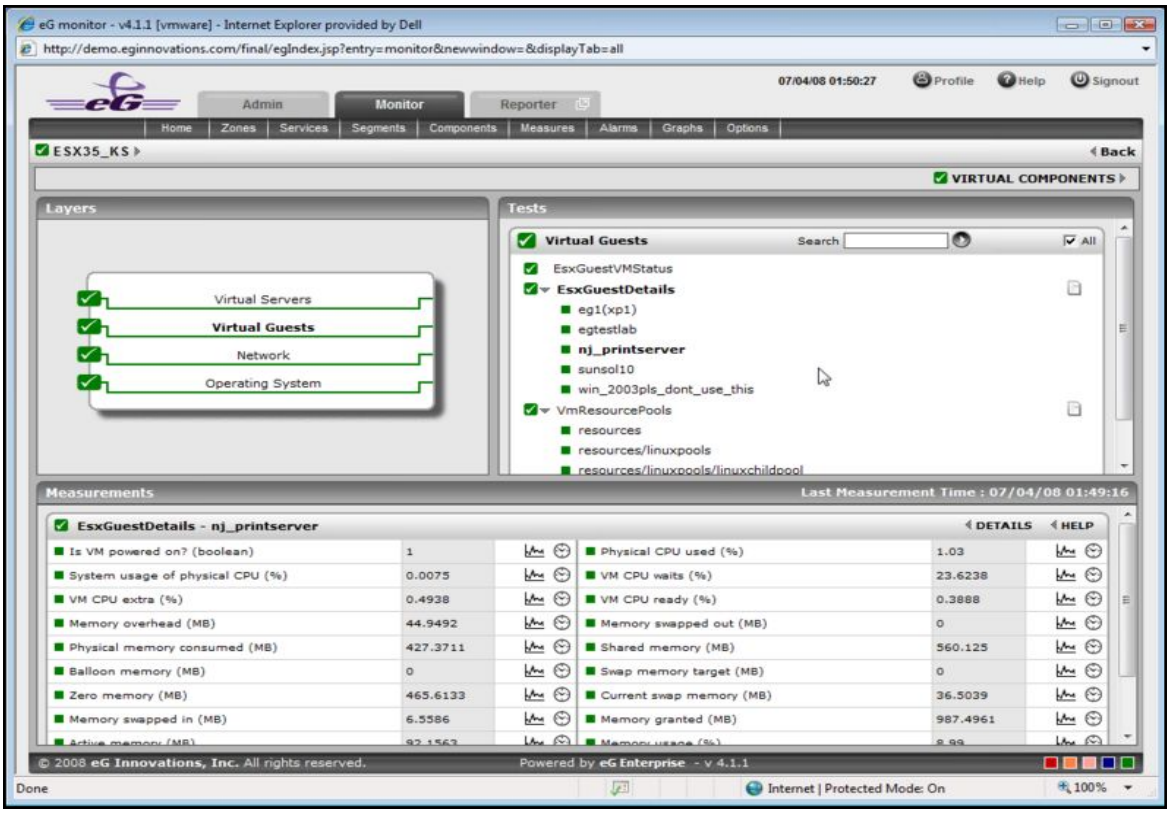


Figure 3.57: The “outside view” measures for the nj_printserver VM

Click on the **VIRTUAL COMPONENTS** link to determine whether any applications are executing on the *nj_printserver* VM. Figure 3.58 then appears revealing that 3 applications are executing on the VMs of the *ESX35_KS* server. One of three is a Print server application (nick name: *Nj_Printserver*), which is obviously executing on the *nj_printserver* VM.

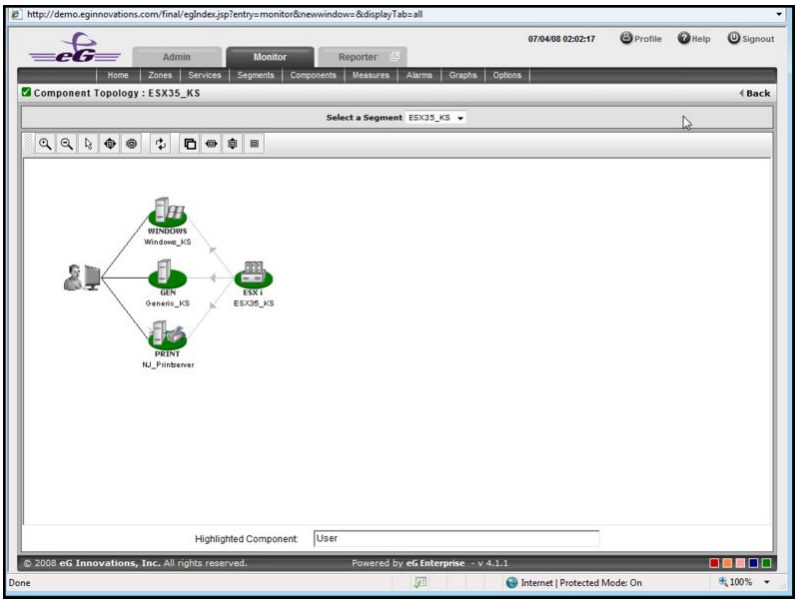


Figure 3.58: The topology of the ESX server, ESX35-KS

Let us now switch to the topology of the *ESX3i* server and figure out if any applications are executing on the VMs configured on that server. To switch to a different topology, select *ESX3i* from the **Select a Segment** list in Figure 3.58. Figure 3.59 that then appears indicates that only an external Oracle server is managed on the *ESX3i* host.

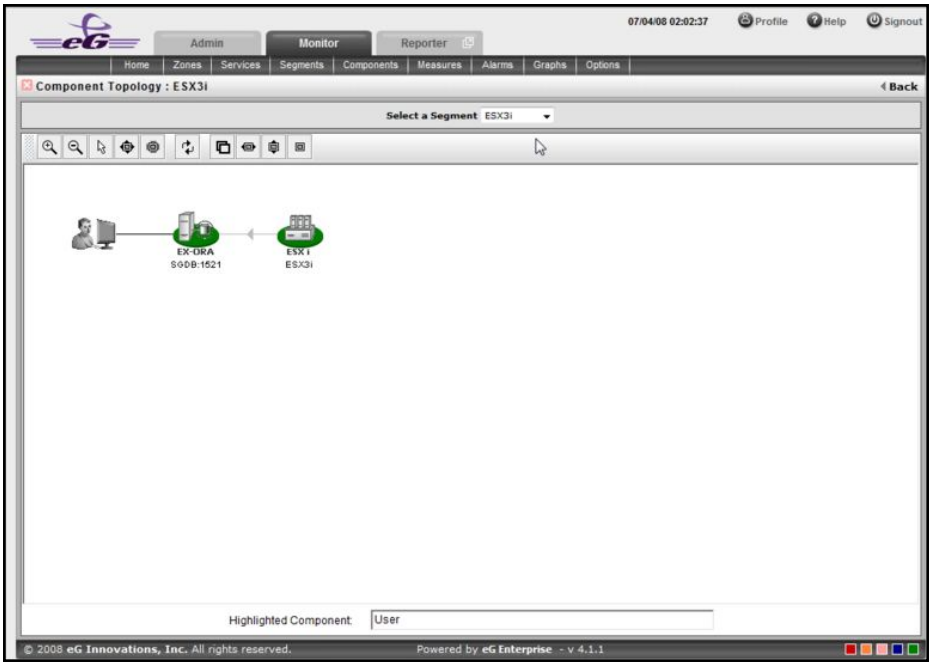


Figure 3.59: The topology of the ESX3i server

Next, let us proceed to VMotion *nj_printserver* VM on *ESX35_KS* to *ESX3i*. For that, do the following:

1. Open the VI Client interface.
2. Drill down the tree-structure in the left panel of the VI Client, to view the *Nj_Printserver* node. This node represents the *nj_printserver* VM that we would be moving shortly.
3. Right-click on the *Nj_Printserver* node and select **Migrate** from the shortcut menu that appears (see Figure 3.60).

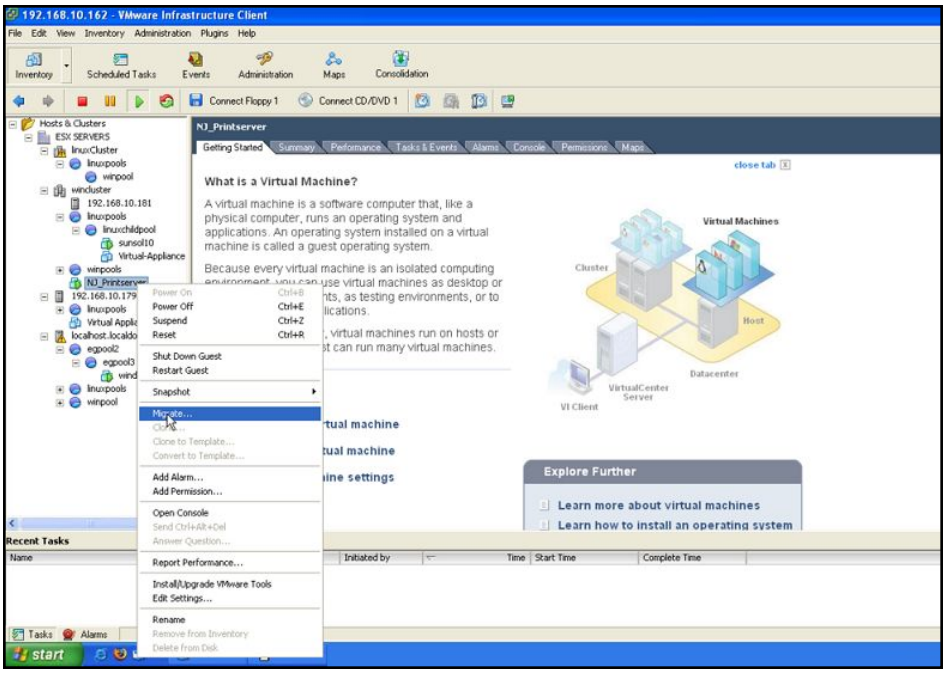


Figure 3.60: Selecting the Migrate option from the shortcut menu

4. Figure 3.61 then appears, from which you need to select the ESX server and resource pool to which the *Nj_Printserver* is to be moved. Then, click **Next** in Figure 3.61 to continue.

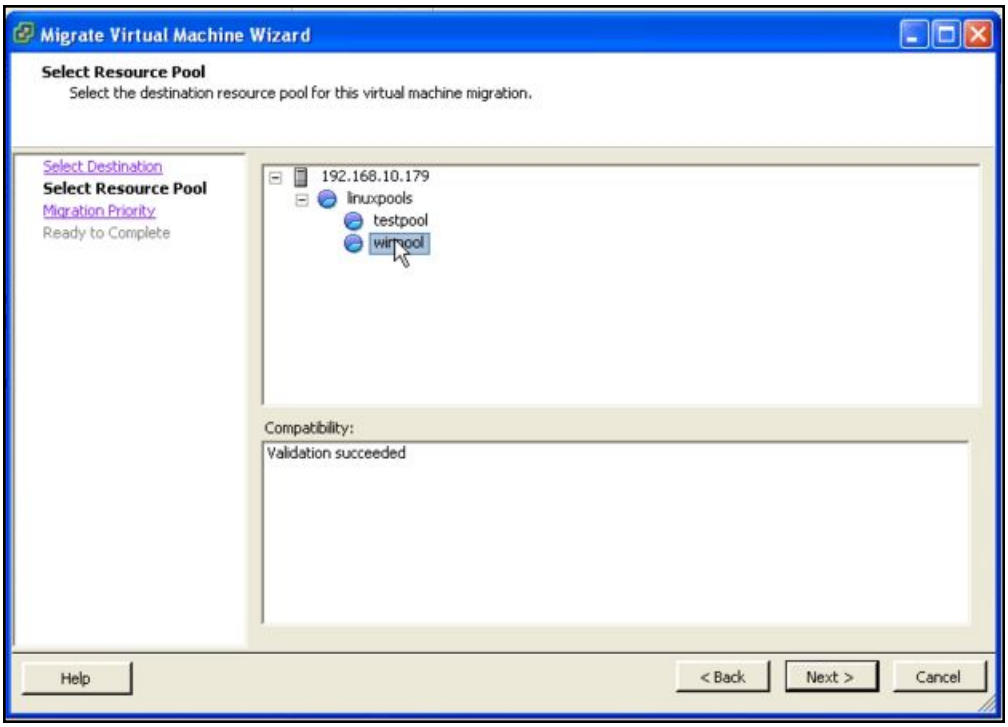


Figure 3.61: Selecting the resource pool to which the VM is to be moved.

5. In Figure 3.62 that then appears, set **Low Priority** as the **Migration Priority** and click **Next** to continue.

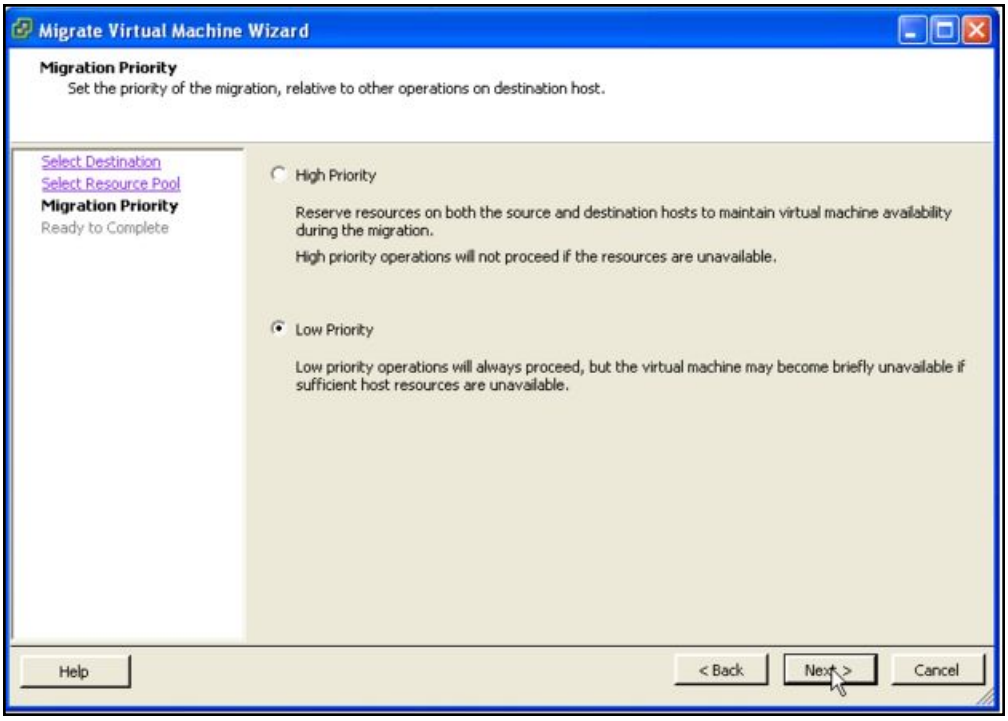


Figure 3.62: Setting the migration priority

6. Click **Finish** in Figure 3.63 that then appears to end the migration.

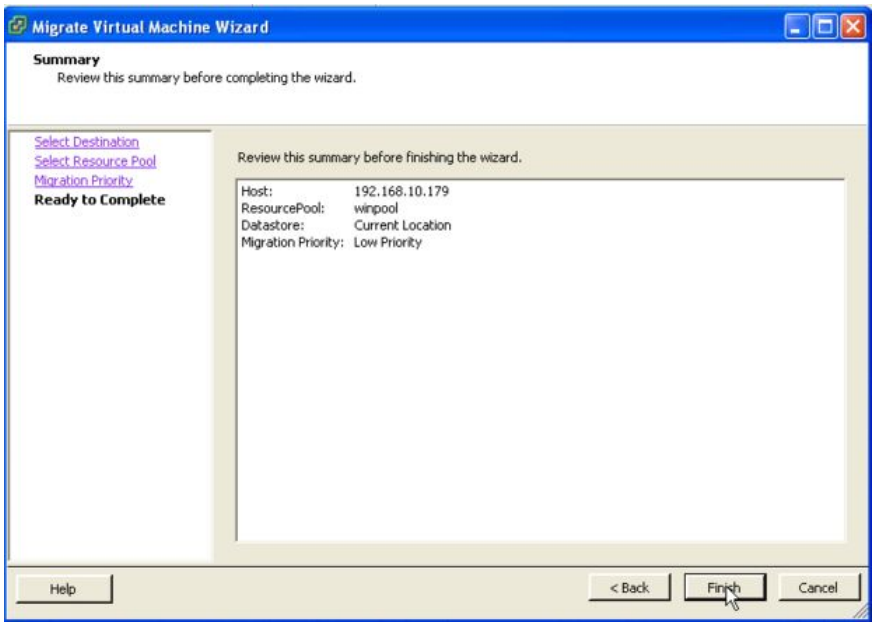


Figure 3.63: Finishing the migration

7. Figure 3.64 that then appears indicates the progress of the migration activity.

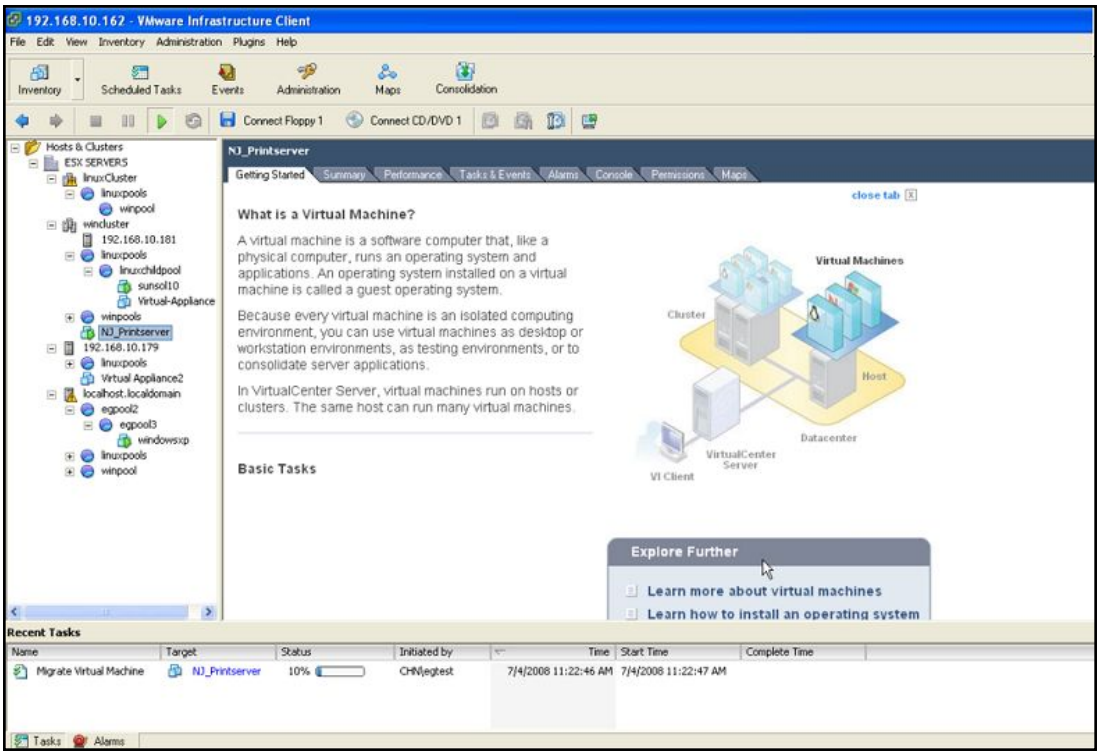


Figure 3.64: Progress of the migration

Once migration completes, proceed to view the impact of the migration on the other ESX server - i.e., *ESX3i* - by following the steps given below:

1. Login to the eG management console as a monitor user.
2. Follow the Components -> Virtual menu sequence to view all the managed virtual hosts.
3. You will find that the ESX server, *ESX3i*, is currently in a **Minor** state. Click on the server representation to view its layer model, tests, and measurements.
4. When the layer model appears, click on the **Virtual Machines** layer in the **Minor** state, and then the **EsxGuestVMStatus** test mapped to this layer, which has apparently reported the problem measure(s) (see Figure 3.65).

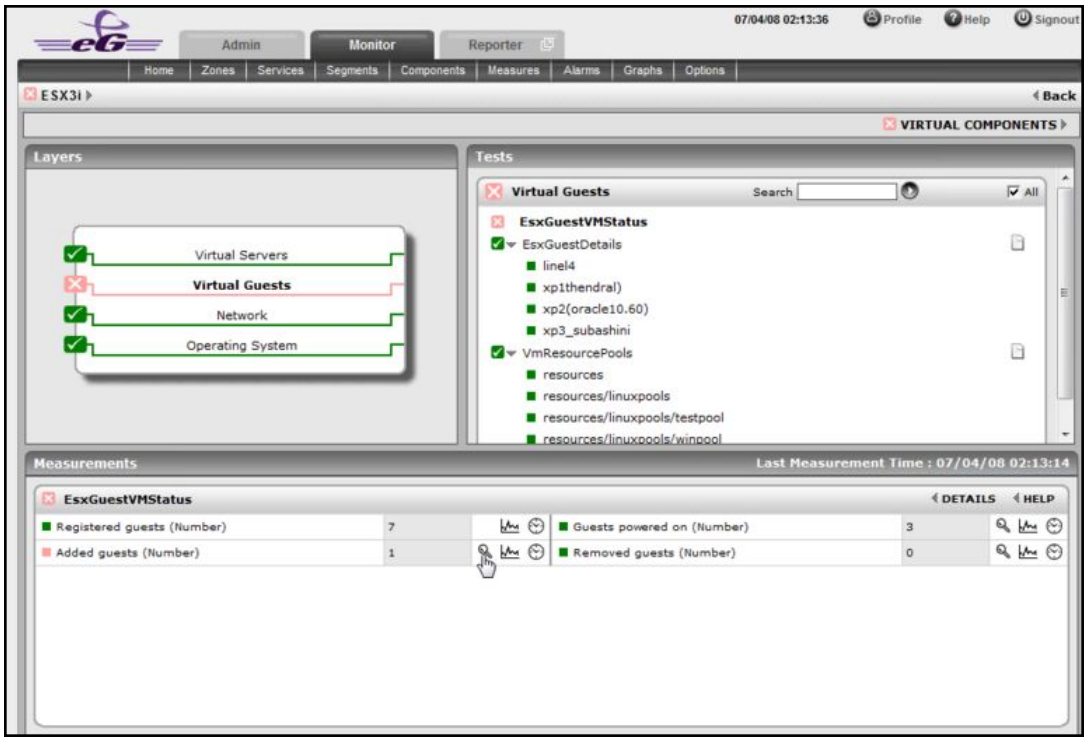


Figure 3.65: Investigating the minor problem with the ESX server, 192.168.10.179

5. Figure 3.65 clearly indicates that the *Added guests* measure reported by the **EsxVMGuestStatus** test is currently in a **Minor** state - this is owing to the addition of a VM to the ESX server *ESX3i*. To know which VM has been migrated to *ESX3i*, click on the **DIAGNOSIS** icon against **Added guests**. Figure 3.66 then appears, revealing that the addition of the *nj_printserver* caused a **Minor** alarm on the ESX server, *ESX3i*.

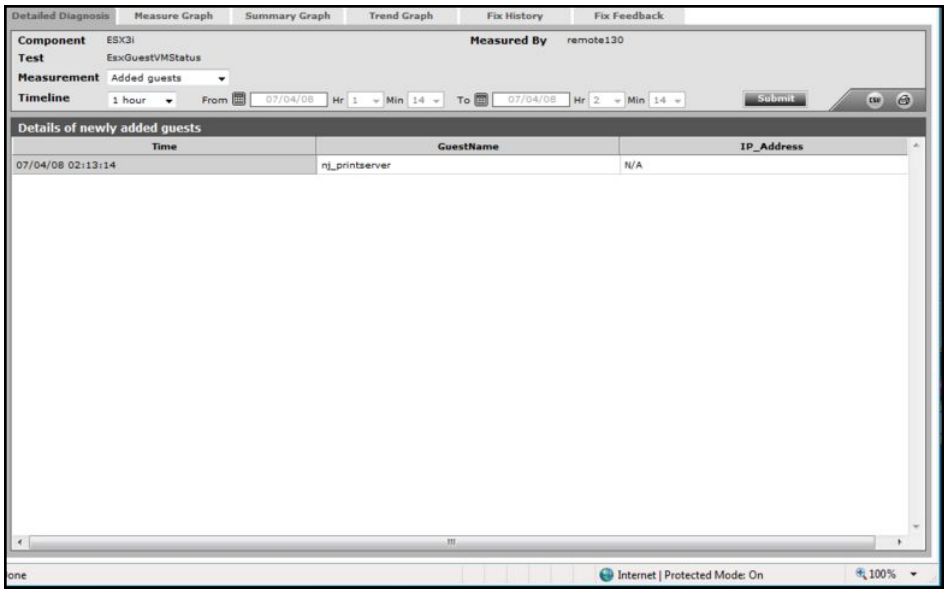


Figure 3.66: Detailed diagnosis of the Added guests measure revealing the addition of a guest to the ESX server, ESX3i

6. Now, return to Figure 3.65, and click the **VIRTUAL COMPONENTS** link at the right, top corner of the page to verify whether the *nj_printserver* has indeed been Vmotioned to *ESX3i*.

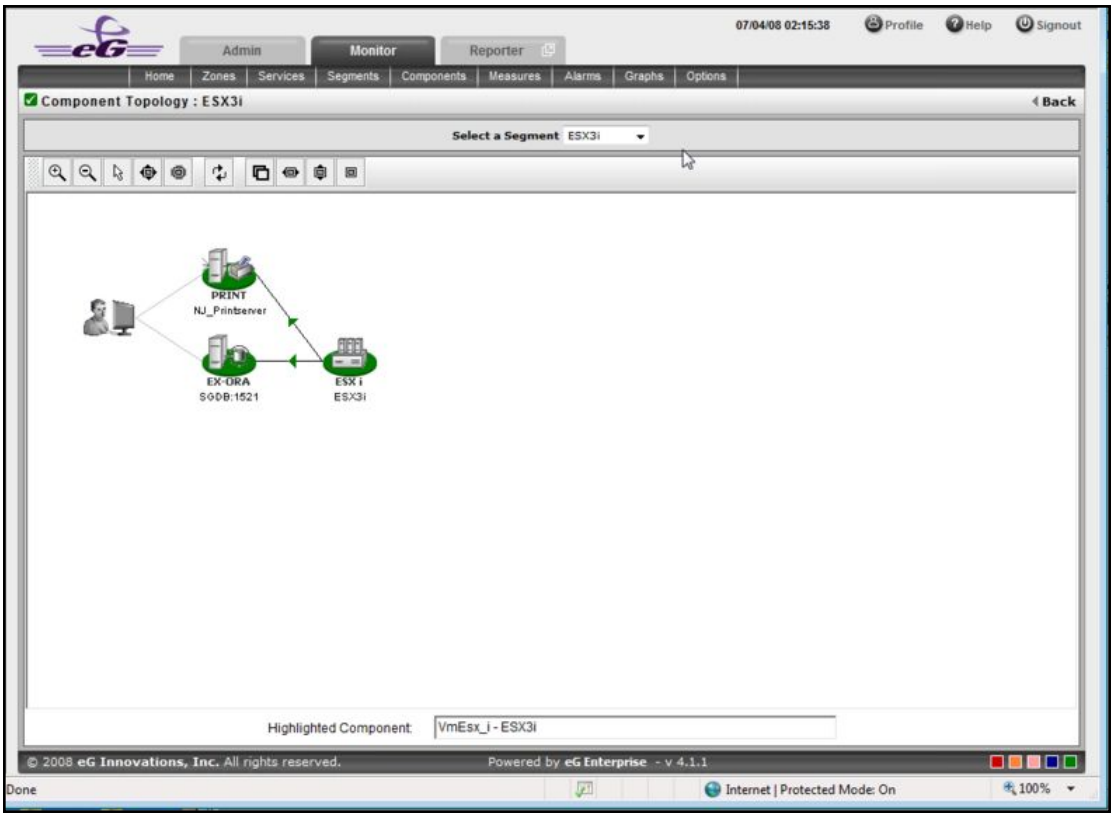


Figure 3.67: The topology of ESX3i server with the Print server application

7. Figure 3.67 then appears. The topology depicted by Figure 3.67 clearly indicates that the externally managed Oracle server that pre-existed on *ESX3i* has now been joined by the print server application, *Nj_Printserver*.

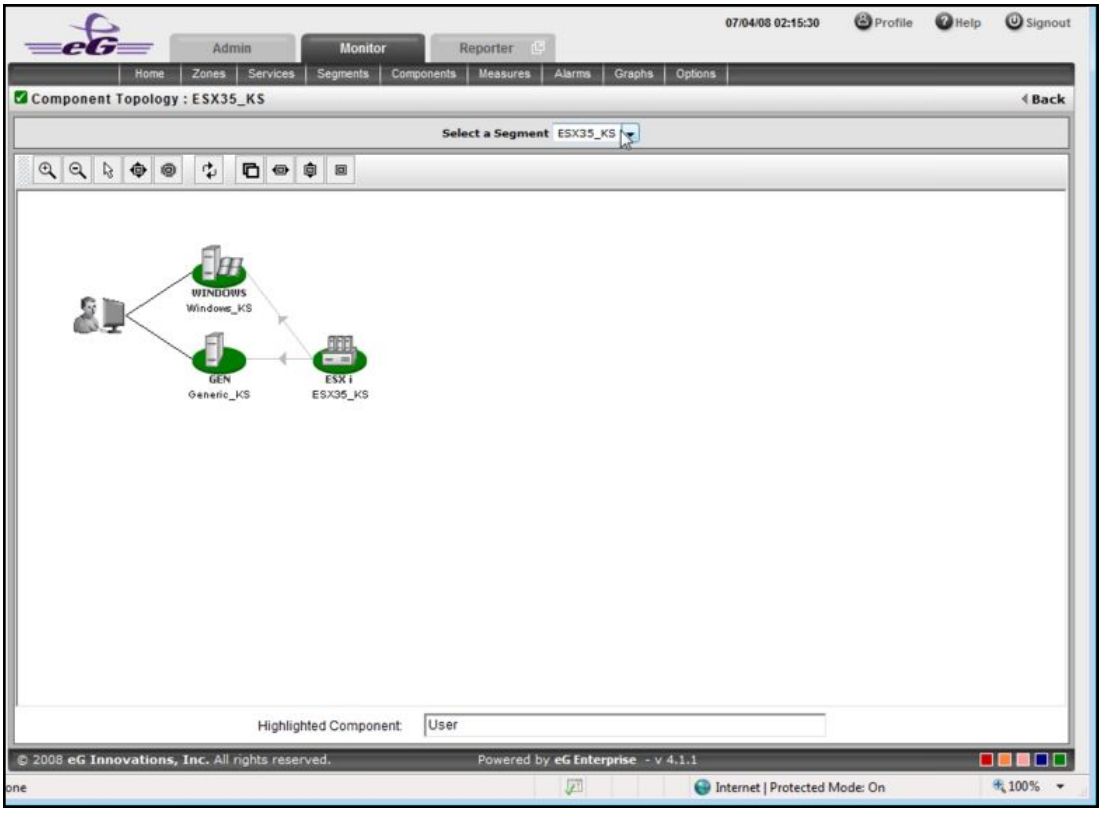


Figure 3.68: The topology of the ESX35_KS server without the Print server application

3.6.3 Resource Pools – ESX Test

Resource pools can be used to hierarchically partition available CPU and memory resources. Each standalone host has an (invisible) root resource pool that groups the resources of that host. The root resource pool is not displayed because the resources of the host and the root resource pool are always the same. If you do not create child resource pools, only the root resource pools exist. Users can create child resource pools of the root resource pool or of any user-created child resource pool. Each child resource pool owns some of the parent's resources and can, in turn, have a hierarchy of child resource pools to represent successively smaller units of computational capability. A resource pool can contain child resource pools, virtual machines, or both. You can therefore create a hierarchy of shared resources. The resource pools at a higher level are called parent resource pools. Resource pools and virtual machines that are at the same level are called siblings.

This test monitors the amount of the physical server's resources that each resource pool on an vSphere/ESXi server is taking up.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each resource pool configured on the ESX server host that is monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** of this document to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG

manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server

via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **LIST ALL POOLS** - By default, resource pools are discovered by first discovering the VMs and then determining which resource pools they are a part of. According to this default behavior, the **LIST ALL POOLS** flag is set to **No** by default. In this case, if a resource pool does not have any VMs, such resource pools will not be discovered. **Note that there is not much benefit in having a resource pool without any VMs assigned to it.** If you have such "empty" resource pools that have no VMs, but you want these discovered as well, then you will have to set the **LIST ALL POOLS** flag to **Yes**. This will force the agent to change its discovery process - to list resource pools by parent child relationships first and then to identify which resource pools have VMs.
10. **REPORT CLUSTER RESOURCE POOLS** - A VMware ESX server can be a part of a cluster or it may not.

    If a VMware ESX server is not part of a cluster, its resource pools will be discovered by the agent and metrics reported for the resource pools. These metrics will appear in the *VMware vSphere ESX* model of eG Enterprise.

    If a VMware ESX server is part of a cluster, its resource pools and its resources will be managed by vCenter, and these clusters and resource pools will show up in the *VMware vCenter* model offered by eG Enterprise. If a VMware ESX server is not part of a cluster, its resource pools will not show up in the *VMware vCenter* model.

    Therefore, when managing an ESX server, if that server is found to be a part of a cluster, then, by default, the agent will not report metrics for its resource pools (since this information will already be available in the *VMware vCenter* model). This is the default behavior and corresponds to the **REPORT CLUSTER RESOURCE POOLS** flag being set to **No** by default. If this flag is changed to **Yes**, metrics for resource pools will be reported in the *VMware vSphere ESX* model irrespective of whether the ESX server is part of a cluster or not.
11. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

- The eG manager license should allow the detailed diagnosis capability
- Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Child resource pools:	Indicates the total number of child resource pools under this resource pool.	Number	
VMs directly assigned to pool:	Indicates the number of virtual machines that were directly assigned to this resource pool.	Number	
VMs powered on:	Indicates the number of virtual machines under this resource pool that is currently powered on.	Number	If the value of this measure is equal to the value of the VMs directly assigned to pool measure, it indicates that all VMs in the pool are in a powered-on state. If the values do not match, it indicates that one/more VMs are currently in a powered-off state.
Physical CPU utilization:	Indicates the percentage of physical CPU resources used from this pool.	Percent	A high value of this measure could indicate excessive CPU usage by the child resource pools/virtual machines configured under this resource pool. By comparing the value of this measure across resource pools, you can quickly identify that resource pool which is draining the CPU resources of the ESX host.
Physical CPU used:	Indicates the CPU usage of the resource pool in Mhz	Mhz	The percent CPU usage measure serves as an effective indicator of how resource-intensive a particular resource pool is on a

Measurement	Description	Measurement Unit	Interpretation
			specific ESX server. However, for performing capacity planning or what-if analysis, the CPU usage of a pool measured in absolute terms would be more useful. For instance, the physical CPU usage of a pool could be 30% on a particular ESX server - this means that the resource pool is consuming 30% of the total physical CPU capacity of that ESX server. If you are planning to migrate the resource pool to another ESX server, then it would be unwise to assume that the pool will only consume 30% of CPU on the other ESX server as well, as the percentage will vary depending upon the physical CPU resources that are available to the other ESX server. The absolute measure however will remain unchanged across ESX server. Therefore, to decide which ESX server a VM/resource pool is to be moved to, and to analyze the impact of this movement on the CPU resources of the new ESX host, you would require an absolute measure of CPU usage.
CPU reservation:	Indicates the amount of CPU resources that is guaranteed available to the resource pool.	Mhz	Typically, the Reservation setting for a resource pool specifies the minimum acceptable amount of CPU or memory - not the amount you would like to have available. Follow the guidelines mentioned below to define the Reservation setting: • Do not set the Reservation too high. A very high reservation can limit the number of virtual machines you can power on in a resource pool. • Always leave some headroom while reserving resource. Do not commit all

Measurement	Description	Measurement Unit	Interpretation
			resources. As you move closer to fully reserving all capacity in the system, it becomes increasingly difficult to make changes to reservations and to the resource pool hierarchy without violating admission control. • If you expect frequent changes to the total available resources, use Shares, not Reservation, to allocate resources fairly across resource pools.
CPU reservation used:	Indicates the total amount of CPU resources that have been used to satisfy the reservation requirements of all descendants of this resource pool.	Mhz	CPU resources are typically reserved for powered- on virtual machines in a resource pool only. Virtual machines that are not currently running do not use any CPU resources. Reserved resources are not wasted if they are not used. If the utilization is less than the reservation, the resources can be utilized by other running virtual machines.
CPU unreserved:	Indicates the total amount of resources available to satisfy a reservation for a child resource pool.	Mhz	In the undercommitted state, this value is limited by the capacity at the root node. In the overcommitted case, this could be higher since we do not perform the dynamic capacity checks. Consider a resource pool with reservation=2GHz that is totally idle. It has 2GHz reserved, but it is not actually using any of its reservation. In such a case: • Other resource pools cannot reserve these 2GHz. • Other resource can use these 2GHz, that is, idle CPU reservations are not wasted.

Measurement	Description	Measurement Unit	Interpretation
CPU allocation used:	Indicates the percentage of allocated CPU resources that are being used by this resource pool.	Percent	A high value of this measure indicates that one/more VMs in the pool are consuming the allocated CPU resources excessively.
CPU limit:	Indicates the limit beyond which this resource pool should not use CPU resources, even if there are resources available.	Mhz	If the **UNLIMITED** flag is enabled for a resource pool, then this measure will report the value Unlimited. This setting helps avoid wastage of idle resources.
Memory used:	Indicates the amount of memory used by this resource pool.	MB	
Memory reservation:	Indicates the amount of memory resources that is guaranteed available to the resource pool.	MB	Typically, the Reservation setting for a resource pool specifies the minimum acceptable amount of CPU or memory - not the amount you would like to have available. Follow the guidelines mentioned below to define the Reservation setting: • Do not set the Reservation too high. A very high reservation can limit the number of virtual machines you can power on in a resource pool. • Always leave some headroom while reserving resource. Do not commit all resources. As you move closer to fully reserving all capacity in the system, it becomes increasingly difficult to make changes to reservations and to the resource pool hierarchy without violating admission control. • If you expect frequent changes to the total available resources, use Shares, not Reservation, to allocate resources

Measurement	Description	Measurement Unit	Interpretation
			fairly across resource pools.
Memory reservation used:	Indicates the total amount of memory resources that have been used to satisfy the reservation requirements of all descendants of this resource pool.	MB	Memory resources are typically reserved for powered-on virtual machines in a resource pool only. Virtual machines that are not currently running do not use any memory resources. Reserved resources are not wasted if they are not used. If the utilization is less than the reservation, the resources can be utilized by other running virtual machines.
Memory unreserved:	Indicates the total amount of resources available to satisfy a reservation for a child resource pool.	MB	In the undercommitted state, this value is limited by the capacity at the root node. In the overcommitted case, this could be higher since we do not perform the dynamic capacity checks. If the resource pool does not use the reserved memory, then remember the following: • Other resource pools cannot reserve these memory resources • Other resource pools can use these memory resources, that is, idle memory reservations are not wasted.
Memory limit:	Indicates the limit beyond which this resource pool should not use memory resources, even if there are resources available.	MB	If the **UNLIMITED** flag is enabled for a resource pool, then this measure will report the value Unlimited. This setting helps avoid wastage of idle resources.
Active memory:	Indicates the amount of memory resources actively used from this pool.	MB	High memory consumption by a resource pool, besides eroding the physical memory resources of the host, can also impact the memory allocations to other resource pools/virtual machines; the lack

Measurement	Description	Measurement Unit	Interpretation
			of adequate memory to a VM can significantly strain its performance.
Active memory used:	Indicates the percentage of total configured or available memory in this pool.	Percent	
Zero memory:	Indicates the amount of memory that is zeroed out.	MB	
Swapped memory:	Indicates the amount of memory that is swapped.	MB	ESX Server hosts use swapping to forcibly reclaim memory from a virtual machine when no vmmemctl driver is available because the vmmemctl driver: • Was never installed • Has been explicitly disabled • Is not running (for example, while the guest operating system is booting) • Is temporarily unable to reclaim memory quickly enough to satisfy current system demands Standard demand- paging techniques swap pages back in when the virtual machine needs them. Swap space must be reserved on disk for any unreserved virtual machine memory. This swap reservation is required to ensure the system is able to preserve virtual machine memory under any circumstances. In practice, only a small fraction of the swap space may actually be used. Typically, swap space usage for each VM should be low. Since access from RAM is much faster than access from physical disk, excessive usage of swap memory will slow down the performance of a VM. Watch for VMs that are seeing higher

Measurement	Description	Measurement Unit	Interpretation
			swap usage and more swap reads and writes.
Memory overhead:	Indicates the memory overhead incurred by this resource pool.	MB	ESX Server virtual machines can incur two kinds of memory overhead: • The additional time to access memory within a virtual machine. • The extra space needed by the ESX Server host for its own code and data structures, beyond the memory allocated to each virtual machine. The ESX server's memory virtualization ensures that the time overhead to memory accesses is minimal. The memory space overhead, on the other hand, is composed of two other components: • A fixed system-wide overhead for the service console (in the case of ESX 3/3.5) and the VMkernel. • Additional overhead for each virtual machine In addition, the space reserved for the virtual machine frame buffer and various virtualization data structures, also add to the memory overhead. Addition of more virtual CPUs, allocation of more memory to the guests, and choosing to configure 32- bit guest operating systems instead of 64-bit ones, can go a long way in reducing this overhead. The ESX server also provides optimizations such as memory sharing to save up memory.
Balloon memory:	Indicates the amount of memory used by memory	MB	The vmmectl driver that is installed on a

Measurement	Description	Measurement Unit	Interpretation
	control.		virtual machine, emulates an increase or decrease in memory pressure on the guest operating system; this way, it forces the guest OS to place memory pages into its local swap file. This driver differs from the VMware swap file method as it forces the operating system to determine what memory it wishes to page. Once the memory is paged locally on the guest operating system, the free physical pages of memory may be reallocated to other guests. As the ESX hosts sees that memory demand has been reduced, it will instruct vmmemctl to "deflate" the balloon and reduce pressure on the guest OS to page memory. The maximum amount of memory that can be reclaimed from a guest may be configured by modifying the "sched.mem.maxmemctl" advanced option. If the memory reclaimed from a guest (i.e., the value of this measure) is very low, it indicates excessive memory usage by the guest. Under such circumstances, you might want to consider allocating more memory to the guest.
Shared memory:	Indicates the amount of memory in the resource pool that is shared.	MB	
Memory granted:	Indicates the amount of memory that is granted to this resource pool.	MB	

3.6.4 VM Connectivity Test

Sometimes, a VM could be in a powered-on state, but the failure of the VM operating system or any fatal error in VM operations could have rendered the VM inaccessible to users. In order to enable administrators to promptly detect such 'hidden' anomalies, the eG agent periodically runs a connectivity check on each VM using this test, and reports whether the VM is accessible over the network or not.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each VM configured on the ESX host being monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .

   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the

statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in

your environment listens so that the eG agent communicates with that port.

8. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

   Note:

   While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
10. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

    If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

    On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

    On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

11. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**.

12. **PACKETSIZE** - The size of packets used for the test (in bytes).
13. **PACKETCOUNT** - The number of packets to be transmitted during the test.
14. **TIMEOUT** - How long after transmission should a packet be deemed lost (in seconds).
15. **PACKETINTERVAL** - Represents the interval (in milliseconds) between successive packet transmissions during the execution of the network test for a specific target.
16. **REPORTUNAVAILABILITY** – By default, this flag is set to **No**. This implies that, by default, the test will not report the unavailability of network connection to any VM. In other words, if the Network availability measure of this test registers the value 0 for any VM, then, by default, this test will not report any measure for that VM; under such circumstances, the corresponding VM name will not appear as a descriptor of this test. You can set this flag to **Yes**, if you want the test to report and alert you to the unavailability of the network connection to a VM.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Avg network delay:	Indicates the average delay between transmission of packet to a VM and receipt of the response to the packet at the source.	Secs	An increase in network latency could result from misconfiguration of the router (s) along the path, network congestion, retransmissions at the network, etc.
Min network delay:	The minimum time between transmission of a packet and receipt of the response back.	Secs	A significant increase in the minimum round-trip time is often a sure sign of network congestion.
Packet loss:	Indicates the percentage of	Percent	Packet loss is often caused by network

Measurement	Description	Measurement Unit	Interpretation
	packets lost during transmission from source to target and back.		buffer overflows at a network router or by packet corruptions over the network. The detailed diagnosis for this measure provides a listing of routers that are on the path from the external agent to target server, and the delays on each hop. This information can be used to diagnose the hop(s) that could be causing excessive packet loss/delays.
Network availability of VM:	Indicates whether the network connection is available or not.	Percent	A value of 100 indicates that the VM is connected. The value 0 indicates that the VM is not connected. Typically, the value 100 corresponds to a Packet loss of 0.

3.6.5 VM Datastores Test

This test monitors the availability, usage, and I/O activity of every datastore allocated to every virtual machine on an ESX server. If users are unable to access a VM, then this test will enable administrators to determine whether it can be attributed to the non-availability of the datastore allocated to the VM, the lack of space in the datastore, or a high I/O latencies experienced by the datastore.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each datastore used by each VM configured on the ESX host being monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

- **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6.

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session

timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**.

10. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

11. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Availability of datastore for the VM:	Indicates whether/not this datastore is available.	Percent	While the value 0 indicates that the datastore is not available, the value 100 indicates that the datastore is currently

Measurement	Description	Measurement Unit	Interpretation
			available. If a datastore becomes unavailable, then VMs that are currently using that datastore could be rendered inaccessible to users.
Portion of datastore space used by VM:	Indicates the storage space on this datastore that is actually being used by this virtual machine.	MB	Comparing the value of this measure across VMs will enable you to quickly identify the VM that is consuming maximum datastore space.
Datastore space used by other files of the VM:	Indicates the percentage of space on this datastore that is being consumed by this VM.	Percent	If a datastore is experiencing a space crunch, then you can compare the value of this measure across VMs to identify which VM is responsible for it.
Datastore space used by other files of the VM:	Indicates the total space on this datastore that is being used by files such as configuration, suspend data, NVRAM, screenshots and others. This does not include Virtual Disks, snapshots or swap files.	MB	If a datastore is being rapidly drained of space, then, you can compare the value of the *Other VM space* and *Space used* measure for each VM, so as to quickly isolate what type of files associated with which VM are occupying too much space.
Space used by VM in datastore:	Indicates the percentage of space used on this datastore by other files.	Percent	If a datastore is being rapidly drained of space, then, you can compare the value of the *Percent other VM space* and *Percent space used* measures across VMs, so as to quickly isolate what type of files associated with which VM are occupying too much space.
Uncommitted space of VM in datastore:	Indicates the additional storage space potentially used by this virtual machine on this datastore.	MB	Additional space may be needed for example when lazily allocated disks grow, or storage for swap is allocated when powering on the virtual machine. If the virtual machine is running off delta disks (for example because a snapshot was taken), then only the potential

Measurement	Description	Measurement Unit	Interpretation
			growth of the currently used delta-disks is considered.
Unshared space of VM in datastore:	Indicates the storage space, in bytes, occupied by the virtual machine on this datastore that is not shared with any other virtual machine.	MB	
Provisioned space of VM in datastore:	Indicates the amount of space on this datastore that has been provisioned for this VM.	MB	This is typically the sum total of the disk space used by the VM and the uncommitted space for this VM on this datastore.
Number of virtual disks:	Indicates the number of virtual disks in this datastore.	Number	
Swap file space:	Indicates the amount of space in this datastore that is occupied by the swap file of this VM.	MB	A swap file is created by the ESX/ESXi host when a virtual machine is powered on. If this file cannot be created, the virtual machine cannot power on. By default, the swap file is created in the same location as the virtual machine's configuration file. In the event of a space crunch, you may want to compare the values of the Swap file space, Virtual disk file space, and Snapshot file space measures to figure out which file is contributing to the space contention.
Swap file space usage:	Indicates the percentage of space in this datastore that is occupied by the swap file of this VM.	Percent	
Snapshot file space:	Indicates the amount of space in this datastore that is used for storing a snapshot of this VM.	MB	A snapshot captures the entire state of the virtual machine at the time you take the snapshot. Snapshots are useful when you need to revert repeatedly to the same state but you do not want to create multiple virtual machines.

Measurement	Description	Measurement Unit	Interpretation
			A snapshot includes the following information: a. Contents of the virtual machine's memory b. Virtual machine settings c. State of all the virtual machine's virtual disks In the event of a space crunch, you may want to compare the values of the *Swap file space, Virtual disk file space*, and *Snapshot file space* measures to figure out which file is contributing to the space contention.
Snaphot file space usage:	Indicates the percentage of space that is used for creating a snapshot of this VM in this datastore.	Percent	
Virtual disk file space:	Indicates the amount of space in this datastore that is occupied by the virtual disk file of this VM.	MB	A virtual machine uses a virtual hard disk to store its operating system, program files, and other data associated with its activities. A virtual disk is a large physical file, or a set of files, that can be copied, moved, archived, and backed up as easily as any other file. To store virtual disk files and manipulate the files, a host requires dedicated storage space. In the event of a space crunch, you may want to compare the values of the *Swap file space, Virtual disk file space*, and *Snapshot file space* measures to figure out which file occupies the maximum space in a datastore.
Virtual disk file space usage:	Indicates the percentage of space available for the virtual disk of this VM in this datastore.	Percent	
Is thin provisioned:	Indicates whether this datastore is thin provisioned	Number	This measure reports either *Yes* or *No* while indicating whether this datastore

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>or not.</td><td></td><td>is thin provisioned or not.

The numeric values that correspond to those states are as follows:

<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Yes</td><td>0</td></tr><tr><td>No</td><td>1</td></tr></table>
Note:

By default, this measure reports the above- mentioned states while indicating whether this datastore is thin provisioned or not. However, the graph of this measure will be represented using the corresponding numeric equivalents of the states as mentioned in the table above.</td></tr>
<tr><td>Percentage of saved space:</td><td>Indicates the percentage of space saved by this VM in this datastore.</td><td>Percent</td><td></td></tr>
<tr><td>Average write requests rate:</td><td>Indicates the average number of write commands issued per second to this datastore by this VM during the collection interval.</td><td>Commands/Sec</td><td></td></tr>
<tr><td>Average read requests rate:</td><td>Indicates the average number of read commands issued per second to this datastore by this VM during the collection interval.</td><td>Commands/Sec</td><td></td></tr>
<tr><td>Write latency:</td><td>Indicates the average amount of time taken for a write by this VM from the perspective of this datastore.</td><td>Secs</td><td>A high value indicates that the VM is taking too long to write to the datastore. Compare the value of this metric across VMs and datastores to know which VM registered the maximum write latency</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			with respect to which datastore.
Read latency:	Indicates the average amount of time taken for a read by this VM from the perspective of this datastore.	Secs	A high value indicates that the VM is taking too long to read from the datastore. Compare the value of this metric across VMs and datastores to know which VM registered the maximum read latency with respect to which datastore.
Read rate:	Indicates the rate at which data is read from this datastore.	MB/Sec	A high value is desired for these measures. A low value or a consistent dip in the value is indicative of an I/O latency.
Write rate:	Indicates the rate at which data is written to this datastore.	MB/Sec	
Total latency:	Indicates the total amount of time taken to read from and write into this datastore from the perspective of guest operating system.	Secs	A high value or a consistent rise in the value of this measure is a cause for concern, as it indicates a current/potential I/O processing bottleneck with the datastore.
Throughput:	Indicates the rate at which the data is read from or written into this datastore.	MB/Sec	Ideally, the value of this measure should be high. A sudden or steady drop in this value could indicate a gradual deterioration in the performance of the datastore, probably owing to a processing bottleneck.
Total IOPS:	Indicates the total number of write commands issued and read commands issued per second to this datastore by this VM.	Commands/Sec	

3.6.6 VM Snapshots Test

A disk snapshot is a copy of the virtual machine (VM) disk file recorded at a specific time, much like a Windows restore point. The snapshot preserves the original VM disk file by disabling writes to the original disk file - all new writes are made to the snapshot version of the VM. If you create more than one snapshot of your virtual machine (VM), then you'll have multiple restore points available to revert to. When you create a snapshot, what was currently writable becomes read-only from that point on. Using in-file delta technology, new files are created that contain all changes (delta) to the original disk files.

The size of a snapshot file can never exceed the size of the original disk file. When requests are made to change a block on the original disk, it is instead changed in the delta file. If the previously changed disk block in a delta file is changed again it will not increase the size of the delta file because it simply updates the existing block in the delta file.

Though snapshot files are small in size initially, they will grow as writes are made to the VM's disk files. If the number and size of the snapshot files grow significantly over time, they might end up eroding considerable datastore space, thereby choking VM operations. To conserve disk space, administrators need to continuously track snapshot growth per VM, identify 'heavy-weight' snapshots that may not be of use any longer, and purge them. The **VM Snapshots** test helps administrators achieve the same. While the measures reported by the test capture the snapshot file count per VM and the total size of the snapshot files of a VM, the detailed diagnosis reveals the size of each snapshot, thus enabling administrators to quickly spot those snapshot files that are too large in size.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each VM configured on the ESX host being monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the

"Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.

6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1**.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain

administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**.

10. **AGELIMIT** - By default, 15 days is set as **AGELIMIT**. This implies that the test will report all those snapshots that are more than 15 days old as **Old snapshots**. If required, you can change the **AGELIMIT**.
11. **SIZELIMIT**- By default, 10000 KB is set as the **SIZELIMIT**. This implies that the test will report all those snapshots that have a size more than 10000 KB as **Large snapshots**. If required, you can change the **SIZELIMIT**.
12. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For instance, if you set to *4:1*, it means that detailed measures will be generated every fourth time this test runs, and also every time the test detects a problem.
13. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Number of snapshots:	Indicates the number of snapshot files of this VM that are currently available.	Number	A number of snapshots of a VM provides administrators with multiple restore points. On the flipside though, a high number of snapshots can also be considered a waste of valuable disk space, especially if many of the snapshots hold less critical, but heavy-weight changes/writes to the disk. To accurately identify those snapshots

Measurement	Description	Measurement Unit	Interpretation
			that are consuming disk space excessively, and to learn when they were created, who their parents are, and their current consistent file system state, use the detailed diagnosis of this measure.
Total size of snapshots:	Indicates the total size of all snapshots of a VM.	MB	Snapshots typically grow in 16 MB increments to help reduce SCSI reservation conflicts. Though small in size initially (16 MB), snapshots can grow with time, but can never grow beyond the original disk file size. If a marked increase is noticed in the value of this measure over time, it could indicate that one/more snapshots are rapidly growing in size. To know which snapshots are contributing to this phenomenon, use the detailed diagnosis of the Number of snapshots measure. The rate of growth of a snapshot will be determined by how much disk write activity occurs on your server. Servers that have disk write intensive applications, such as SQL and Exchange, will have their snapshot files grow rapidly. On the other hand, servers with mostly static content and fewer disk writes, such as Web and application servers, will grow at a much slower rate.
Large size snapshots count:	Indicates the number of snapshots that are of a size more than the configured **SIZELIMIT**.	Number	Snapshots typically grow in 16 MB increments to help reduce SCSI reservation conflicts. Though small in size initially (16 MB), snapshots can grow with time, but can never grow beyond the original disk file size.

Measurement	Description	Measurement Unit	Interpretation
			If a marked increase is noticed in the value of this measure over time, it could indicate that a number of snapshots are rapidly growing in size. To know which snapshots are growing beyond the size limit set, use the detailed diagnosis of this measure.
Aged snapshots count:	Indicates the number of snapshots that are of an age over the configured **AGELIMIT**.	Number	Use the detailed diagnosis of this measure to identify the old snapshots, so that you can figure out whether they deserve to be retained or not. While many snapshots provide essential restore points for VMs, many others hold less critical information. The 'less useful' snapshots can be eliminated to save disk space.

To accurately identify those snapshots that are consuming disk space excessively, and to learn when they were created, who their parents are, and their current consistent file system state, use the detailed diagnosis of this measure.

Details of snapshots of VMs

Time	VMName	VMStatus	SnapshotId	SnapshotCreatedTime	SnapshotName	ParentSnapshotName	ConsistentStateOfFileSystem	SnapshotSize (MB)
Nov 24, 2009 17:18:51								
	VMware Studio - karthik & srihari team	poweredOff	4	22/09/2009 13:22:10	sak	vmsnapshot-4	false	177
	VMware Studio - karthik & srihari team	poweredOff	2	14/09/2009 11:16:48	VMwareStudioSnapShot-2	VMwareStudioSnapShot	false	177
	VMware Studio - karthik & srihari team	poweredOff	1	14/09/2009 11:16:10	VMwareStudioSnapShot	Root SnapShot	false	177
	VMware Studio - karthik & srihari team	poweredOff	3	21/09/2009 15:09:57	vmsnapshot-4	VMwareStudioSnapShot-2	false	177

Figure 3.69: The detailed diagnosis of the VM Snapshots test

3.6.7 VM Device Status Test

For each VM configured on a vSphere host, this test reports the availability of VMware Tools on the VM, and also reports the count and status of hardware/peripheral devices (such as floppy drives, CD drives, virtual disks, CPUs etc.) allocated to the VM.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each VM configured on the ESX host being monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG

manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server

via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
VMware tools status:	Indicates the current status of VMware Tools on this VM.		VMware Tools is a suite of utilities that enhances the performance of the virtual machine's guest operating system and improves management of the virtual machine. Without VMware Tools installed in your guest operating system, guest performance lacks important functionality. Installing VMware Tools eliminates or improves the following issues:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
a. Low video resolution
b. Inadequate color depth
c. Incorrect display of network speed
d. Restricted movement of the mouse
e. Inability to copy and paste and drag-and-drop files
f. Missing sound

If VMware Tools is running on the VM, then this test will report the value Running. On the other hand, if VMware Tools has been installed, but is not running on the VM, this test will report Not Running as the status. If VMware Tools has been installed and is just being started, then the value reported by this measure will be Starting. If VMware Tools has not at all been installed on the VM, the measure will report the value Not Installed. If VMware Tools is installed and running, but its version is not current, then, the measure will report the value Out of Date.

The numeric values that correspond to the states discussed above are listed in the table below:
<table>
<tr><th>Numeric Value</th><th>State</th></tr>
<tr><td>0</td><td>Not Running</td></tr>
<tr><td>1</td><td>Running</td></tr>
<tr><td>2</td><td>Starting</td></tr>
<tr><td>3</td><td>Not Installed</td></tr>
<tr><td>4</td><td>Out of Date</td></tr>
</table>
Note:
</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			By default, this measure reports only the States discussed in the table above. The graph of this measure however, represents the VMware Tools status using the numeric equivalents - 0 to 4.
CD/DVD drives in the VM:	Indicates the number of CD/DVD drives in the VM.	Number	
CD/DVD drives connected to the VM:	Indicates the number of CD/DVD drives that are currently connected to the VM.	Number	
Total floppy drives in the VM:	Indicates the total number of floppy drives in the VM.	Number	
Floppy drives connected to the VM:	Indicates the number of floppy drives that are currently connected to the VM.	Number	
Virtual CPUs:	Indicates the f virtual Us allocated to this VM.	Number	
Virtual disks:	Indicates the number of virtual disks allocated to this VM.	Number	

3.6.8 VM Virtual Disks Test

This test auto-discovers the virtual disk partitions associated with each VM on a vSphere/ESX server, monitors the read/write request load on each disk partition, and reports how swiftly the disk partition is able to process the load. In the process, delays in request processing come to light.

This test is disabled by default for a *VMware vSphere VDI* component.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each virtual disk associated with the VM on the ESX host being monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG

manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server

via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average write requests rate:	Indicates the average number of write commands issued per second on this virtual disk during the collection interval.	Commands/Sec	These measures are good indicators of the request load on the virtual disk. Comparing the values of these measures across virtual disks will enable you to identify overloaded disks.
Average read requests rate:	Indicates the average number of read commands issued per second on this virtual disk during the collection interval.	Commands/Sec	

Measurement	Description	Measurement Unit	Interpretation
Write latency:	Indicates the average amount of time taken for a write to this virtual disk.	Secs	A very high value is indicative of slowdowns in the virtual disk while performing write operations. The disk with the highest write latency can be identified by quickly comparing the value of this measure across virtual disks.
Read latency:	Indicates the average amount of time taken for a read in this virtual disk.	Secs	A very high value is indicative of slowdowns in the virtual disk while performing read operations. The disk with the highest read latency can be identified by quickly comparing the value of this measure across virtual disks.
Read rate:	Indicates the rate at which the reads are performed on this virtual disk.	MB/Sec	
Write rate:	Indicates the rate at which writes are performed on this virtual disk.	MB/Sec	

3.7 The Inside View of VMs Layer

The **Outside View of VMs** layer provides an “outside” view of the different VM guests - the metrics reported at this layer are based on what the VMware ESX server is seeing about the performance of the individual guest VMs. However, an outside view of the VM guest operating system and its applications may not be sufficient. For instance, suppose one of the disk partitions of the guest operating system has reached capacity. This information cannot be gleaned from host operating system. Likewise, bottlenecks such as a longer process run queue or a higher disk queue length are more visible using an internal monitor. Internal monitoring (from within the guest operating system) also provides details about the resource utilization of different application(s) or processes.

The tests mapped to the **Inside View of VMs** layer provide an “inside” view of the workings of each of the guests - these tests execute on an ESX server host, but send probes into each of the guest operating systems to analyze how well each guest utilizes the resources that are allocated to it, and how well it handles user sessions, TCP traffic, and network loading.

By default however, clicking on the **Inside View of VMs** layer, does not display the list of tests associated with that layer. Instead, Figure 3.70 appears. This figure provides you with a list of all VM guests and their respective state (see Figure 3.70).

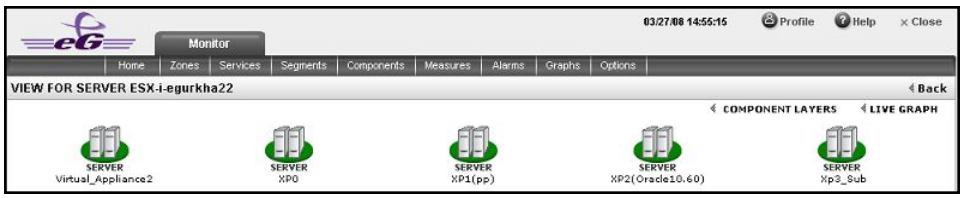


Figure 3.70: A list of guest operating systems on an ESX server host and their current state

To return to the layer model of the *VMware vSphere ESX* server and view the tests associated with the **Inside View of VMs** layer, click on the **COMPONENT LAYERS** link in Figure 3.70. You can now view the list of tests mapped to the **Inside View of VMs** layer, as depicted by Figure 3.71 below.

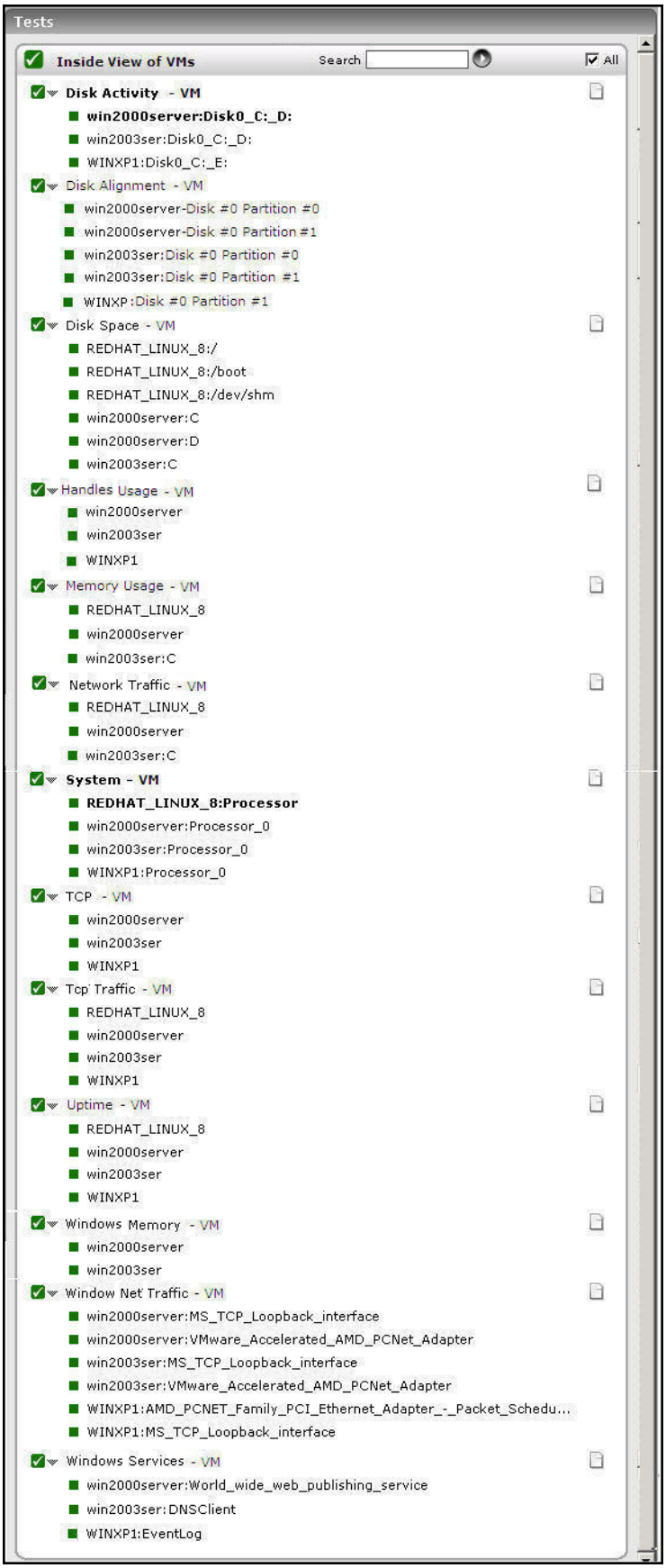

Tests
Inside View of VMs
Search
All
Disk Activity - VM
win2000server:Disk0_C:_D:
win2003ser:Disk0_C:_D:
WINXP1:Disk0_C:_E:
Disk Alignment - VM
win2000server-Disk #0 Partition #0
win2000server-Disk #0 Partition #1
win2003ser:Disk #0 Partition #0
win2003ser:Disk #0 Partition #1
WINXP:Disk #0 Partition #1
Disk Space - VM
REDHAT_LINUX_8:/
REDHAT_LINUX_8:/boot
REDHAT_LINUX_8:/dev/shm
win2000server:C
win2000server:D
win2003ser:C
Handles Usage - VM
win2000server
win2003ser
WINXP1
Memory Usage - VM
REDHAT_LINUX_8
win2000server
win2003ser:C
Network Traffic - VM
REDHAT_LINUX_8
win2000server
win2003ser:C
System - VM
REDHAT_LINUX_8:Processor
win2000server:Processor_0
win2003ser:Processor_0
WINXP1:Processor_0
TCP - VM
win2000server
win2003ser
WINXP1
Tcp Traffic - VM
REDHAT_LINUX_8
win2000server
win2003ser
WINXP1
Uptime - VM
REDHAT_LINUX_8
win2000server
win2003ser
WINXP1
Windows Memory - VM
win2000server
win2003ser
Window Net Traffic - VM
win2000server:MS_TCP_Loopback_interface
win2000server:VMware_Accelerated_AMD_PCNet_Adapter
win2003ser:MS_TCP_Loopback_interface
win2003ser:VMware_Accelerated_AMD_PCNet_Adapter
WINXP1:AMD_PCNET_Family_PCI_Ethernet_Adapter_-_Packet_Schedu...
WINXP1:MS_TCP_Loopback_interface
Windows Services - VM
win2000server:World_wide_web_publishing_service
win2003ser:DNSClient
WINXP1:EventLog

Figure 3.71: The tests associated with the Inside View of Vms layer of a VMware vSphere ESX server

As indicated in Figure 3.71, the tests associated with this layer monitor different aspects of each virtual guest. Disk space utilization, disk activity levels, CPU utilization, memory usage levels, network traffic, etc. are all monitored and reported for each VM guest hosted on the ESX server. Detailed diagnosis for these tests provide details of individual processes and their utilization levels. These tests have been discussed in detail in the subsequent sections.

3.7.1 Disk Activity - VM Test

This test reports statistics pertaining to the input/output utilization of each physical disk on a guest.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every combination of *virtual_guest:disk_partition* or *guest_user:disk_partition*.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80.

Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns,

for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

Note:

While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.
13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **USE SUDO** –**This parameter is of significance to Linux and Solaris VMs only**. By default, the **USE SUDO** parameter is set to **No**. This indicates that, by default, this test will report the detailed diagnosis for the Percent virtual disk busy measure of each disk partition being monitored by executing the */usr/bin/iotop* command or */usr/sbin/iotop* command. However, in some highly secure virtualized environments, this command cannot be executed directly on a Linux. In such cases, set this parameter to **Yes**. This will enable the eG agent to execute the *sudo/usr/bin/iotop* command or *sudo/usr/sbin/iotop* and retrieve the detailed diagnosis of the *Percent virtual disk busy* measure.

17. **USE IOTOP** – The **Disk Activity – VM** test uses the *iotop* command to collect detailed diagnostics on disk activity from Linux VMs. By default, this flag is set to **No**, indicating that the test will not use the *iotop* command – this in turn implies that this test will not report detailed diagnostics for Linux VMs by default. If you want the test to report detailed metrics on disk activity on Linux VMs, set this flag to **Yes.**
18. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Percent virtual disk busy:	Indicates the percentage of elapsed time during which the disk is busy processing requests (i.e., reads or writes).	Percent	Comparing the percentage of time that the different disks are busy, an administrator can determine whether load is properly balanced across the different disks.
Percent reads from virtual disk:	Indicates the percentage of elapsed time that the selected disk drive is busy servicing read requests.	Percent	
Percent writes to virtual disk:	Indicates the percentage of elapsed time that the selected disk drive is busy servicing write requests.	Percent	
Virtual disk read time:	Indicates the average time in seconds of a read of data from the disk.	Secs	
Virtual disk write	Indicates the average time	Secs	

Measurement	Description	Measurement Unit	Interpretation
time:	in seconds of a write of data from the disk.		
Avg. queue for virtual disk:	Indicates the average number of both read and write requests that were queued for the selected disk during the sample interval.	Number	
Current queue for virtual disk:	The number of requests outstanding on the disk at the time the performance data is collected.	Number	This measure includes requests in service at the time of the snapshot. This is an instantaneous length, not an average over the time interval. Multi-spindle disk devices can have multiple requests active at one time, but other concurrent requests are awaiting service. This counter might reflect a transitory high or low queue length, but if there is a sustained load on the disk drive, it is likely that this will be consistently high. Requests experience delays proportional to the length of this queue minus the number of spindles on the disks. This difference should average less than two for good performance.
Reads from virtual disk:	Indicates the number of reads happening on a logical disk per second.	Reads/Sec	A dramatic increase in this value may be indicative of an I/O bottleneck on the guest.
Data reads from virtual disk:	Indicates the rate at which bytes are transferred from the disk during read operations.	KB/Sec	A very high value indicates an I/O bottleneck on the guest.
Writes to virtual disk:	Indicates the number of writes happening on a local disk per second.	Writes/Sec	A dramatic increase in this value may be indicative of an I/O bottleneck on the guest.

Measurement	Description	Measurement Unit	Interpretation
Data writes to virtual disk:	Indicates the rate at which bytes are transferred from the disk during write operations.	KB/Sec	A very high value indicates an I/O bottleneck on the guest.
Disk service time:	Indicates the average time that this disk took to service each transfer request (i.e., the average I/O operation time)	Secs	A sudden rise in the value of this measure can be attributed to a large amount of information being input or output. A consistent increase however, could indicate an I/O processing bottleneck.
Disk queue time:	Indicates the average time that transfer requests waited idly on queue for this disk.	Secs	Ideally, the value of this measure should be low.
Disk I/O time:	Indicates the average time taken for read and write operations of this disk.	Secs	The value of this measure is the sum of the values of the Disk service time and Disk queue time measures. A consistent increase in the value of this measure could indicate a latency in I/O processing.

The detailed diagnosis of the *Percent virtual disk busy* measure, if enabled, provides information such as the Process IDs executing on the disk, the Process names, the rate at which I/O read and write requests were issued by each of the processes, and the rate at which data was read from and written into the disk by each of the processes. In the event of excessive disk activity, the details provided in the detailed diagnosis page will enable users to figure out which process is performing the I/O operation that is keeping the disk busy. **The detailed diagnosis for this test is available for Windows guests only, and not Linux guests**.

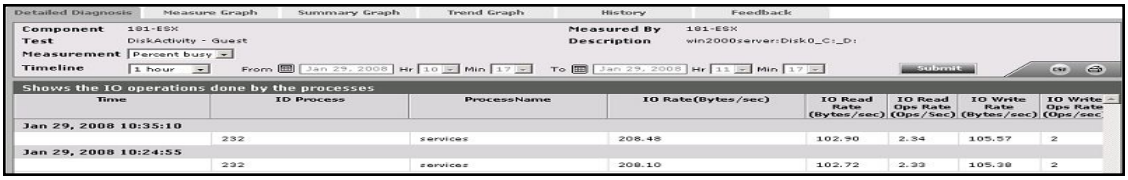


Figure 3.72: The detailed diagnosis of the Percent virtual busy measure

3.7.2 Disk Space - VM Test

This test monitors the space usage of every disk partition on a guest.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every combination of *virtual_guest:disk_partition* or *guest_user:disk_partition*.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the

ESX USER and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with

that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs

configured in the **IGNORE VMS INSIDE VIEW** text box.

11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to

implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total capacity:	Indicates the total capacity of a disk partition; for the Total descriptor, this measure reports the sum of the total capacity of all disk partitions.	MB	

Measurement	Description	Measurement Unit	Interpretation
Used space:	Indicates the amount of space used in a disk partition; for the Total descriptor, this measure reports the sum of space used across all disk partitions.	MB	
Free space:	Indicates the current free space available for each disk partition of a system; for the Total descriptor, this measure reports the sum of the unused space in all disk partitions.	MB	
Percent usage:	Indicates the percentage of space usage on each disk partition of a system; for the Total descriptor, this measure reports the percentage of disk space used across all disk partitions.	Percent	A value close to 100% can indicate a potential problem situation where applications executing on the guest may not be able to write data to the disk partition(s) with very high usage.

3.7.3 System Details - VM Test

This test collects various metrics pertaining to the CPU and memory usage of every processor supported by a guest. The details of this test are as follows:

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every combination of *virtual_guest:processor* or *guest_user:processor*.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.

3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.

4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6**.

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

   Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the

credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list,

then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside

view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
    - If the **INSIDE VIEW USING** flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can

be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to *none*.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **ENABLE MEMORY DIAGNOSIS** - By default, the **ENABLE MEMORY DIAGNOSIS** flag is set to **NO**, indicating that detailed diagnosis will not be available for the *Free memory in VM* measure reported by this test by default. If you want to view the detailed diagnosis of the *Free memory in VM* measure - i.e., to view the top 10 processes on the VM that are utilizing memory excessively - you can change this flag to **YES**.

17. **USE TOP FOR DD** - **This parameter is applicable only to Linux VMs**. By default, this parameter is set to **No**. This indicates that, by default, this test will report the detailed diagnosis of the *Virtual CPU utilization* measure for each processor on a Linux VM by executing the *usr/bin/ps* command. On some Linux flavors however, this command may not function properly. In such cases, set the **USE TOP FOR DD** parameter to **Yes**. This will enable the eG agent to extract the detailed diagnosis of the *Virtual CPU utilization* measure by executing the */usr/bin/top* command instead.

18. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Virtual CPU utilization:	This measurement indicates the percentage of CPU utilized by the processor.	Percent	A high value could signify a CPU bottleneck. The CPU utilization may be high because a few processes are consuming a lot of CPU, or because there are too many processes contending for a limited resource. The detailed diagnosis of this test reveals the top-10 CPU-intensive processes on the guest.
System usage of virtual CPU:	Indicates the percentage of CPU time spent for system-level processing.	Percent	An unusually high value indicates a problem and may be due to too many system- level tasks executing simultaneously.
Run queue in VM:	Indicates the instantaneous length of the queue in which threads are waiting for the processor cycle. This length does not include the threads that are currently being executed.	Number	A value consistently greater than 2 indicates that many processes could be simultaneously contending for the processor.
Blocked processes in VM:	Indicates the number of processes blocked for I/O, paging, etc.	Number	A high value could indicate an I/O problem on the guest (e.g., a slow disk).
Swap memory in VM:	Denotes the committed amount of virtual memory. This corresponds to the space reserved for virtual memory on disk paging file (s).	MB	An unusually high value for the swap usage can indicate a memory bottleneck. Check the memory utilization of individual processes to figure out the process (es) that has (have) maximum memory consumption and look to tune their memory usages and allocations accordingly.
Free memory in VM:	Indicates the free memory available.	MB	A very low value of free memory is also an indication of high memory utilization

Measurement	Description	Measurement Unit	Interpretation
			on a guest. The detailed diagnosis of this measure, if enabled, lists the top 10 processes responsible for maximum memory consumption on the guest.
Scan rate in VM:	Indicates the memory scan rate.	Pages/Sec	A high value is indicative of memory thrashing. Excessive thrashing can be detrimental to guest performance.

Note:

For multi-processor systems, where the CPU statistics are reported for each processor on the system, the statistics that are system-specific (e.g., run queue length, free memory, etc.) are only reported for the "Summary" descriptor of this test.

The detailed diagnosis capability of the *Virtual CPU utilization* measure, if enabled, provides a listing of the top 10 CPU-consuming processes (see Figure 3.73). In the event of a Cpu bottleneck, this information will enable users to identify the processes consuming a high percentage of CPU time. The users may then decide to stop such processes, so as to release the CPU resource for more important processing purposes.

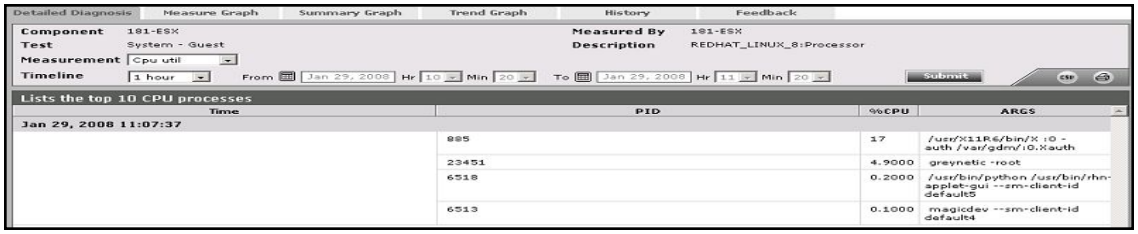


Figure 3.73: The top 10 CPU consuming processes

Note:

While instantaneous spikes in CPU utilization are captured by the eG agents and displayed in the Measures page, the detailed diagnosis will not capture/display such instantaneous spikes. Instead, detailed diagnosis will display only a consistent increase in CPU utilization observed over a period of time.

The detailed diagnosis of the *Free memory in VM* measure, if enabled, lists the top 10 processes responsible for maximum memory consumption on the guest (see Figure 3.74). This information will enable administrators to identify the processes that are causing the depletion in the amount of free memory on the host. The administrators can then decide to kill such expensive processes.

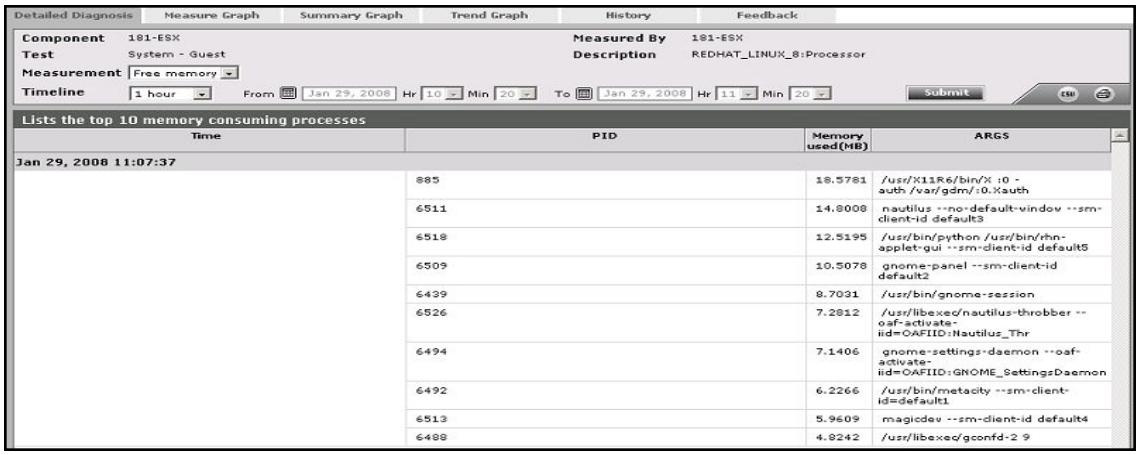


Figure 3.74: The detailed diagnosis of the Free memory measure listing the top 10 memory consuming processes

3.7.4 Uptime - VM Test

In most virtualized environments, it is essential to monitor the uptime of VMs hosting critical server applications in the infrastructure. By tracking the uptime of each of the VMs, administrators can determine what percentage of time a VM has been up. Comparing this value with service level targets, administrators can determine the most trouble-prone areas of the virtualized infrastructure.

In some environments, administrators may schedule periodic reboots of their VM. By knowing that a specific VM has been up for an unusually long time, an administrator may come to know that the scheduled reboot task is not working on a VM.

The Uptime - VM test included in the eG agent monitors the uptime of each VM on a VMware ESX server.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each guest discovered on the ESX server being monitored

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the VmPerl API/CLI**: By default, the eG agent uses the *VmPerl API/ESX CLI*

for monitoring an ESX server and discovering the guests executing on it. Since this monitoring approach is very straight-forward and requires no additional configuration, the **ESX USER** and **ESX PASSWORD** parameters are set to none by default.

- **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section Section **2.6** .

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

  Refer to Section **2.2** to know how to create a user on vCenter.

  If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

  Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to View and Terminate Sessions (from vCenter 4.1, this is called the View and stop sessions permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with

each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

   Note:

   While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

10. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

11. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

12. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

- If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1**.

- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

14. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

15. **REPORTMANAGERTIME** - By default, this flag is set to **Yes**, indicating that, by default, the detailed diagnosis of this test, if enabled, will report the shutdown and reboot times of the VMs in the manager's time zone. If this flag is set to **No**, then the shutdown and reboot times are shown in the time zone of the system where the agent is running (i.e., the system being managed for agent-based monitoring, and the system on which the remote agent is running - for agentless monitoring).
16. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Has the VM been rebooted?:	Indicates whether this guest has been rebooted during the last measurement period or not.	Boolean	If this measure shows 1, it means that the guest was rebooted during the last measurement period. By checking the time periods when this metric changes from 0 to 1, an administrator can determine the times when this guest was rebooted.
Uptime of the VM during the last measure period:	Indicates the time period that the guest has been up since the last time this test ran.	Secs	If the guest has not been rebooted during the last measurement period and the agent has been running continuously, this value will be equal to the measurement period. If the VM was rebooted during the last measurement period, this value will be less than the measurement period of the test. For example, if the measurement period is 300 secs, and if the VM was rebooted 120 secs back, this metric will report a value of 120 seconds. The accuracy of this metric is dependent on the

Measurement	Description	Measurement Unit	Interpretation
			measurement period - the smaller the measurement period, greater the accuracy.
Total uptime of the VM:	Indicates the total time that the guest has been up since its last reboot.		This measure displays the number of years, months, days, hours, minutes and seconds since the last reboot. Administrators may wish to be alerted if a VM has been running without a reboot for a very long period. Setting a threshold for this metric allows administrators to determine such conditions.

Note:

If a value less than a minute is configured as the **TEST PERIOD** of the Uptime - Guest test, then, the **Uptime during the last measure period** measure will report the value 0 for Unix VMs (only) until the minute boundary is crossed. For instance, if you configure the Uptime - Guest test to run every 10 seconds, then, for the first 5 test execution cyles (i.e., 10 x 5 = 50 seconds), the **Uptime during the last measure period** measure will report the value 0 for Unix VMs; however, the sixth time the test executes (i.e, when test execution touches the 1 minute boundary), this measure will report the value 60 seconds for the same VMs. Thereafter, every sixth measurement period will report 60 seconds as the uptime of the Unix VMs. This is because, Unix-based operating systems report uptime only in minutes and not in seconds.

3.7.5 Windows Memory - VM Test

To understand the metrics reported by this test, it is essential to understand how memory is handled by the operating system. On any Windows system, memory is partitioned into a part that is available for user processes, and another that is available to the OS kernel. The kernel memory area is divided into several parts, with the two major parts (called “pools”) being a nonpaged pool and a paged pool. The nonpaged pool is a section of memory that cannot, under any circumstances, be paged to disk. The paged pool is a section of memory that can be paged to disk. (Just being stored in the paged pool doesn’t necessarily mean that something has been paged to disk. It just means that it has either been paged to disk or it could be paged to disk.) Sandwiched directly in between the nonpaged and paged pools (although technically part of the nonpaged pool) is a section of memory called the “System Page Table Entries,” or “System PTEs.” The **Windows Memory - VM** test tracks critical metrics corresponding to the System PTEs and the pool areas of kernel memory of a Windows virtual machine.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every Windows VM guest/user on the monitored ESX server.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for

guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires

that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface

embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1**.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Free entries in system page table:	Indicates the number of page table entries not currently in use by the guest.	Number	The maximum number of System PTEs that a server can have is set when the server boots. In heavily-used servers, you can run out of system PTEs. You can use the registry to increase the number of system PTEs, but that encroaches into the paged pool area, and you could run out of paged pool memory. Running out of either one is bad, and the goal should be to tune your server so that you run out of both at the exact same time. Typically, the value of this metric should be above 3000.
Page read rate in VM:	Indicates the average number of times per second	Reads/Sec	A hard page fault occurs when a program requests a data which is not in

Measurement	Description	Measurement Unit	Interpretation
	the disk was read to resolve hard fault paging.		physical memory. In this case, the operating system finds the specific data on disk and restores it to the physical memory. By tracking the variations to this measure over time, you can keep tabs on hard page faults.
Page write rate in VM:	Indicates the average number of times per second the pages are written to disk to free up the physical memory.	Writes/Sec	
Page input rate in VM:	Indicates the number of times per second that a process needed to access a piece of memory that was not in its working set, meaning that the guest had to retrieve it from the page file.	Pages/Sec	
Page output rate in VM:	Indicates the number of times per second the guest decided to trim a process's working set by writing some memory to disk in order to free up physical memory for another process.	Pages/Sec	This value is a critical measure of the memory utilization on a guest. If this value never increases, then there is sufficient memory in the guest. Instantaneous spikes of this value are acceptable, but if the value itself starts to rise over time or with load, it implies that there is a memory shortage on the guest.
Memory pool non-paged data in VM:	Indicates the total size of the kernel memory nonpaged pool.	MB	The kernel memory nonpage pool is an area of guest memory (that is, memory used by the guest operating system) for kernel objects that cannot be written to disk, but must remain in memory as

Measurement	Description	Measurement Unit	Interpretation
			long as the objects are allocated. Typically, there should be no more than 100 MB of non-paged pool memory being used.
Memory pool paged data in VM :	Indicates the total size of the Paged Pool.	MB	If the Paged Pool starts to run out of space (when it's 80% full by default), the guest will automatically take some memory away from the System File Cache and give it to the Paged Pool. This makes the System File Cache smaller. However, the system file cache is critical, and so it will never reach zero. Hence, a significant increase in the paged pool size is a problem. This metric is a useful indicator of memory leaks in a guest. A memory leak occurs when the guest allocates more memory to a process than the process gives back to the pool. Any time of process can cause a memory leak. If the amount of paged pool data keeps increasing even though the workload on the guest remains constant, it is an indicator of a memory leak.

3.7.6 Windows Network Traffic - VM Test

This is an internal test that monitors the incoming and outgoing traffic through each Windows guest of an ESX server host.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every *Windows_virtual_guest:network_interface* combination or *Windows_VM_guest_user:network_interface* combination

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed

2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6**.

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

   Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing

sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the **SSL** flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the **ESX USER** parameter is set to *none*), then the status of the **SSL** flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside

view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
    - If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be

obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Incoming traffic:	Indicates the rate at which data (including framing characters) is received on a network interface.	Mbps	An abnormally high rate of incoming traffic may require additional analysis.
Outgoing traffic:	Represents the rate at which data (including framing characters) is sent on a network interface.	Mbps	An abnormally high rate of outgoing traffic may require additional analysis.
Maximum bandwidth:	An estimate of the capacity of a network interface.	Mbps	
Bandwidth usage:	Indicates the percentage of bandwidth used by a network interface.	Percent	By comparing the bandwidth usage with the maximum bandwidth of an interface, an administrator can determine times when the network interface is overloaded or is being a performance bottleneck.

Measurement	Description	Measurement Unit	Interpretation
Output queue length:	Indicates the length of the output packet queue (in packets)	Number	If this is longer than 2, delays are being experienced and the bottleneck should be found and eliminated if possible.
Outbound packet errors:	The number of outbound packets that could not be transmitted because of errors	Number	Ideally, number of outbound errors should be 0.
Inbound packet errors:	The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.	Number	Ideally, number of inbound errors should be 0.

If the WindowsNetTraffic - VM test is not reporting measures for a VM, make sure that you have enabled the SNMP service for the VM.

3.7.7 Network Traffic - VM Test

This is an internal test that monitors the incoming and outgoing traffic through each Linux guest on a VMware ESX server.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every *Linux virtual_guest:network_interface* combination.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

- **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required

credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the **SSL** flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the **ESX USER** parameter is set to *none*), then the status of the **SSL** flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the

credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.
13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual

guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

- **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
- If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_*

virtualmachinename.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Incoming network traffic:	Indicates the rate of incoming traffic.	Pkts/Sec	An increase in traffic to the guest can indicate an increase in accesses to the guest (from users or from other applications) or that the guest is under an attack of some form.
Outgoing network traffic:	Represents the rate of outgoing traffic.	Pkts/Sec	An increase in traffic from the guest can indicate an increase in accesses to the guest (from users or from other applications).

3.7.8 Tcp - VM Test

This test tracks various statistics pertaining to TCP connections to and from each guest of an ESX server host. The details of the test are provided below:

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each powered-on guest/logged-in user on the ESX server monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target

ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

- **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to

the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the **SSL** flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the **ESX USER** parameter is set to *none*), then the status of the **SSL** flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP

address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
    - If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified

using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Incoming connections to VM:	Indicates the connections per second received by the guest.	Conns/Sec	A high value can indicate an increase in input load.
Outgoing connections to VM:	Indicates the connections per second initiated by the guest.	Conns/Sec	A high value can indicate that one or more of the applications executing on the guest have started using a number of TCP connections to some other guest or host.
Current connections to VM:	Indicates the currently established connections.	Number	A sudden increase in the number of connections established on a guest can indicate either an increase in load to one or more of the applications executing on the guest, or that one or more of the applications are experiencing a problem (e.g., a slow down). On Microsoft Windows, the current connections metrics is the total number of TCP connections that are currently in the *ESTABLISHED* or *CLOSE_ WAIT* states.
Connection drops	Indicates the rate of	Conns/Sec	This value should be 0 for most of the

Measurement	Description	Measurement Unit	Interpretation
on VM:	established TCP connections dropped from the TCP listen queue.		time. Any non-zero value implies that one or more applications on the guest are under overload.
Connection failures on VM:	Indicates the rate of half open TCP connections dropped from the listen queue.	Conns/Sec	This value should be 0 for most of the time. A prolonged non-zero value can indicate either that the server is under SYN attack or that there is a problem with the network link to the server that is resulting in connections being dropped without completion.

3.7.9 Tcp Traffic - VM Test

Since most popular applications rely on the TCP protocol for their proper functioning, traffic monitoring at the TCP protocol layer can provide good indicators of the performance seen by the applications that use TCP. The most critical metric at the TCP protocol layer is the percentage of retransmissions. Since TCP uses an exponential back-off algorithm for its retransmissions, any retransmission of packets over the network (due to network congestion, noise, data link errors, etc.) can have a significant impact on the throughput seen by applications that use TCP. This test monitors the TCP protocol traffic to and from a guest, and particularly monitors retransmissions.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each powered-on guest/currently logged-in user on the ESX server monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

- **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required

credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the **SSL** flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the **ESX USER** parameter is set to *none*), then the status of the **SSL** flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the

credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.
13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual

guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

- **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
- If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- If the **INSIDE VIEW USING** flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to *none*.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_*

virtualmachinename.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **SEGMENTS SENT MIN** - Specify the minimum threshold for the number of segments sent/transmitted over the network. The default value is 10; in this case, the test will compute/report the *Retransmit ratio from VM* measure only if more than 10 segments are sent over the network – i.e., if the value of the Segments sent by VM measure crosses the value 10. On the other hand, if the *Segments sent by VM* measure reports a value less than 10, then the test will not compute/report the *Retransmit ratio from VM* measure. This is done to ensure that no false alerts are generated by the eG Enterprise system for the *Retransmit ratio from* VM measure. You can change this minimum threshold to any value of your choice.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Segments received by VM:	Indicates the rate at which segments are received by the guest.	Segments/Sec	
Segments sent by VM:	Indicates the rate at which segments are sent to clients or other guests	Segments/Sec	
Retransmits by VM:	Indicates the rate at which segments are being retransmitted by the guest	Segments/Sec	
Retransmit ratio from VM:	Indicates the ratio of the rate of data retransmissions to the rate of data being sent by the guest	Percent	Ideally, the retransmission ratio should be low (< 5%). Most often retransmissions at the TCP layer have significant impact on application performance. Very often a large number of retransmissions are caused by a congested network link, bottlenecks at

Measurement	Description	Measurement Unit	Interpretation
			a router causing buffer/queue overflows, or by lousy network links due to poor physical layer characteristics (e.g., low signal to noise ratio). By tracking the percentage of retransmissions at a guest, an administrator can quickly be alerted to problem situations in the network link(s) to the guest that may be impacting the service performance.

3.7.10 Handles Usage - VM Test

This test monitors and tracks the handles opened by processes running in a target Windows virtual machine.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each powered-on guest/currently logged-in user on the ESX server monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has “Read-only” privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the “Read-only” role to him/her. The steps for achieving

this have been elaborately discussed in Section **2.5** .

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the **SSL** flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the

other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the **ESX USER** parameter is set to *none*), then the status of the **SSL** flag is irrelevant.

Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.
9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to

denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the

ADMIN PASSWORD field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

- **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)**: In this case, specify "*none*" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .

- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to `none`.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **HANDLES GROWTH LIMIT** - This defines the upper limit of the handles opened by any process. By default, this parameter is set to 8000.

16. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report

measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

17. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Handles used by processes of the VM:	Indicates the number of handles opened by various processes running in a target Windows virtual machine in the last measurement period.	Number	Use the detailed diagnosis of this measure to determine the top- 10 processes in terms of number of handles opened. This information brings to light those processes with too many open handles. By closely tracking the handle usage of these processes over time, you can identify potential handle leaks.
Processes using handles above limit in the VM:	Indicates the number of processes that have opened the handles on or above the value defined in the input parameter - **HANDLES GROWTH LIMIT**.	Number	Using the detailed diagnosis of this measure, you can accurately isolate the process (es) that has opened more handles than the permitted limit. A high value of this measure indicates that too many processes are opening handles excessively. You might want to closely observe the handle usage of these processes over time to figure out whether the spike in usage is sporadic or consistent. A consistent increase in

Measurement	Description	Measurement Unit	Interpretation
			handle usage could indicate a handle leak.

The detailed diagnosis of the *Handles used by processes* measure, if enabled, lists the names of top-10 processes in terms of handle usage, the number of handles each process uses, the process ID, and the ID of the parent process.

List of top 10 processes in a VM that are holding handles

Time	Process Name	Handles used	Process ID	Parent PID
Jan 29, 2009 12:00:49				
	System	3359	0	4
	js	1718	540	6420
	svchost	1208	540	1012
	lsass	1112	492	552
	csrss	1097	420	468
	winlogon	564	420	492
	ImaSrv	559	540	3696
	Rtvscan	536	540	3936
	tomcat	485	540	6572
	services	482	492	540

Figure 3.75: The detailed diagnosis of the Handles used by processes measure

The detailed diagnosis of the *Processes using handles above limit in VM* measure, if enabled, lists the details of processes that are using more handles than the configured limit.

List of processes in a VM that are using handles above the configured handle growth value

Time	Process Name	Handles used	Process ID	Parent PID
Jan 29, 2009 17:54:18	eGRSvc	62410	412	11512

Figure 3.76: The detailed diagnosis of the Processes using handles above limit in VM measure

3.7.11 Windows Services - VM Test

This test tracks the status (whether running or have stopped) of services executing on Windows virtual machines.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each powered-on guest/currently logged-in user on the ESX server monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.

4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

   Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1,

this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the

VIRTUAL CENTER list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using

the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
    - If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **IGNORESERVICES** - Provide a comma-separated list of services that need to be ignored while monitoring. When configuring a service name to exclude, make sure that you specify the **Display Name** of the service, and not the service **Name** you see in the **Services** window on your Windows VM.

17. **TOTAL AUTO SERVICES STARTED DD** – By default, this flag is set to **No**. Accordingly, this test, by default, will not report detailed diagnostics for the *Total automatic services started* measure. To view the list of services with startup type as 'Automatic', which are in the 'Started' state in each of the Windows VMs of the vSphere host, set this flag to **Yes**.

18. **TOTAL MANUAL SERVICES STARTED DD** - By default, this flag is set to **NO**. Accordingly, this test, by default, will not report detailed diagnostics for the *Total manual services started* measure. To view the list of services with startup type as 'Manual', which are in the 'Started' state in each of the Windows VMs of the vSphere host, set this flag to **Yes**.

19. **TOTAL MANUAL SERVICES STOPPED DD** - By default, this flag is set to **No**. Accordingly, this test, by default, will not report detailed diagnostics for the *Total manual services stopped* measure. To view the list of services with startup type 'Manual', which are in the 'Stopped' state in each of the Windows VMs of the vSphere host, set this flag to **Yes**.

20. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

21. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

- The eG manager license should allow the detailed diagnosis capability
- Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New automatic services started:	Indicates the number of Windows services with startup type as *automatic*, which were running in the last measurement period.	Number	The detailed diagnosis of this measure lists the services (with startup type as *1*) that are running.
New automatic services stopped:	Indicates the number of Windows services with startup type as *automatic*, which were not running in the last measurement period.	Number	To know which services stopped, use the detailed diagnosis of this measure (if enabled).
New manual services started:	Indicates the number of Windows services with startup type as *manual*, which were running in the last measurement period.	Number	Use the detailed diagnosis of this measure to identify the *manual* services that are running.
New manual services stopped:	Indicates the number of Windows services with startup type as manual, which stopped running in the last measurement period.	Number	To identify the services that stopped, use the detailed diagnosis of this measure.
Total automatic services stopped:	Indicates the total number of Windows services with startup type as *automatic*, which are not running in this VM presently.	Number	To identify the services that stopped, use the detailed diagnosis of this measure.

Measurement	Description	Measurement Unit	Interpretation
Total automatic services started:	Indicates the total number of Windows services with startup type as *automatic*, which are running in this VM currently.	Number	If the **TOTAL AUTO SERVICES STARTED DD** flag is set to **Yes**, then the test will report detailed diagnostics for this measure. The detailed diagnosis will reveal the names of the services with startup type as automatic, which are running in the VM now.
Total manual services stopped:	Indicates the total number of Windows services with startup type as *manual*, which are not running in this VM presently.	Number	If the **TOTAL MANUAL SERVICES STOPPED DD** flag is set to **Yes**, then the test will report detailed diagnostics for this measure. The detailed diagnosis will reveal the names of the services with startup type as *manual*, which are not running in the VM now.
Total manual services started:	Indicates the total number of Windows services with startup type as *manual*, which are not running in this VM presently.	Number	If the **TOTAL MANUAL SERVICES STARTED DD** flag is set to **Yes**, then the test will report detailed diagnostics for this measure. The detailed diagnosis will reveal the names of the services with startup type as *manual*, which are running in the VM now.

3.7.12 Memory Usage - VM Test

This test reports statistics related to the usage of physical memory of the VMs.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each powered-on guest/currently logged-in user on the ESX server monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.

4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

   Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1,

this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the

VIRTUAL CENTER list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for

more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
    - If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin

password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
16. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total memory:	Indicates the total memory of this VM.	MB	
Used memory:	Indicates the used memory of this VM.	MB	
Free memory:	Indicates the free memory of the VM.	MB	This measure typically indicates the amount of memory available for use by applications running on the target VM.

Measurement	Description	Measurement Unit	Interpretation
			On Unix operating systems (AIX and Linux), the operating system tends to use parts of the available memory for caching files, objects, etc. When applications require additional memory, this is released from the operating system cache. Hence, to understand the true free memory that is available to applications, the eG agent reports the sum of the free physical memory and the operating system cache memory size as the value of the Free physical memory measure while monitoring AIX and Linux guest operating systems.
Memory utilized:	Indicates the percent usage of memory by this VM.	Percent	Ideally, the value of this measure should be low. While sporadic spikes in memory usage could be caused by one/more rogue processes on the VM, a consistent increase in this value could be a cause for some serious concern, as it indicates a gradual, but steady erosion of valuable memory resources. If this unhealthy trend is not repaired soon, it could severely hamper VM performance, causing anything from a slowdown to a complete system meltdown. You can use the detailed diagnosis of this measure to figure out which processes on the VM are consuming memory excessively.
Available memory:	Indicates the amount of memory, immediately available for allocation to a process or for system use.	MB	Not all of the Available memory is Free memory. Typically, Available memory is made up of the Standby List, Free List, and Zeroed List. When Windows wants to trim a process' working set, the trimmed

Measurement	Description	Measurement Unit	Interpretation
			pages are moved (usually) to the Standby List. From here, they can be brought back to life in the working set with only a soft page fault (much faster than a hard fault, which would have to talk to the disk). If a page stays in the standby List for a long time, it gets freed and moved to the Free List. In the background, there is a low priority thread (actually, the only thread with priority 0) which takes pages from the Free List and zeros them out. Because of this, there is usually very little in the Free List. All new allocations always come from the Zeroed List, which is memory pages that have been overwritten with zeros. This is a standard part of the OS' cross-process security, to prevent any process ever seeing data from another. If the Zeroed List is empty, Free List memory is zeroed and used or, if that is empty too, Standby List memory is freed, zeroed, and used. It is because all three can be used with so little effort that they are all counted as "available". A high value is typically desired for this measure. **This measure will be available for Windows 2008 VMs only.**
Modified memory:	Indicates the amount of memory that is allocated to the modified page list.	MB	This memory contains cached data and code that is not actively in use by processes, the system and the system cache. This memory needs to be written out before it will be available for allocation to a process or for system use.

Measurement	Description	Measurement Unit	Interpretation
			Cache pages on the modified list have been altered in memory. No process has specifically asked for this data to be in memory, it is merely there as a consequence of caching. Therefore it can be written to disk at any time (not to the page file, but to its original file location) and reused. However, since this involves I/O, it is not considered to be Available physical memory. **This measure will be available for Windows 2008 VMs only.**
Standby memory:	Indicates the amount of memory assigned to the standby list.	MB	This memory contains cached data and code that is not actively in use by processes, the system and the system cache. It is immediately available for allocation to a process or for system use. If the system runs out of available free and zero memory, memory on lower priority standby cache page lists will be repurposed before memory on higher priority standby cache page lists. Typically, Standby memory is the aggregate of Standby Cache Core Bytes,Standby Cache Normal Priority Bytes, and Standby Cache Reserve Bytes. Standby Cache Core Bytes is the amount of physical memory, that is assigned to the core standby cache page lists. Standby Cache Normal Priority Bytes is the amount of physical memory, that is assigned to the normal priority standby cache page lists. Standby Cache Reserve Bytes is the amount of physical memory, that is assigned to the reserve standby cache page lists. **This measure will be available for**

Measurement	Description	Measurement Unit	Interpretation
			Windows 2008 VMs only.
Cached memory:	This measure is an aggregate of Standby memory and Modified memory.	MB	**This measure will be available for Windows 2008 VMs only.**

Note:

While monitoring Linux/AIX guest operating systems, you may observe discrepancies between the value of the *Physical memory utilized* measure and the memory usage percentages reported per process by the detailed diagnosis of the same measure. This is because, while the *Physical memory utilized* measure takes into account the memory in the OS cache of the Linux/AIX VM, the memory usage percent that the detailed diagnosis reports per process does not consider the OS cache memory.

3.7.13 Disk Alignment - VM Test

In a SAN environment, the smallest hardware unit used by a SAN storage array to build a LUN out of multiple physical disks is called a chunk or a stripe. To optimize I/O, chunks are usually much larger than sectors. Thus, a SCSI I/O request that intends to read a sector in reality reads one chunk.

On top of this, in a Windows environment, NTFS is formatted in blocks ranging from 1MB to 8MB. The file system used by the guest operating system optimizes I/O by grouping sectors into so called clusters (allocation units).

Figure 3.77 shows these three layers at issue. There are the SAN blocks at the bottom, then the VMFS blocks in the middle, and then the NTFS blocks used by the Windows VM.

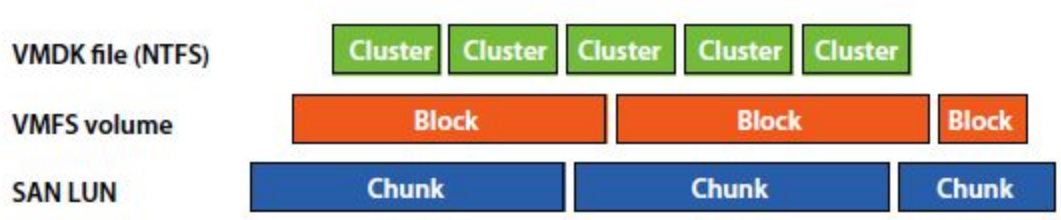


Figure 3.77: The SAN, VMFS, and NTFS blocks

If these three layers are not aligned, your SAN may be working harder than it needs to. For example, a call to read a single NTFS block may require the SAN to read three blocks as shown below:

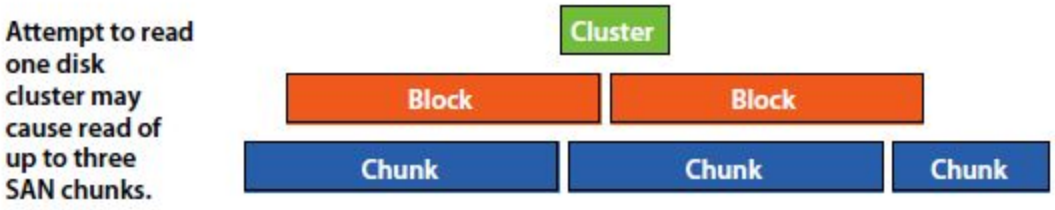


Figure 3.78: Unaligned partitions

An unaligned partition therefore, results in a track crossing and an additional I/O, incurring a penalty on latency and throughput. The additional I/O (especially if small) can impact system resources significantly on some host types.

What would hence be ideal is for the three layers in Figure 3.78 above to be aligned so that a single NTFS block requires only one SAN block to be read as illustrated below:

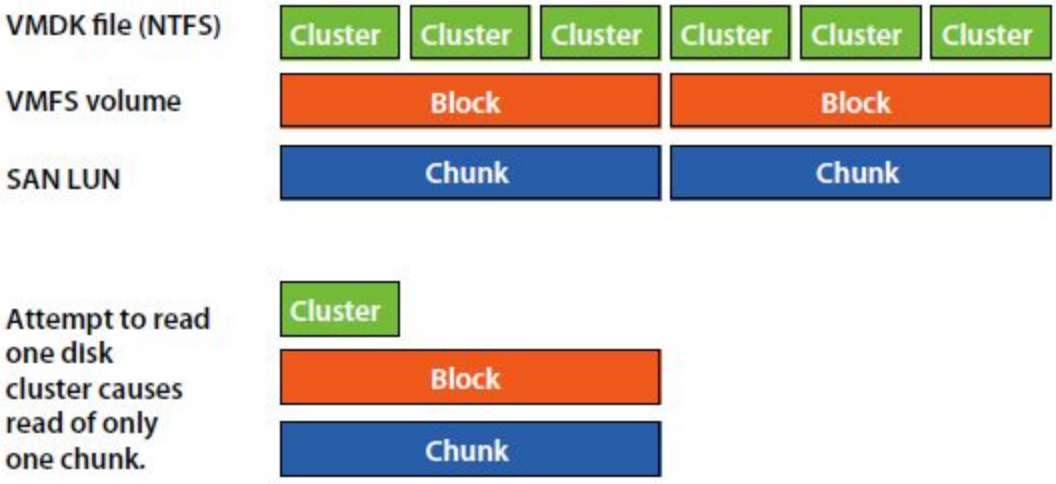


Figure 3.79: Aligned partitions

An aligned partition ensures that the single I/O is serviced by a single device, eliminating the additional I/O and resulting in overall performance improvement.

Therefore, whenever users to Windows VMs complaint that the VM is running slower than usual, you may want to check the disk alignment to determine whether the slowdown can be attributed to one/more unaligned disk partitions. This test enables you to perform such a check.

Note:

This test reports metrics for Windows VMs only.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each disk partition on every Windows VM on a vSphere/ESX host being monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for

guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires

that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface

embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .

- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
16. **IGNORESERVICES** - Provide a comma-separated list of services that need to be ignored while monitoring.
17. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Disk partition alignment status:	Indicates whether this disk partition is aligned or not.		If the partition is unaligned, this test reports the value *Partition is not aligned*.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>For an aligned partition, this test reports the value Partition is aligned.

The numeric values that correspond to the above-mentioned measure values are described in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Partition is aligned</td><td>100</td></tr>
<tr><td>Partition is not aligned</td><td>0</td></tr>
</table>

Note:

By default, this measure reports one of the Measure Values listed in the table above. The graph of this measure however will represent the disk alignment status using the numeric equivalents - 100 or 0.

If a partition is found to be misaligned, you can use the detailed diagnosis of this test to figure out the caption, device ID, logical partition name, and block size of the faulty partition.</td></tr>
</table>

If a partition is found to be misaligned, you can use the detailed diagnosis of this test to figure out the caption, device ID, logical partition name, and block size of the faulty partition.

Details of physical disk partition

Time	Caption	Device id	Logical partition name	Block size
May 31, 2011 17:38:37	VMware Virtual disk SCSI Disk Device	\\.\PHYSICALDRIVE0	C:	512

Figure 3.80: The detailed diagnosis of the Disk partition alignment status measure

3.7.14 GPU - VM Test

GPU-accelerated computing is the use of a graphics processing unit (GPU) together with a CPU to accelerate scientific, analytics, engineering, consumer, and enterprise applications. GPU-accelerated computing enhances application performance by offloading compute-intensive portions of the application to the GPU, while the remainder of the code still runs on the CPU. Architecturally, while a CPU has only few cores and

handles few hundred threads at a time, a GPU is composed of hundreds of cores that can handle thousands of threads simultaneously and render a flawless rich graphics experience.

Now, imagine if you could access your GPU-accelerated applications, even those requiring intensive graphics power, anywhere on any device. **NVIDIA GRID** makes this possible. With NVIDIA GRID, a virtualized GPU designed specifically for virtualized server environments, data center managers can bring true PC graphics-rich experiences to users.

The NVIDIA GRID GPUs will be hosted in enterprise data centers and allow users to run virtual desktops or virtual applications on multiple devices connected to the internet and across multiple operating systems, including PCs, notebooks, tablets and even smartphones. Users can utilize their online-connected devices to enjoy the GPU power remotely.

In VDI/virtualized server environments, the NVIDIA GRID delivers GPU resources to virtual desktops/VMs. This way, graphics can be rendered on a virtual machine's (VM's) host server rather than on a physical end-point device. This technology now makes it possible to use virtual desktop technology to support users accessing graphics intensive workloads. There are two modes of making GPU resources available to virtual desktops:

- **Dedicated GPU or GPU Pass-through Technology:** NVIDIA GPU pass-through technology lets you create a virtual workstation that gives users all the benefits of a dedicated graphics processor at their desk. By directly connecting a dedicated GPU to a virtual machine through the hypervisor, you can now allocate the full GPU and graphics memory capability to a single virtual machine without any resource compromise.

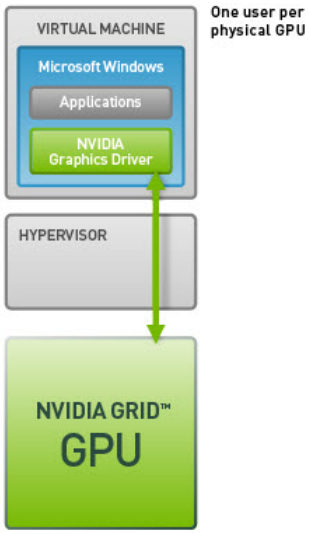


Figure 3.81: Dedicated GPU Technology

- **Shared GPU or Virtual GPU (vGPU) Technology:** GRID vGPU is the industry's most advanced technology for sharing true GPU hardware acceleration between multiple virtual desktops—without compromising the graphics experience. With GRID vGPU technology, the graphics commands of each virtual machine are passed directly to the GPU, without translation by the hypervisor. This allows the GPU hardware to be time-sliced to deliver improved shared virtualized graphics performance. The GRID vGPU manager allows for management of user profiles. IT managers can assign the optimal amount of

graphics memory and deliver a customized graphics profile to meet the specific needs of each user. Every virtual desktop has dedicated graphics memory, just like they would at their desk, so they always have the resources they need to launch and run their applications.

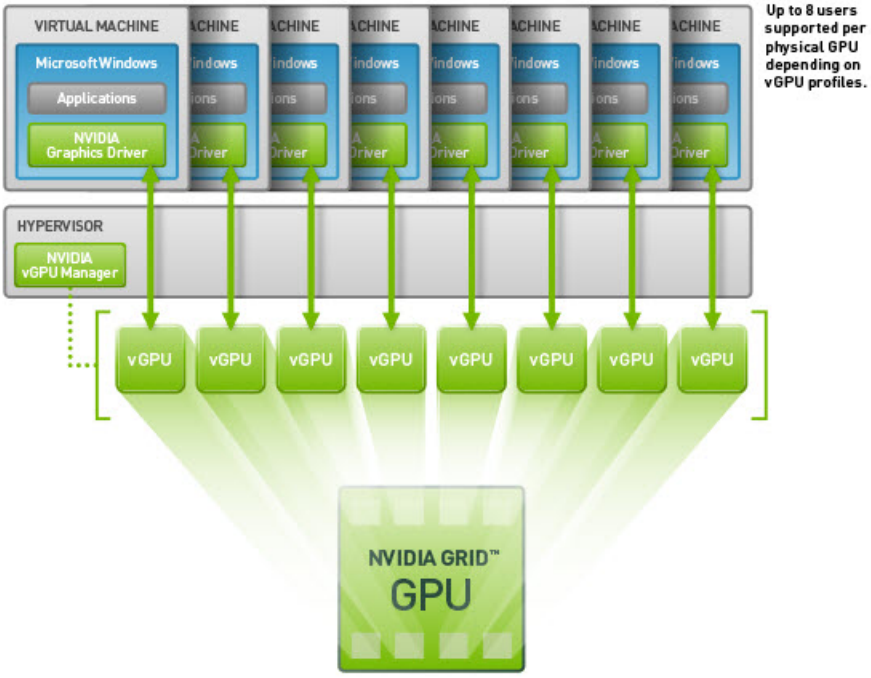


Figure 3.82: Shared vGPU Technology

In GPU-enabled Citrix XenServer environments, if users to VMs/virtual desktops complain of slowness when accessing graphic applications, administrators must be able to instantly detect the slowness and figure out its root-cause – is it because adequate GPU resources are not allocated to the VMs/virtual desktops? Is it because of excessive utilization of GPU memory and processing resources by a few VMs/virtual desktops? Or is it because the GPU clock frequencies are improperly set for one/more GPUs used by a VM/virtual desktop?

- The VMs/virtual desktops have been allocated enough vGPUs;
- The vGPUs are configured with enough graphics memory;
- The vGPU clock frequencies are rightly set;
- The GPU technology in use – i.e., the GPU Pass-through technology or the Shared vGPU technology – is ideal for the graphics processing requirements of the environment;

Measures to right-size the host and fine-tune its GPU configuration can be initiated based on the results of this analysis. This is exactly what the **GPU – VM** test helps administrators achieve!

This test tracks the rate at which each vGPU processes frames, and thus pinpoints those vGPUs that are experiencing a processing bottleneck. The test also monitors the memory usage on each vGPU and helps administrators identify the vGPUs where memory is over-used. The test also reveals how each of these VMs/virtual desktops use each of the allocated vGPUs, thus enabling administrators to determine whether/not the allocated vGPUs are sufficient for the current and future processing requirements of the

VMs/virtual desktops. In the process, the test also pinpoints those VMs/virtual desktops that are over-utilizing the graphical processors assigned to them. Also, to make sure that the assigned vGPUs are functioning without a glitch, the power consumption, temperature, and clock frequency of each vGPU is also checked at periodic intervals, so that abnormalities can be quickly detected.

Note:

This test will report metrics for only those Windows VMs where the **NVWMI** is installed. The steps for installing **NVWMI** and configuring the eG agent to use it have been detailed in the *Monitoring Citrix XenServers* document.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results every vGPU assigned to each Windows VM on the vSphere/ESX server being monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has “Read-only” privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the “Read-only” role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects

to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port

443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing

spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

Note:

While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux

guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1**.

- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
16. **IGNORESERVICES** - Provide a comma-separated list of services that need to be ignored while monitoring.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Cooler rate:	Indicates the percentage of device cooler rate for this GPU of this VM/virtual desktop.	Percent	
Virtual GPU compute utilization:	Indicates the proportion of time over the past sample period during which one or more kernels was executing on this vGPU of this VM/virtual desktop.	Percent	A value close to 100% indicates that the vGPU of the VM/virtual desktop is busy processing graphic requests almost all the time. In a Shared vGPU environment, the same vGPU card can be allocated to multiple VMs on a server. In such cases, you can quickly compare the value of this measure across VMs that share the same vGPU card to know which VM is utilizing that card most of the time. fa vGPU may be in use almost all the time, if the VM/virtual desktop it is allocated to runs graphic- intensive applications. A resource-hungry VM/virtual desktop can impact the performance of other VMs/virtual desktops that are sharing the same vGPU card. If you find that only a single VM/virtual desktop has been consistently hogging the GPU resources, you may want to switch to the Dedicated GPU mode, so that excessive GPU usage by that VM/virtual desktop has no impact on the performance of other VMs/virtual desktops on that host. If all GPUs assigned to a VM/virtual desktop are found to be busy most of the time, you may want to consider allocating more GPU resources to that VM/virtual desktop.
Power	Indicates the current power	Watts	A very high value is indicative of

Measurement	Description	Measurement Unit	Interpretation
consumption:	usage of this GPU allocated to this VM/virtual desktop.		excessive power usage by the GPU. Compare the value of this measure across GPUs to know which VM's/virtual desktop's GPU is consuming power excessively.
Core GPU temperature:	Indicates the current temperature of this GPU allocated to this VM/virtual desktop.	Celsius	Ideally, the value of this measure should be low. A very high value is indicative of abnormal GPU temperature. Compare the value of this measure across VMs/virtual desktops to identify that VM/virtual desktop for which GPU temperature soared since the last reading. To reduce the heat output of the GPU and consequently its temperature, you may consider performing underclocking. For instance, it is possible to set a GPU to run at lower clock rates when performing everyday tasks (e.g. internet browsing and word processing), thus allowing the card to operate at lower temperature and thus lower, quieter fan speeds.
	Total framebuffer memory: Indicates the total size of frame buffer memory of this GPU of this VM/virtual desktop.	MB	Frame buffer memory refers to the memory used to hold pixel properties such as color, alpha, depth, stencil, mask, etc.
Used frame buffer memory:	Indicates the amount of frame buffer memory on-board this GPU that is being used by this VM/virtual desktop.	MB	Frame buffer memory refers to the memory used to hold pixel properties such as color, alpha, depth, stencil, mask, etc. Properties like the screen resolution,

Measurement	Description	Measurement Unit	Interpretation
			color level, and refresh speed of the frame buffer can impact graphics performance. Also, if *Error-correcting code (ECC)* is enabled, the available frame buffer memory may be decreased by several percent. This is because, ECC uses up memory to detect and correct the most common kinds of internal data corruption. Moreover, the driver may also reserve a small amount of memory for internal use, even without active work on the GPU; this too may impact frame buffer memory. For optimal graphics performance therefore, adequate frame buffer memory should be allocated to the VM/virtual desktop.
Free frame buffer memory:	Indicates the amount of frame buffer memory on-board this GPU that is yet to be used by this VM/virtual desktop.	MB	
Frame buffer memory utilization:	Indicates the percentage of total frame buffer memory that has been allocated to this VM/virtual desktop.	Percent	Ideally, the value of this measure should be low. A value close to 100% is indicative of excessive usage of frame buffer memory. Properties like the screen resolution, color level, and refresh speed of the frame buffer can impact graphics performance. Also, if Error-correcting code (ECC) is enabled, the frame buffer memory usage will increase by several percent. This is

Measurement	Description	Measurement Unit	Interpretation
			because, ECC uses up memory to detect and correct the most common kinds of internal data corruption. Moreover, the driver may also reserve a small amount of memory for internal use, even without active work on the GPU; this too may impact frame buffer memory usage. For optimal graphics performance therefore, adequate frame buffer memory should be allocated to the VM/virtual desktop.
Virtual memory:	Indicates the virtual memory of this GPU device of this VM/virtual desktop.	MB	
GPU memory utilization:	Indicates the proportion of time over the past sample period during which global (device) memory was being read or written on this GPU of this VM/virtual desktop.	Percent	A value close to 100% is a cause for concern as it indicates that the graphics memory on a GPU is almost always in use. In a Shared vGPU environment, memory may be consumed all the time if one/more VMs/virtual desktops utilize the graphics memory excessively and constantly. If you find that only a single VM/virtual desktop has been consistently hogging the graphic memory resources, you may want to switch to the Dedicated GPU mode, so that excessive memory usage by that VM/virtual desktop has no impact on the performance of other VMs/virtual desktops on that host. If the value of this measure is high almost all the time for most of the GPUs, it could mean that the VM/virtual desktop is not sized with adequate graphics

Measurement	Description	Measurement Unit	Interpretation
			memory.
Total BAR1 memory:	Indicates the total size of the BAR1 memory of this GPU allocated to this VM/virtual desktop.	MB	BAR1 is used to map the frame buffer (device memory) so that it can be directly accessed by the CPU or by 3rd party devices (peer-to-peer on the PCIe bus).
Used BAR1 memory:	Indicates the amount of BAR1 memory on this GPU that is being used by this VM/virtual desktop.	MB	For better user experience with graphic applications, enough BAR1 memory should be available to the VM/virtual desktop.
BAR1 memory utilization:	Indicates the percentage of the allocated BAR1 memory that is currently being utilized by this VM/virtual desktop.	Percent	A value close to 100% is indicative of excessive usage of the BAR1 memory by a VM/virtual desktop. For best graphics performance, this value should be low. To ensure that, adequate BAR1 memory should be allocated to the VM.
Free BAR1 memory:	Indicates the total size of BAR1 memory of this GPU that is still not used by this VM/virtual desktop.	MB	
Power management:	Indicates whether/not power management is enabled for this GPU of this VM/virtual desktop.	q	Many **NVIDIA** graphics cards support multiple performance levels so that the server can save power when full graphics performance is not required. The default Power Management Mode of the graphics card is Adaptive. In this mode, the graphics card monitors GPU usage and seamlessly switches between modes based on the performance demands of the application. This allows the GPU to always use the minimum amount of power required to run a given application. This mode is recommended by NVIDIA for best overall balance of

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>power and performance. If the power management mode is set to Adaptive, the value of this measure will be Supported.

Alternatively, you can set the Power Management Mode to Maximum Performance. This mode allows users to maintain the card at its maximum performance level when 3D applications are running regardless of GPU usage. If the power management mode of a GPU is Maximum Performance, then the value of this measure will be Maximum.

The numeric values that correspond to these measure values are discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Supported</td><td>1</td></tr><tr><td>Maximum</td><td>0</td></tr></table>
Note:

By default, this measure will report the Measure Values listed in the table above to indicate the power management status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Power limit:</td><td>Indicates the power limit configured for this GPU of this VM/virtual desktop.</td><td>Watts</td><td>This measure will report a value only if the value of the ‘Power management’ measure is ‘Supported’.

The power limit setting controls how much voltage a GPU can use when under load. Its not advisable to set the power limit at its maximum – i.e., the value of this measure should not be the</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			same as the value of the Max power limit measure - as it can cause the GPU to behave strangely under duress.
Default power limit:	Indicates the default power management algorithm's power ceiling for this GPU.	Watts	This measure will report a value only if the value of the 'Power management' measure is 'Supported'.
Enforced power limit:	Indicates the power management algorithm's power ceiling for this GPU of this VM/virtual desktop.	Watts	**This measure will report a value only if the value of the 'Power management' measure is 'Supported'**. The total board power draw is manipulated by the power management algorithm such that it stays under the value reported by this measure.
Min power limit:	The minimum value that the power limit be set to for this GPU of this VM/virtual desktop.	Watts	**This measure will report a value only if the value of the 'Power management' measure is 'Supported'**.
Max power limit:	The maximum value that the power limit for this GPU of this VM/virtual desktop can be set to.	Watts	**This measure will report a value only if the value of the 'Power management' measure is 'Supported'**. If the value of this measure is the same as that of the Power limit measure, then the GPU may behave strangely.
Core clock:	Indicates current frequency of the graphics clock on this GPU of this VM/virtual desktop.	MHz	GPU has many more cores than your average CPU but these cores are much simpler and much smaller so that many more actually fit on a small piece of silicon. These smaller, simpler cores go

Measurement	Description	Measurement Unit	Interpretation
			by different names depending upon the tasks they perform. Stream processors are the cores that perform a single thread at a slow rate. But since GPUs contain numerous stream processors, they make overall computation high. The streaming multiprocessor clock is how fast the stream processors run. The memory clock is how fast the memory on the card runs. The GPU core clock is the speed at which the GPU assigned to the VM/virtual desktop operates. By correlating the frequencies of these clocks – i.e., the value of these measures - with the memory usage, power usage, and overall performance of the GPU, you can figure out if overclocking is required or not. Overclocking is the process of forcing a GPU core/memory to run faster than its manufactured frequency. Overclocking can have both positive and negative effects on GPU performance. For instance, memory overclocking helps on cards with low memory bandwidth, and with games with a lot of post-processing/textures/filters like AA that are VRAM intensive. On the other hand, speeding up the operation frequency of a shader/streaming processor/memory clock, without properly analyzing its need and its effects, may increase its thermal output in a linear fashion. At the same time, boosting voltages will cause the generated heat to sky rocket. If improperly managed, these increases in temperature can cause permanent physical damage to the core/memory or even “heat death”. Putting an adequate cooling system into place, adjusting the power provided to the GPU, monitoring your results with the right tools and doing the necessary research are all critical steps on the path to safe and successful overclocking.

Measurement	Description	Measurement Unit	Interpretation
Memory clock:	Indicates current memory clock frequency on this GPU of this VM/virtual desktop.	MHz	
Streaming multiprocessor clock:	Indicates the current frequency of the streaming multiprocessor clock on this GPU of this VM/virtual desktop.	MHz	
Frame rate:	Indicates the rate at which frames are processed by this GPU of this VM/virtual desktop.	Frames/Sec	FPS is how fast your graphics card can output individual frames each second. It is the most time- tested and ideal measure of performance of a GPU. Higher the value of this measure, healthier is the GPU. This measure will be reported only if the 'Dedicated GPU' or 'GPU Pass-through' technology is used to deliver GPU resources to virtual desktops/VMs.
Fan speed:	Indicates the percent of maximum speed that this GPU's fan is currently intended to run at.	Percent	This measure will be reported only if the 'Dedicated GPU' or 'GPU Pass-through' technology is used to deliver GPU resources to virtual desktops/VMs. The value of this measure could range from 0 to 100%. An abnormally high value for this measure could indicate a problem condition – eg., a sudden surge in the temperature of the GPU that could cause the fan to spin faster. Note that the reported speed is only the intended fan speed. If the fan is physically blocked and unable to spin, this output will not match the actual fan speed. Many parts do not report fan

Measurement	Description	Measurement Unit	Interpretation
			speeds because they rely on cooling via fans in the surrounding enclosure. By default the fan speed is increased or decreased automatically in response to changes in temperature.
Compute processes:	Indicates the number of processes having compute context on this GPU of this VM.	Number	Use the detailed diagnosis of this measure to know which processes are currently using the GPU. The process details provided as part of the detailed diagnosis include, the PID of the process, the process name, and the GPU memory used by the process. Note that the GPU memory usage of the processes will not be available in the detailed diagnosis, if the Windows platform on which XenApp operates is running in the WDDM mode. In this mode, the Windows KMD manages all the memory, and not the NVIDIA driver. Therefore, the NVIDIA SMI commands that the test uses to collect metrics will not be able to capture the GPU memory usage of the processes.
Volatile single bit errors:	Indicates the number of volatile single bit errors in this GPU of this VM/virtual desktop.	Number	**This measure will be reported only if the 'Dedicated GPU' or 'GPU Pass-through' technology is used to deliver GPU resources to virtual desktops/VMs**. Volatile error counters track the number of errors detected since the last driver load. Single bit ECC errors are automatically corrected by the hardware and do not result in data corruption. Ideally, the value of this measure should be 0.

Measurement	Description	Measurement Unit	Interpretation
Volatile double bit errors:	Indicates the total number of volatile double bit errors in this GPU of this VM/virtual desktop.	Number	**This measure will be reported only if the 'Dedicated GPU' or 'GPU Pass-through' technology is used to deliver GPU resources to virtual desktops/VMs**. Volatile error counters track the number of errors detected since the last driver load. Double bit errors are detected but not corrected. Ideally, the value of this measure should be 0.
Aggregate single bit errors:	Indicates the total number of aggregate single bit errors in this GPU of this VM/virtual desktop.	Number	**This measure will be reported only if the 'Dedicated GPU' or 'GPU Pass-through' technology is used to deliver GPU resources to virtual desktops/VMs**. Aggregate error counts persist indefinitely and thus act as a lifetime counter. Single bit ECC errors are automatically corrected by the hardware and do not result in data corruption. Ideally, the value of this measure should be 0.
Aggregate double bit errors:	Indicates the total number of aggregate double bit errors in this GPU of this VM/virtual desktop.	Number	**This measure will be reported only if the 'Dedicated GPU' or 'GPU Pass-through' technology is used to deliver GPU resources to virtual desktops/VMs**. Aggregate error counts persist indefinitely and thus act as a lifetime counter. Double bit errors are detected but not corrected. Ideally, the value of this measure should be 0.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Mode:</td><td>Indicates the mode using which the GPU resources were delivered to this VMs.</td><td></td><td>The values that this measure can take and their corresponding numeric values are as follows:<table><tr><th>Measure Value</th><th>Numeric Values</th></tr><tr><td>Pass through</td><td>0</td></tr><tr><td>Shared</td><td>1</td></tr><tr><td>Unavailable (GPU card is not allocated to any VM)</td><td>2</td></tr></table>Note:
By default, this test reports the Measure Values listed in the table above to indicate the mode of GPU delivery. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Physical GPU compute utilization:</td><td>Indicates the proportion of time over the past sample period during which one or more kernels were executing on the physical GPU of this VM/virtual desktop.</td><td>Percentage</td><td>This measure will report metrics only for the Tesla GPU card.
A value close to 100% indicates that the physical GPU is busy processing graphic requests from this VM almost all the time.
In a Shared vGPU environment a GPU may be in use almost all the time, if the VM/virtual desktop it is allocated to run graphic- intensive applications. A resource-hungry VM/virtual desktop on a server can impact the performance of other VMs/virtual desktops on the same server. If you find that only a single VM/virtual desktop has been consistently hogging the GPU resources, you may want to switch to the Dedicated GPU mode, so that excessive GPU</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			usage by that VM/virtual desktop has no impact on the performance of other VMs/virtual desktops on that host. If all GPUs assigned to a VM/virtual desktop are found to be busy most of the time, you may want to consider allocating more GPU resources to that VM/virtual desktop.

As stated earlier, by default, clicking on the **Virtual Servers** layer of a managed *VMware ESX* server, leads you to a page displaying the current status of the virtual guests executing on that server. If you want to override this default setting - i.e., if you prefer to view the tests mapped to the **Virtual Servers** layer first, and then proceed to focus on individual guest performance, follow the steps given below:

- Edit the **eg_ui.ini** file in the **<EG_INSTALL_DIR>\manager\config** directory
- Set the **LAYERMODEL_LINK_TO_VIRTUAL** flag in the file to **false**; this is set to **true** by default.
- Save the **eg_ui.ini** file.

Doing so ensures that as soon as the **Virtual Servers** layer is clicked, the list of tests mapped to that layer appears, as depicted by Figure 3.83.

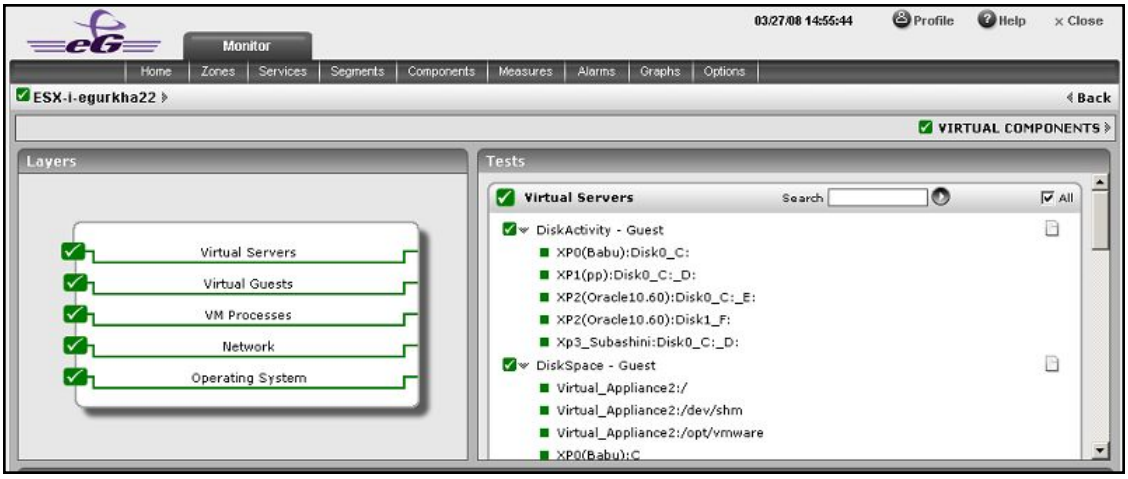


Figure 3.83: The tests mapped to the Virtual Servers layer of a VMware vSphere ESX server

If you now want the **Server view** of Figure 3.70, simply click on **SERVERS** link above the list of tests in Figure 3.83.

Clicking on any of the guests in the **Server view** leads you to Figure 3.84 that displays all the performance metrics extracted from that guest, in real-time. You are thus enabled to cross-correlate across the various metrics, and quickly detect the root-cause of current/probable disturbances to the internal health of a guest. To view the time-of-day variations in a measure, you can view its graph by clicking on that measure in Figure 3.84.

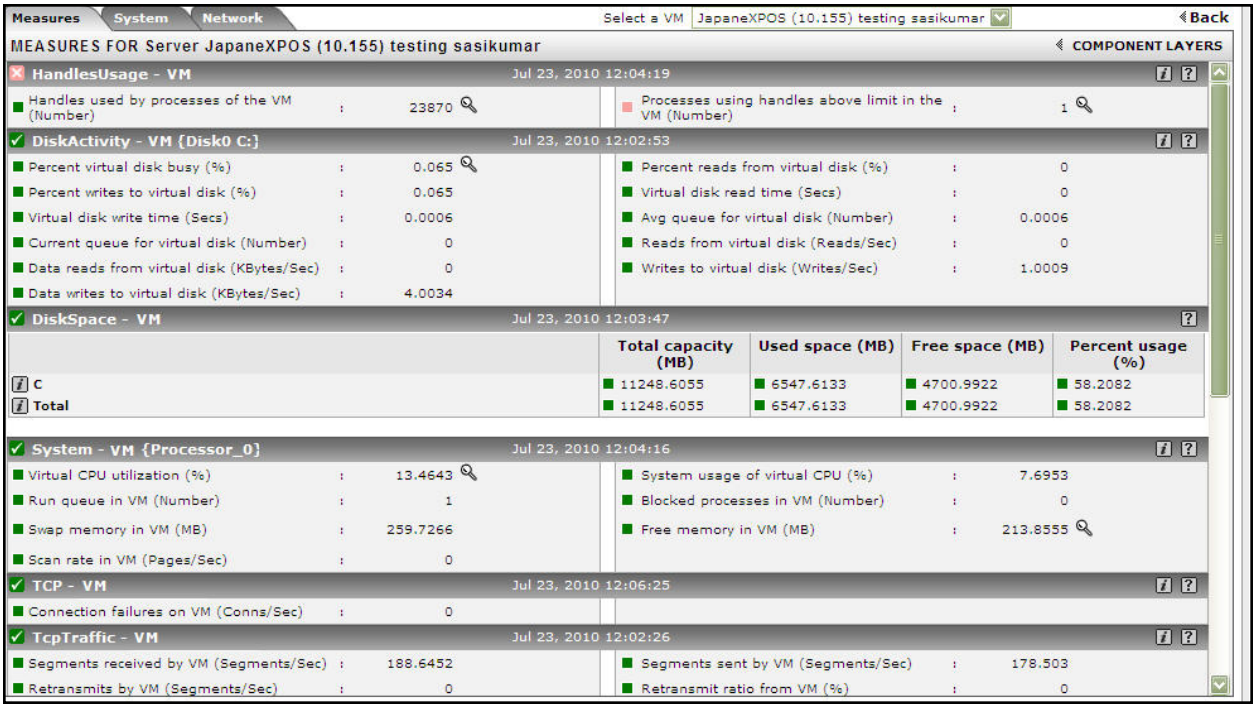


Figure 3.84: Measures pertaining to a chosen guest on a VMware vSphere ESX server

To view real-time graphs of pre-configured measures (pertaining to the ESX host and the guests operating on it), click on the **LIVE GRAPH** link in Figure 3.70. Alternatively, you can click on the icon that appears in the **Tests** panel (see Figure 3.49) when the **Outside View of VMs** layer is clicked. The graph display that appears subsequently (see Figure 3.85) has been organized in such a way that next to every host-pertinent measure graph, the closely related guest-specific measure graph appears. For instance, next to the graph of the 'Physical CPU usage' measure of the *Cpu - Esx* test, you will find a graph of the 'Physical CPU used' measure of the *EsxGuestDetails* test. This way, you can easily compare and correlate how well the physical CPU resources are being utilized by both the ESX host and the guests. On the basis of this analysis, you can proactively isolate potential performance issues, and also determine the root-cause of the issue - is it the ESX host? or is it the virtual guest? If you access this page from the **LIVE GRAPH** link in Figure 3.70, then, by default, you will view live graphs pertaining to the *VMware vSphere ESX* server. However, you can select a different virtualized component-type and a different virtualized component using the **type** and **Component Name** lists (respectively) in Figure 3.85.

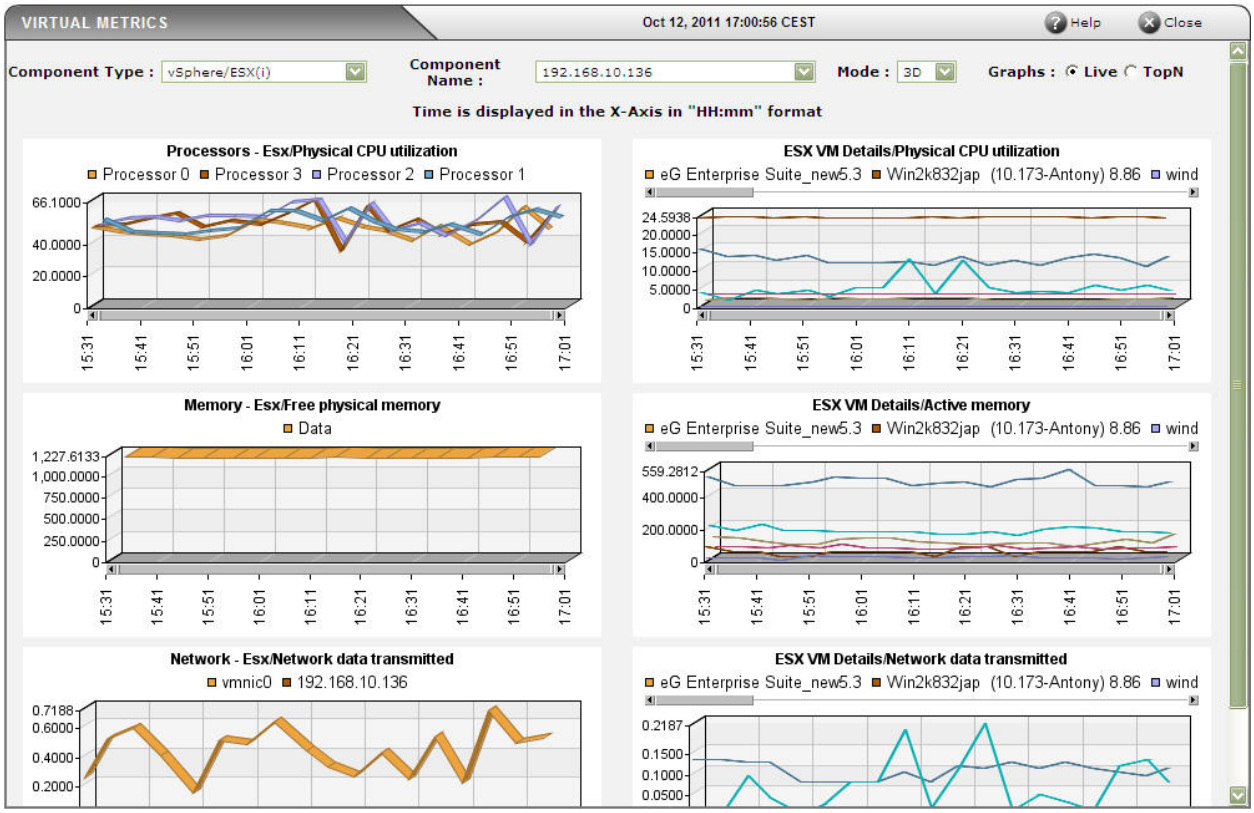


Figure 3.85: Live graph of the VMware vSphere ESX server

Also, using the eG Enterprise administration console, administrators can add applications running on the VM guests for monitoring. To monitor these applications, agents can be installed in the VM guests, or an agentless monitoring approach can be used. To effectively monitor the applications running in a virtual environment, it is important to be able to determine which ESX server an application is running. This mapping of applications to ESX servers is important for root-cause diagnosis - for example, a problem with the ESX server (e.g., excessive disk slowdowns) can impact the performance of all the applications running on the server's virtual machines. eG Enterprise is able to automatically determine the mapping of applications to ESX servers.

Whether eG Enterprise automatically determines the mapping of applications to ESX servers or not is determined by the value of the **AutoVirtualMapping** variable in the **[MISC]** section of the **eg_external.ini** configuration file in the **<EG_INSTALL_DIR>\manager\config** directory of the eG manager. If the value of this variable is **true,** the eG manager auto-discovers the applications to ESX servers mapping.

Note:

- For **AutoVirtualMapping** to work, the detailed diagnosis frequencies set globally (i.e., using the Configure -> Diagnosis menu sequence) should not be set to *0:0*.
- As long as the **Identify agents only using nick names** flag in the **MANAGER SETTINGS** page of the eG administrative interface (Configure -> Manager Settings menu sequence) is **Yes** (which is the default), eG Enterprise can automatically identify the server applications executing on an ESX host, using the host/nick names that are mapped to the IP addresses discovered on the host. If the **Identify agents only**

using nick names flag is set to **No** instead, then make sure that, while managing a server application executing in a virtualized environment, the hostname of the virtual machine is specified as the nick name of the corresponding server application. If more than one server application is executing on the same virtual machine, then any one of those server applications should have the virtual machine name as its nick name.

To disable auto-discovery, set this value to **false.** In such a case, once a *VMware vSphere ESX* or a *VMware ESX* server is added, then, when adding any new server application using the eG administrative interface, you will be prompted to manually set an association between the server application being added and the ESX server. In the example depicted by Figure 3.86, the **VIRTUAL ENVIRONMENT** flag is set to **YES** indicating that the Oracle server is running on a guest operating system. The name of the virtual host on which the component is hosted is indicated in the **VIRTUAL SERVERS** selection.

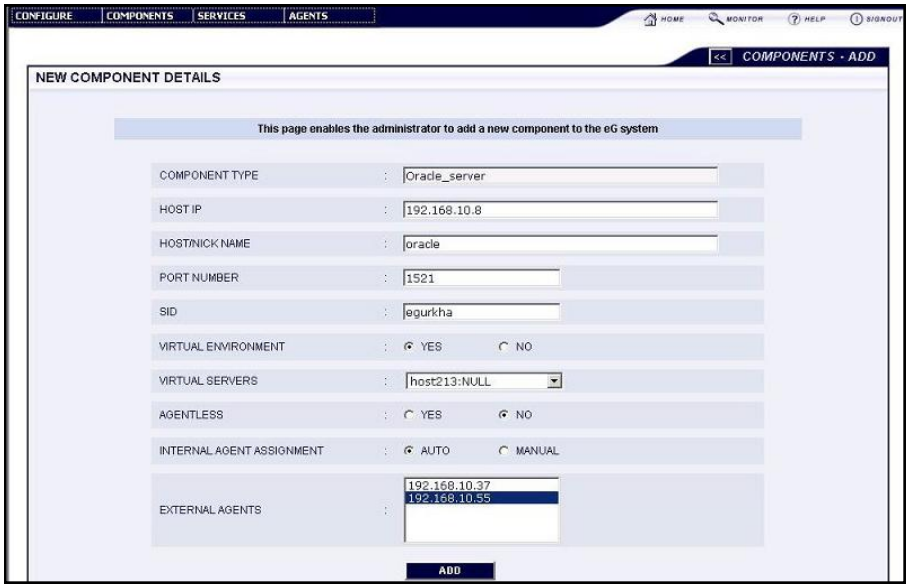


Figure 3.86: Adding a server application to a virtual environment

The mapping of applications to ESX servers is used by eG Enterprise for correlation - e.g., since the application runs on the ESX server, it is most likely that a problem with the ESX server will impact the performance of the application running on one of the guests. To view this application-ESX server association, simply click on the **VIRTUAL COMPONENTS** link in Figure 3.83.

Note:

The **VIRTUAL COMPONENTS** link will also be available in the layer model page of those server applications that are executing on virtual guests.

Doing so reveals Figure 3.87 depicting the *VMware vSphere ESX* sever and the server applications executing on it. By clicking on any of the components in Figure 3.87, the user can drill down into specific layers of this component for specific details on the performance of the component.

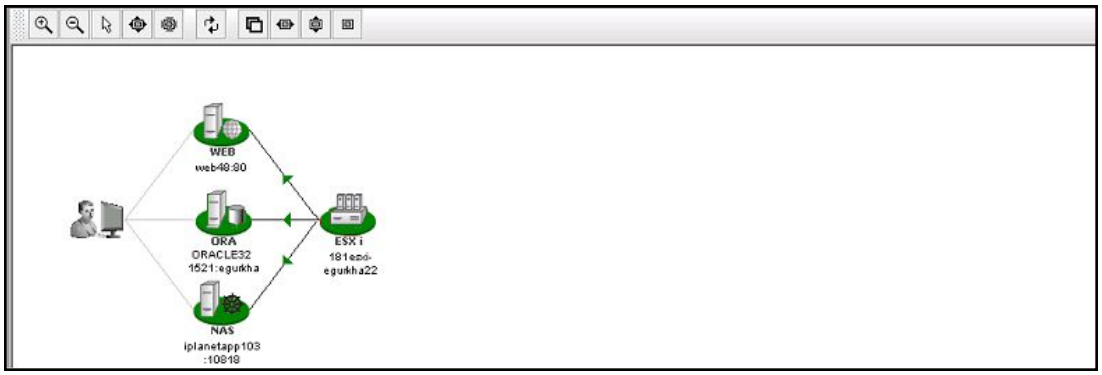


Figure 3.87: Depicts the applications that have been deployed on the guest OS of an ESX server

The arrows in Figure 3.87 depict the dependencies between the ESX host and the applications running on it. Since the applications are hosted on one of the guests running on the ESX host, they depend on the ESX host - i.e., any unusual resource usage on the ESX host impacts the applications running on any of the virtual guests. The dependency information between the ESX host and the applications hosted on it is used by eG Enterprise for end-to-end correlation.

3.7.15 Crash Details - VM Test

Event logs on Windows VMs capture critical error conditions such as service crashes and application crashes on the VMs, application and service hangs, and service errors. Since the crash/slowness experienced by any mission-critical program/service on a Windows VM may affect the uptime of the dependent business services, administrators should be able to instantly capture these serious problem conditions, investigate the reasons for their occurrence, and promptly resolve them. This is exactly what the Crash Details -VM test helps administrators achieve! This test periodically scans the event logs on each Windows VM and reports the count of crashes, hangs, and errors that may have occurred recently on that VM. Detailed diagnostics provided by this test pinpoints the applications/services that crashed, hanged, or encountered errors, and thus enables quick and efficient troubleshooting.

Note:

This test will not report metrics on VMs running Windows 2000/2003/XP.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every Windows VM on the vSphere/ESX server being monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target

ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

- **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5**.

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6**.

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to

the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP

address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1**.
    - **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1**.
    - If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to `none`.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified

using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

17. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Recent application crashes	Indicates the number of application crash events that occurred on this VM during the last measurement period.	Number	An event with the ID 1000 is logged in the event log every time a program terminates unexpectedly on a virtual desktop. This measure reports the number of events in the event log with event ID 1000. Use the detailed diagnosis of this measure to know which programs and

Measurement	Description	Measurement Unit	Interpretation
			modules stopped suddenly.
Recent service crashes	Indicates the number of service crash events that occurred on thisVM during the last measurement period.	Number	An event with the ID 7031 is logged in the Service Control Manager every time a service terminates ungracefully. This measure reports the number of events in the event log with event ID 7031. Use the detailed diagnosis of this measure to know the complete details of such events.
Recent application hangs	Indicates the number of application hang events that occurred on this VM during the last measurement period.	Number	An event with the ID 1002 is logged in the Application Event Log every time an application hangs. This measure reports the number of events in the event log with event ID 1002. Use the detailed diagnosis of this measure to know the complete details of the recent application hang events.
Recent service hangs	Indicates the number of service hang events that occurred on this VM during the last measurement period.	Number	An event with the ID 7022 is logged in the Service Control Manager every time a service hangs. This measure reports the number of events in the event log with event ID 7022. Use the detailed diagnosis of this measure to know the complete details of the recent service hang events.
Recent service errors	Indicates the number of service errors that occurred on this VM during the last measurement period.	Number	Events with the ID 7023, 7024, and 7026 are logged in the Service Control Manager every time a service error occurs. This measure reports the number of events in the event log with the aforesaid event IDs. Use the detailed diagnosis of this measure to know the complete details of the recent service errors.

3.7.16

3.8 The Inside View Dashboard

Though the **Measures** tab page of Figure 3.98 provides all the metrics collected from within a particular VM, these metrics are indicative of only the current health of the VM, and enable you to identify only the current issues related to the VM. To obtain a true picture of the health of a VM however, knowledge of the current status of the VM might alone not suffice; in addition to this, you would have to perform the following:

- Analyze the overall performance of a VM over time to deduce performance trends and past problems;
- Closely observe the changes in the current and historical usage of each of the virtual resources (CPU/memory/disk), so as to deduce usage patterns and potential resource contentions;
- Study the network traffic handled and bandwidth used by the VM over time, so that sudden/steady increases in traffic/bandwidth can be promptly detected;
- Determine the uptime of a VM by observing how long the VM has been up and running and how frequently it was rebooted during a specific time window;

To enable you to perform the aforesaid analysis, eG Enterprise offers dedicated **System** and **Network** dashboards for every guest operating system on an ESX server. The sections that will follow will help you understand each of these dashboards clearly.

3.8.1 The System Dashboard

Clicking on the **System** tab page in Figure 3.84 will take you to Figure 3.88, which reveals a **System Dashboard**.



Figure 3.88: The System Dashboard

The **System Dashboard**, using raw and graphically represented data, helps you understand how the critical resources of the guest operating system are being currently utilized, and how usage has changed over time. Besides enabling you to promptly detect over-utilization of resources, this dashboard also helps you accurately determine when such an unhealthy usage trend began and what caused it - is it owing to the improper allocation of virtual resources or the execution of one/more resource-intensive processes on the VM?

The contents of the dashboard typically vary based on the **Subsystem** chosen. By default, the **System Dashboard** provides an overview of VM performance. Accordingly, the **Overview** option is chosen from the **Subsystem** list by default. To zoom into the usage of a particular resource allocated to the VM, you can pick the corresponding option from the **Subsystem** list (i.e., **CPU/Disk/Memory**). On the other hand, if you want to analyze the uptime of the VM, choose **Uptime** as the **Subsystem**.

Each of these subsystems have been dealt with in detail in the sections that will follow.

1. **Overview**

To receive a macro view of the health of a VM, select **Overview** from the **Subsystem** list. Figure 3.89 then appears.



Figure 3.89: The System Overview Dashboard

The contents of the **Overview** dashboard are as follows:

1. The **Current System Alerts** section indicates the number and type of issues currently encountered by the guest operating system. To know more about these issues, move your mouse pointer over a specific cell in the **Distribution** section; this will reveal the details pertaining to the current alarms of that type, as depicted by Figure 3.90.

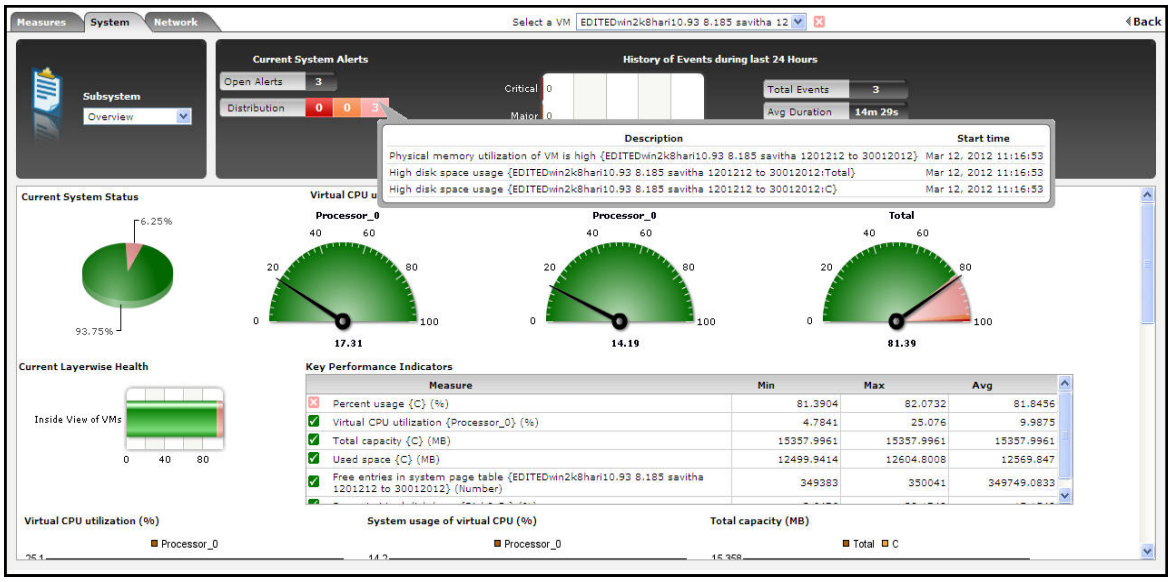


Figure 3.90: Viewing the current system alerts in the System Overview dashboard

2. Clicking on a particular problem's details in Figure 3.90 will lead you to the problem layer, test, and measure.

3. While the **Current System Alerts** focuses on current problems, the **History of Events** section provides a snapshot of problem situations in the past. By default, the bar chart in this section indicates the number and type (i.e., priority) of problems that affected VM operations during the last 24 hours. To know what these issues were, click on a particular bar in the bar chart. Figure 3.91 then appears providing elaborate details of the problems of that priority.

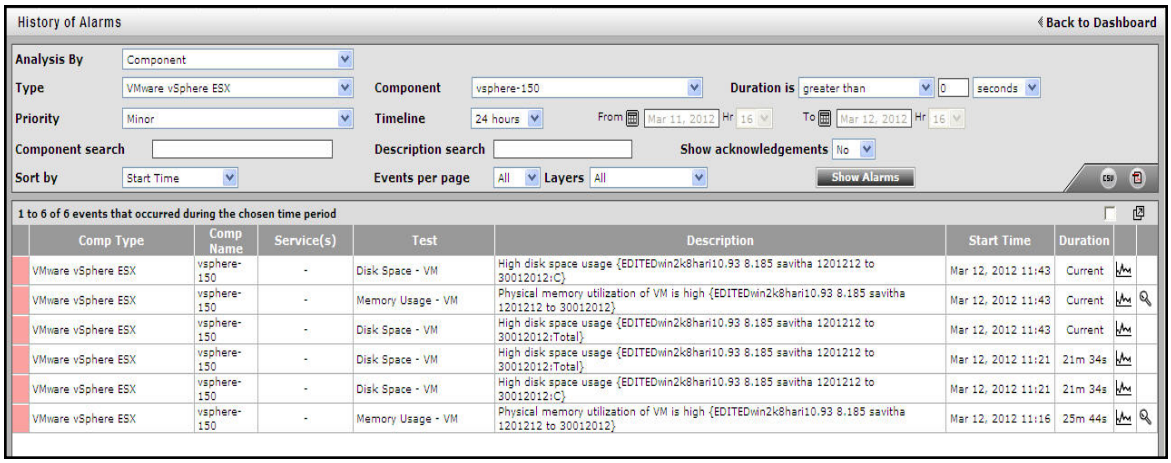


Figure 3.91: History of events of a particular priority

4. Click on the **Back to Dashboard** link at the right, top corner of Figure 3.91 to return to the dashboard.

5. Once back in the dashboard, shift your focus to **Current System Status** section within. The pie chart in this section helps you quickly determine how healthy the guest operating system currently is. The slices in this pie chart typically indicate the percentage of measures that are currently in various states

(normal/unknown/major/minor/critical). Clicking on a slice leads you yet again to the **EVENT HISTORY** page of Figure 3.91.

6. Next to the pie chart, you will find a series of dial charts indicating how well the VM is currently utilizing the CPU and disk space resources allocated to it. Abnormal resource usage by a VM can be instantly detected using these dial charts.

7. To know the current configuration of the VM, refer to the **System Configuration** section (not shown in Figure 3.90). This section will appear only if the eG license enables the **Configuration Management** module. If not, then a **Current Layerwise Health** section (shown in Figure 3.90) will appear instead. This section displays a bar chart that indicates the percentage of system-related measures mapped to the **Virtual Servers** layer that are in various states currently; this indicates how problem-prone the VM currently is.

8. To be promptly alerted to any change in the state of critical system-related parameters, refer to the **Key Performance Indicators** section. This section displays a pre-configured list of key resource usage metrics that govern the overall health of the VM. For every measure, the current value reported by that measure and the measure state will be available. Clicking on a measure name here will take you to Figure 3.92 that reveals the problem layer and test that reports the measure.

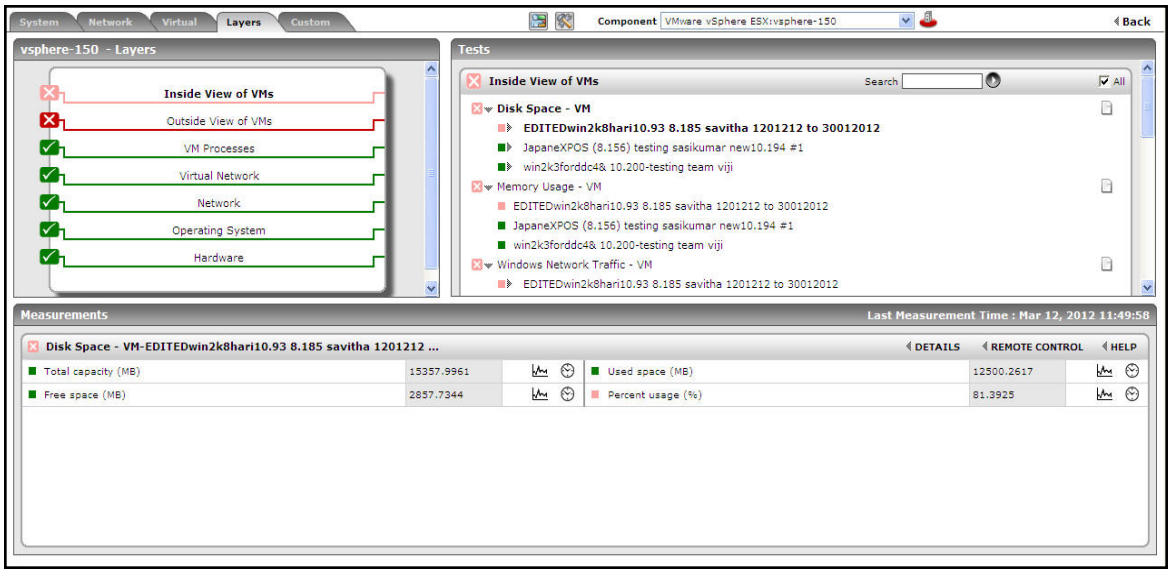


Figure 3.92: The layer model page that appears upon clicking on a key performance indicator

9. Below the **Key Performance Indicators** section, the dashboard provides a variety of measure graphs and top-n bar charts that facilitate effective analysis of both past and current performance. The graphs section begins with a series of measure graphs that reveal the time-of-day variations in the virtual resource usage during the last 1 hour (by default). In the event of excessive resource usage by the VM, these graphs will help you determine whether the increase in resource consumption happened suddenly or over time. This way, you can accurately figure out when a resource contention actually originated, and then investigate further as to what caused it. Some of these measure graphs may be plotted for multiple descriptors, and may hence appear cluttered. To view these graphs clearly and to analyze their performance implications better, click on the corresponding graph and enlarge it. In the enlarged mode (see Figure 3.93), you can even change the **Timeline** of the graph, so as understand

how performance has varied over a longer period of time. Similarly, you can even change the dimension of the enlarged graph from **3D** to **2D**, if need be.

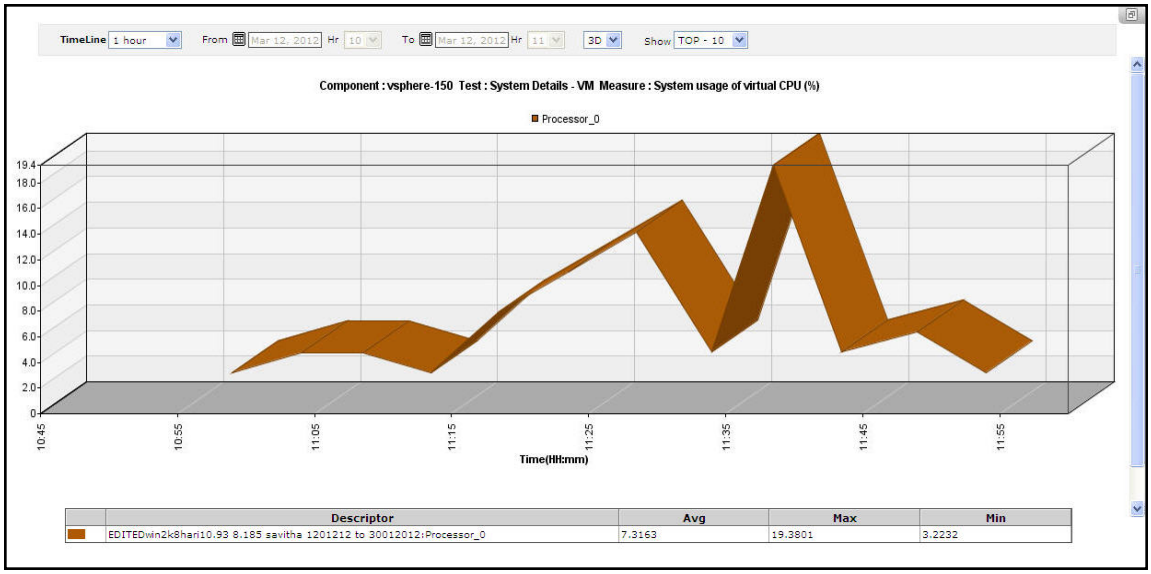


Figure 3.93: An enlarged measure graph in the System Overview Dashboard

10. After the measure graphs, top-n bar charts are available in the dashboard. Each of these bar charts reveal the most CPU-intensive, memory-intensive, and I/O-intensive processes that are currently executing on the VM, thus taking you a step closer to the root-cause of any resource contention that might have occurred currently on the VM. Sometimes, you might want to investigate issues with resource usage that might have occurred in the past, in an effort to zero-in on the process(es) that caused the issue. To achieve this, you need to alter the timeline of the bar charts. For this purpose, click on the required bar chart to enlarge it. Figure 3.94 will then appear. The enlarged bar chart allows you to change the **Timeline**, so that you can easily view the top resource-intensive processes that executed on the VM during the specified time period, and instantly detect root-cause.

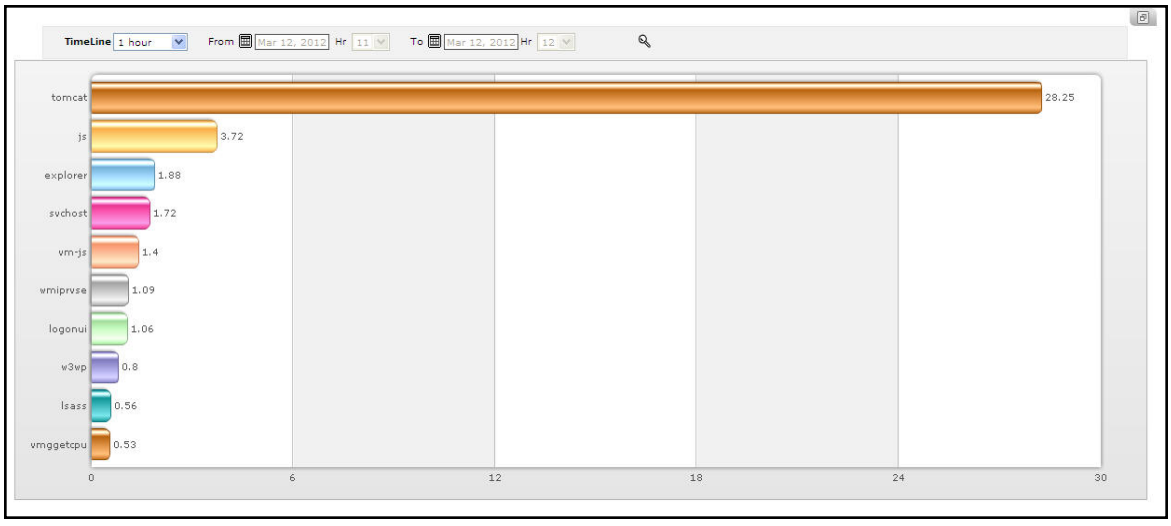
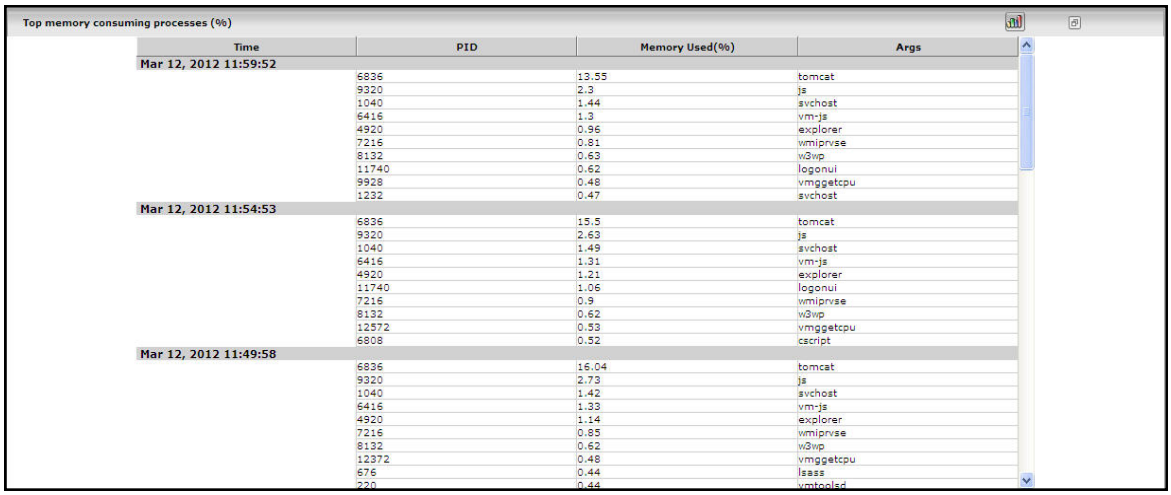


Figure 3.94: An enlarged top-n bar chart

11. You can even access the detailed diagnosis information that corresponds to your enlarged graph by clicking on the button at the right, top corner of Figure 3.94.
12. Figure 3.95 will then appear revealing the detailed diagnosis information.



Top memory consuming processes (%)

Time	PID	Memory Used(%)	Args
Mar 12, 2012 11:59:52			
	6836	13.55	tomcat
	9320	2.3	js
	1040	1.44	svchost
	6416	1.3	vm-js
	4920	0.96	explorer
	7216	0.81	wmiprvse
	8132	0.63	w3wp
	11740	0.62	logonui
	9928	0.48	vmggetcpu
	1232	0.47	svchost
Mar 12, 2012 11:54:53			
	6836	15.5	tomcat
	9320	2.63	js
	1040	1.49	svchost
	6416	1.31	vm-js
	4920	1.21	explorer
	11740	1.06	logonui
	7216	0.9	wmiprvse
	8132	0.62	w3wp
	12572	0.53	vmggetcpu
	6808	0.52	cscript
Mar 12, 2012 11:49:58			
	6836	16.04	tomcat
	9320	2.73	js
	1040	1.42	svchost
	6416	1.33	vm-js
	4920	1.14	explorer
	7216	0.85	wmiprvse
	8132	0.62	w3wp
	12372	0.48	vmggetcpu
	676	0.44	lsass
	220	0.44	vmtoolsd

Figure 3.95: Detailed diagnosis displaying the top memory-consuming processes executing within a VM

2. CPU

While the **Overview** subsystem provides you with quick insights into the overall resource usage situation on the VM, to know how the individual resources are utilized, you would have to select the corresponding option from the **Subsystem** list. Select **CPU** as the **Subsystem** if you want to focus only on the current and historical CPU usage of the VM, and promptly detect potential CPU bottlenecks. Upon selecting this option, Figure 3.96 will appear.



Figure 3.96: The Dashboard of the CPU Subsystem

1. The **CPU Usage Summary** section of Figure 3.96 lists the processors supported by your VM and the percentage of CPU utilized by each processor. In the event of excessive CPU usage by the VM, you can use this section to quickly identify the processor that is contributing to this problem condition.
2. Below this section, a collection of measure and top-n bar graphs are available that facilitate effective analysis of CPU usage over time. Using the measure graphs displayed here, you can figure out how CPU was utilized by the VM during the last 1 hour (by default). In the event of unusually high CPU consumption by the VM, these measure graphs will help you understand whether this increase in CPU usage is just a sudden, one-off incident, a phenomenon that occurred frequently during the last hour (by default), or an anomaly that aggravated over time.
3. To change the timeline of the measure graphs, so that usage analysis can be performed over longer time periods, click on the measure graph of interest to you. This will enlarge the measure graph. In the magnified mode, you can modify the **Timeline** of the graph and its dimension (3D/2D).
4. Besides the measure graphs, the **System Dashboard** of the **CPU Subsystem** displays a **Top CPU consuming processes** table, and a **CPU usage by top processes** graph. The **Top CPU consuming processes** table reveals the top CPU consumers on the VM during the last 1 hour (by default). Besides, the table also indicates how high and how low the CPU usage of each process has

scaled during the same hour. This way, you can understand whether the high CPU usage of a process was just a sudden spike or a consistent phenomenon.

5. The **CPU Usage by Top Processes** chart graphically compares the CPU usage of the processes that were executing on the VM during the last 1 hour (by default), and enables the quick and easy identification of the process that is the top CPU consumer. Clicking on this chart enlarges it enabling you to perform the comparative analysis more effectively.

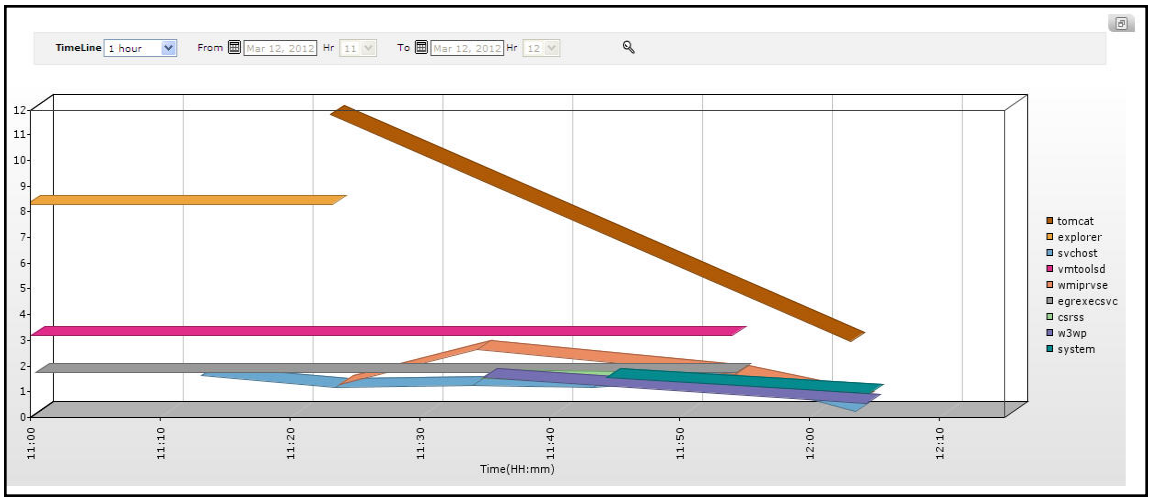


Figure 3.97: The enlarged CPU usage by top processes graph

6. You can even click on the icon in Figure 3.97 to view the detailed diagnosis page (see Figure 146), which reveals the top CPU-consuming processes that were executing on the VM during the **Timeline** specified in Figure 3.98.

CPU usage by top processes

Time	PID	%CPU	ARGS
Jul 23, 2010 17:25:14			
	4	1.56	system
Jul 23, 2010 17:15:19			
	4	2.72	system
	1720	1.7	vm-js
	1048	0.34	svchost
	596	0.34	csrss
	1136	0.34	egrexecsvc
	3076	0.34	csrss

Figure 3.98: Figure 3.98: The detailed diagnosis of the top processes by CPU usage

3. Memory

Be proactively alerted to current and probable memory contentions with the help of the **Memory Dashboard** that appears when the **Memory** option is chosen as the **Subsystem**.

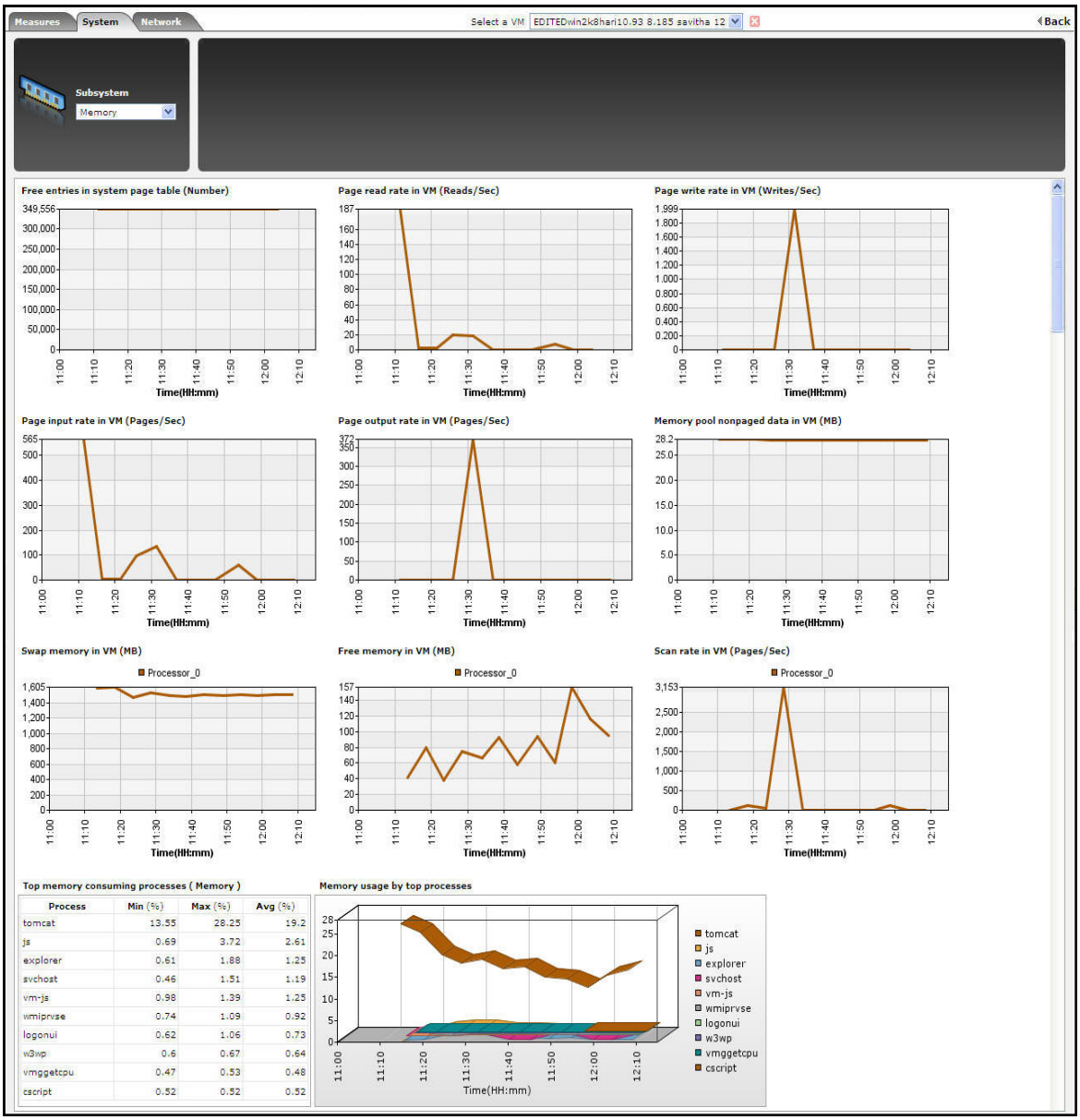


Figure 3.99: The Dashboard of the Memory Subsystem

The contents of this dashboard are as follows:

1. Using the **Memory Usage Summary** section, you can instantly figure out whether the VM in question is currently experiencing a memory crunch or not.
2. If a memory contention is noticed, then the graphs provided in the dashboard will enable you to figure out when the memory drain began, and what caused it. The graphs section typically begins with measure graphs which indicate how the VM has been using its allocated memory resources over the past 1 hour (by default). In the event of excessive memory usage by the VM, you can use these graphs to figure out whether the VM suddenly became memory-hungry, or has been consuming memory excessively for a while. If the memory usage has been consistently high, then you might want to go further back in time - i.e., beyond the default 1 hour period - to determine when exactly the memory

depletion began. For this, you will have to change the timeline of that measure graph that will aid your investigation. To alter the timeline of a graph, click on that graph to enlarge it, and then modify its **Timeline**.

3. On the other hand, if the measure graphs indicate that memory consumption increased only in the last hour, then you might want to know whether any memory-intensive processes were executing on the VM during the same period. To know which processes were executing and which one of them is the leading memory consumer, you can either use the **Top Memory Consuming Processes** table in the dashboard or the **Memory usage by top processes** graph. The table primarily reveals how much memory has been utilized, on an average, by each process on the VM during the last 1 hour (by default); in addition, it also displays the minimum and maximum percentage of memory utilized by every process in the same 1 hour period. Besides enabling you to accurately identify the most memory-intensive process on the VM, this table also helps you isolate those processes which have displayed erratic memory usage trends in the last 1 hour. The graph, on the other hand, aids you in visually comparing the memory usage of the processes on the VM during the last hour, and instantly nailing the process that is consuming the maximum memory resources.
4. If too many processes are executing on the VM, the graph is bound to appear cluttered. You can therefore, click on the graph to expand it and view the values plotted clearly.

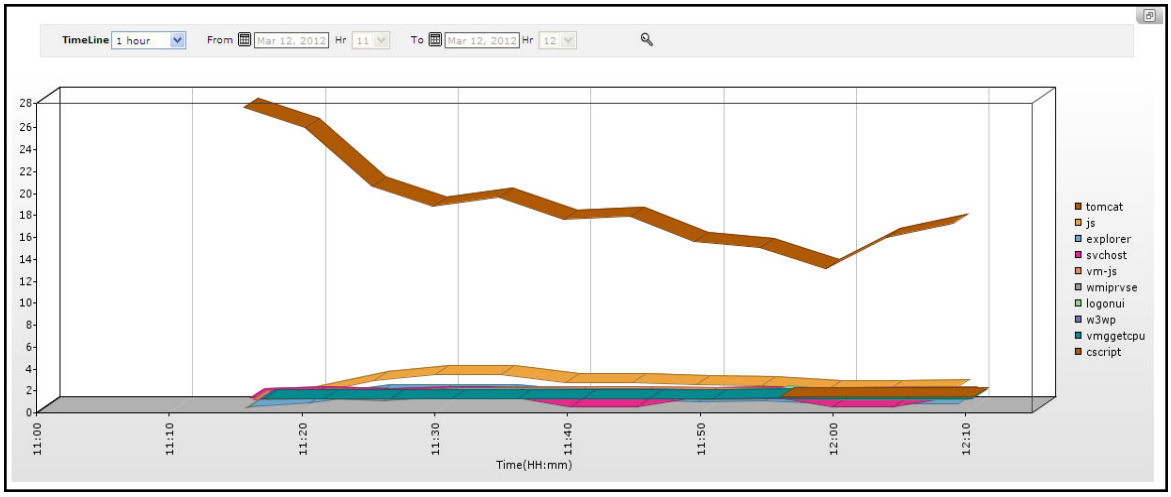
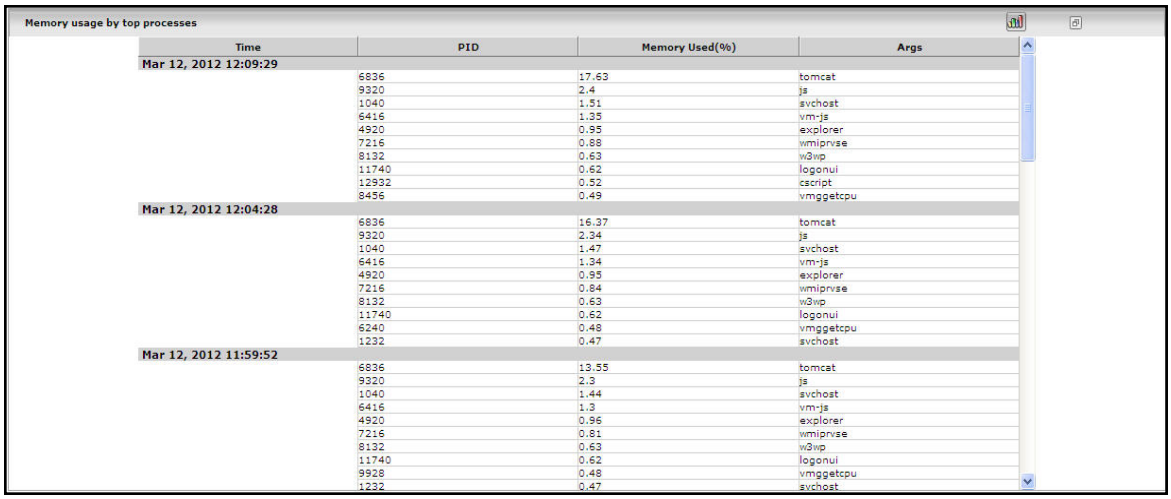


Figure 3.100: The enlarged memory usage graph

5. You can, if need be, alter the **Timeline** of the enlarged graph.
6. You can even click on the  icon in Figure 3.100 to view the detailed diagnosis page, which lists the Process ID (**PID**) of the top-10 memory consuming processes that were executing on the VM during the specified **Timeline**, and the memory used by each process.



Memory usage by top processes

Time	PID	Memory Used(%)	Args
Mar 12, 2012 12:09:29			
	6836	17.63	tomcat
	9320	2.4	js
	1040	1.51	svchost
	6416	1.35	vm-js
	4920	0.95	explorer
	7216	0.88	wmiprvse
	8132	0.63	w3wp
	11740	0.62	logonui
	12932	0.52	cscript
	8456	0.49	vmggetcpu
Mar 12, 2012 12:04:28			
	6836	16.37	tomcat
	9320	2.34	js
	1040	1.47	svchost
	6416	1.34	vm-js
	4920	0.95	explorer
	7216	0.84	wmiprvse
	8132	0.63	w3wp
	11740	0.62	logonui
	6240	0.48	vmggetcpu
	1232	0.47	svchost
Mar 12, 2012 11:59:52			
	6836	13.55	tomcat
	9320	2.3	js
	1040	1.44	svchost
	6416	1.3	vm-js
	4920	0.96	explorer
	7216	0.81	wmiprvse
	8132	0.63	w3wp
	11740	0.62	logonui
	9928	0.48	vmggetcpu
	1232	0.47	svchost

Figure 3.101: The detailed diagnosis displaying the top-10 memory-intensive processes on the VM

4. **Disk**

To closely observe disk usage and disk activity on a VM and to proactively identify partitions that are running out of space, select **Disk** from the **Subsystem** list. The **Disk Dashboard** of ‘ appears.



Figure 3.102: The Dashboard of the Disk Subsystem

The contents of this dashboard are as follows:

1. From the **Disk Usage Summary** section in Figure 3.102, you not only understand how many disk partitions are operating on the VM, but also receive real-time updates on the space usage per partition; this way, you can quickly isolate those partitions that are currently running out of space.
2. In addition to the usage summary, graphs on usage and disk activity are also available in the dashboard. The measure graphs in Figure 3.102 depict the time-of-day variations in disk usage and disk activity over a default period of 1 hour. Using these graphs, the following can be ascertained:
    - How has disk space been utilized across the virtual disks over the last 1 hour? Is any disk exhibiting excessive usage trends? If so, which disk partition is it, and when during the last hour did this trend begin?
    - How busy were the virtual disks during the last hour? Which disk was the busiest?
    - Are all virtual disks able to process requests quickly, or are too many I/O requests currently enqueued for a disk? If so, which disk is it?
    - Was any sudden/steady increase noticed in the length of the I/O request queue of a disk during the past hour?

3. However, analysis of an hour's data might not be enough at all times. For instance, the free space on a virtual disk might have decreased gradually, but steadily over a time span of say 1 week. This anomaly might not be evident in a 1-hour graph. You need to plot a graph for a week to figure this out. Therefore, to facilitate usage analysis over longer time windows, eG Enterprise allows you to change the timeline of the measure graphs. For this, simply click on the measure graph of interest to you to enlarge it. In the magnified mode, you not only get to alter the graph **Timeline**, but can also study the graph more clearly.

4. If the default measure graphs reveal sporadic spikes/a consistent rise in the disk activity during the last hour, then, it is imperative that you determine the reason for the same. One of the key reasons for an inexplicable rise in disk activity is the execution of one/more I/O-intensive processes on the VM during the default period (i.e., 1 hour). To know what these processes are and to identify the most I/O-intensive process, refer to the **Disk activity - Top processes** table or the **Disk busy by top processes** graph in Figure 3.102. The table lists those processes on the VM, which have, on an average generated high I/O activity on the virtual disks over the last 1 hour (by default). In addition, the table also indicates the minimum and maximum rate of disk activity generated by each of the top processes during the same 1 hour period. This not only helps you accurately identify the most I/O-intensive process on the VM, but also enables you to ascertain whether disk activity has varied dramatically or marginally during the last hour.

5. The **Disk busy by top processes** graph visually represents the **Avg** disk activity data available in the **Disk activity - Top processes** table, and thus aids you in quickly identifying that process which has generated the maximum disk I/O. To view this graph clearly, click on it to enlarge it (see Figure 3.103).

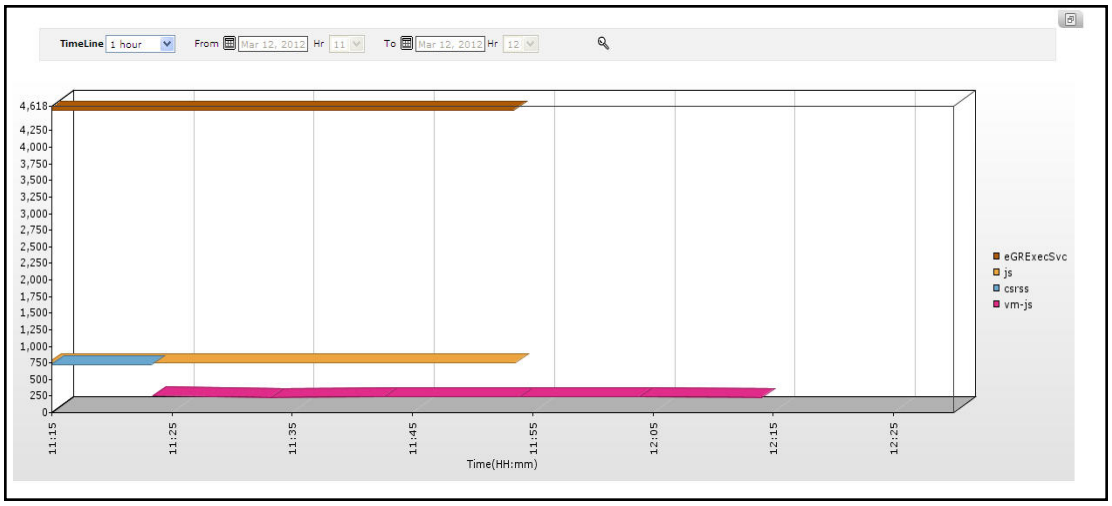


Figure 3.103: The enlarged Disk busy by top processes graph

6. You can change the **Timeline** of the enlarged graph. To view the detailed diagnosis information that has been plotted in the graph, click on the icon in Figure 3.103. Figure 3.104 then appears providing the detailed measures on disk activity for the time period chosen in the enlarged graph.

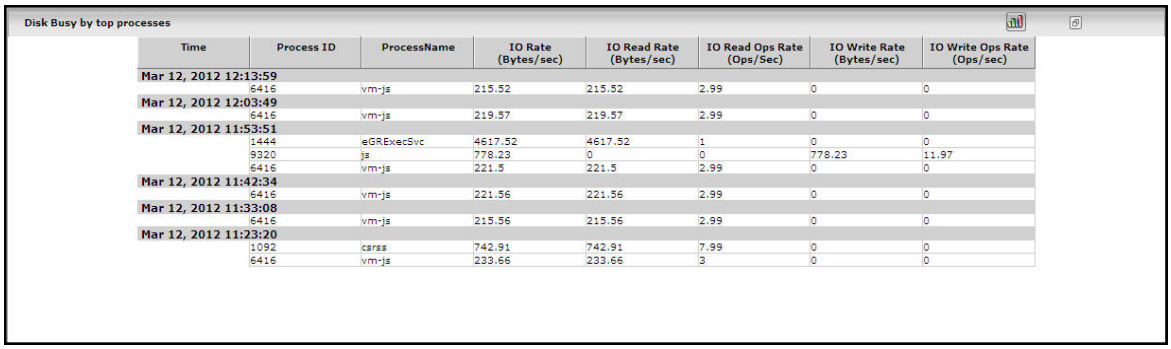


Figure 3.104: The detailed diagnosis displaying the top I/O intensive processes on the VM

5. Uptime

If **Uptime** is chosen as the **Subsystem**, the resulting dashboard enables you to promptly detect unscheduled reboots and determine how long the VM has been down.

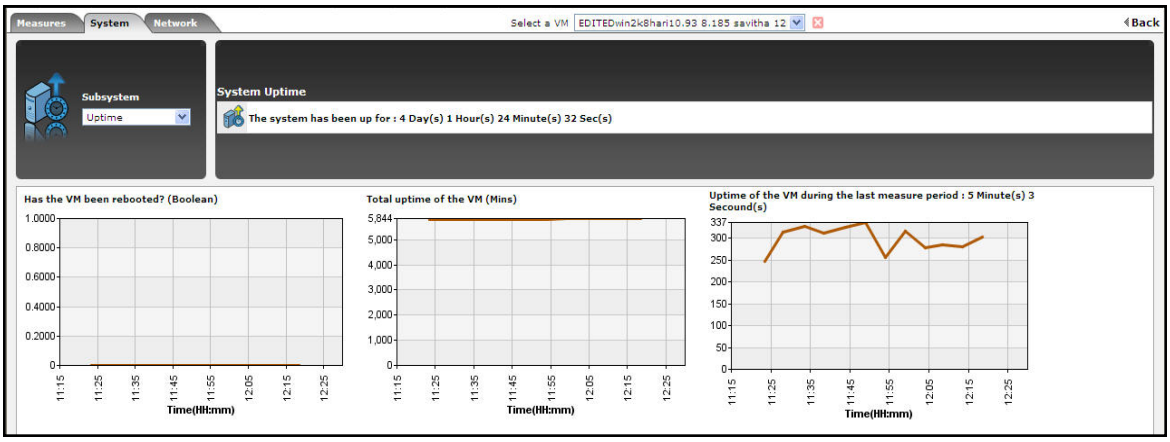


Figure 3.105: The Uptime Dashboard

The contents of this dashboard are as follows:

1. The **System Uptime** section indicates the total time for which the VM has been up since its last reboot.
2. The measure graphs that appear below indicate how long during the last hour (by default) the system has been up, whether the reboots scheduled for the system have occurred during the last hour or not, and how long during the last measurement period was the system up. Using these measure graphs, you can accurately determine reboot failures/prolonged VM downtime.
3. To alter the timeline of a graph, click on it to enlarge it, and then change the **Timeline** in the enlarged mode.

3.8.2 The Network Dashboard

The **Network Dashboard** provides you with an overview of the network and TCP traffic to and from a VM, and in the process reveals the following:

- Is network load balanced across all the interfaces supported by the VM, or is any interface currently experiencing excessive incoming/outgoing traffic?
- Which network interface is utilizing the maximum bandwidth and why?
- Are too many TCP connections currently alive on the VM?
- Were any connections dropped?
- Did any TCP connection attempt fail?
- Were there too many TCP retransmits?

Like the **System** dashboard, the contents of the **Network** dashboard too are primarily governed by the choice of **Subsystem**. By default, **Overview** is chosen as the network **Subsystem**. If need be, you can choose the **Network** or **Tcp Subsystems** instead.

The sections that will follow will focus on each of these **Subsystems**.

1. **Overview**

   In the **Overview** mode, the dashboard provides an all-round view of both the network and TCP health of the VM, enabling you to ascertain in a single glance:

   - The current status of the network and TCP connections to the VM;
   - Issues (if any) currently affecting network/TCP performance;
   - How network traffic, bandwidth usage, and overall TCP activity on the VM varied over the last 1 hour (by default), and what were the peak points/bottleneck areas



Figure 3.106: The Network Overview Dashboard

The contents of the **Overview** dashboard have been discussed hereunder:

1. The **Current Network Alerts** sections reveals the number and type of network-related issues currently affecting VM health. By moving your mouse pointer over a valid number against **Distribution**, you can view the details of the current network alarms pertaining to the corresponding severity (not shown in Figure 3.106). Clicking on any alarm information displayed in this section will lead you to the exact test and measure that reported the problem represented by the alarm.
2. Once back in the **Overview** dashboard, you can receive a fair idea of how problem-prone the VM has been in the past, by taking a look at the **History of Events** section (not shown in Figure 3.106). For every alarm priority, this section indicates the number of network/TCP-related problems that were experienced by the VM during the last 24 hours. Alongside this depiction, you will also find a table that reveals the average, maximum, and minimum duration for which a problem remained unresolved during the last 24 hours; from this information, you can infer how long your administrative staff took to turnaround, when faced with issues - this enables you to judge the efficiency of your staff better. For more details pertaining to the historical issues, click on a bar in the **History of Events** bar chart. Figure

3.107 will then appear providing the complete event history.

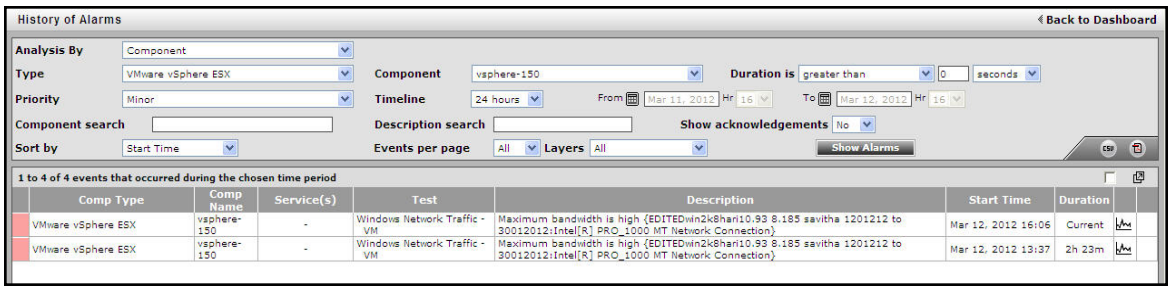


Figure 3.107: The History of network events

3. Below the **History of Events** section is the **Current Network Status** pie chart. This pie chart is a good indicator of the current network health of the VM, as it reveals the percentage of network-related measures that are currently in an abnormal state. Clicking on a slice in this pie chart will lead you to Figure 3.107 again.

4. Similarly, you have the **Current Layerwise Health** bar graph that indicates how healthy the network-related measures reported by the **Virtual Servers** layer of the VM currently are. If this bar chart and the **Current Network Status** pie chart reveal poor network health, then, you might want to know what is causing this - the dial charts and the **Key Performance Indicators** section reveal just that! By displaying the current state of critical network usage metrics, the dial charts instantly update you on current network health issues. By displaying the minimum, maximum, and average values of certain critical factors that influence network performance, the **Key Performance Indicators** section enables you to determine whether any of these factors could have contributed to the problem at hand. To know how any of these factors are currently performing, click on that particular key performance indicator listed in this section. Figure 3.108 will then appear.

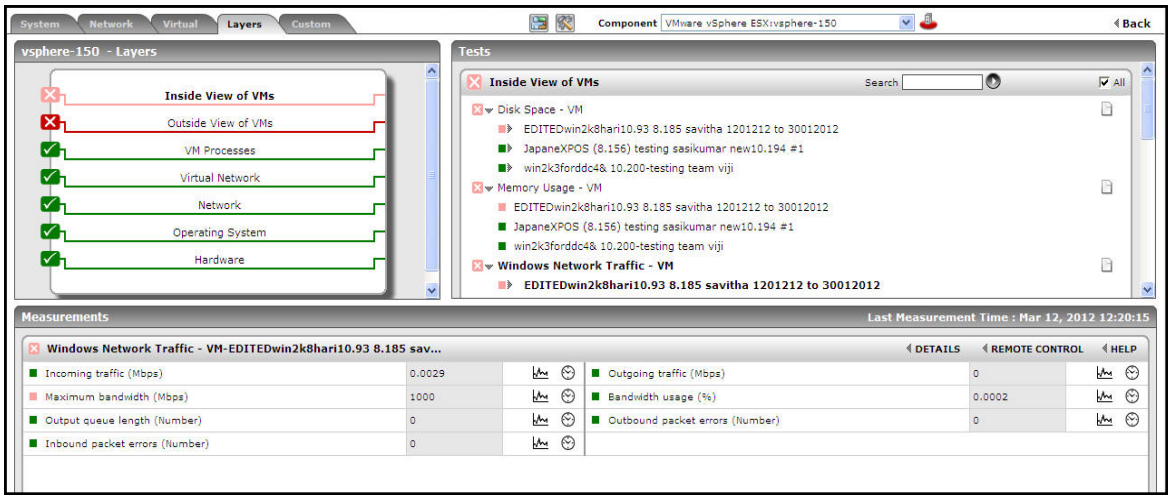


Figure 3.108: Clicking on a Key Performance Indicator

5. For historically analyzing network/TCP health and for detecting exactly when and why bottlenecks surfaced, use the measure graphs provided by the dashboard. These graphs, by default, track the variations in network performance over the last 1 hour. In the event of a network issue with the VM, you

can use these graphs to determine whether the issue occurred suddenly or is only the climax of a consistent deterioration in network performance.

6. Click on a graph to enlarge it. In the magnified mode, you can even change the **Timeline** of the graph, so as to obtain deeper insights into the past performance of the VM. In the case of graphs that plot values for multiple descriptors, you can pick a **TOP-N** or **LAST-N** option from the **Show** list and thus focus on the best/worst performers alone. You can even change the dimension of the graph from the default **3D** to **2D**.

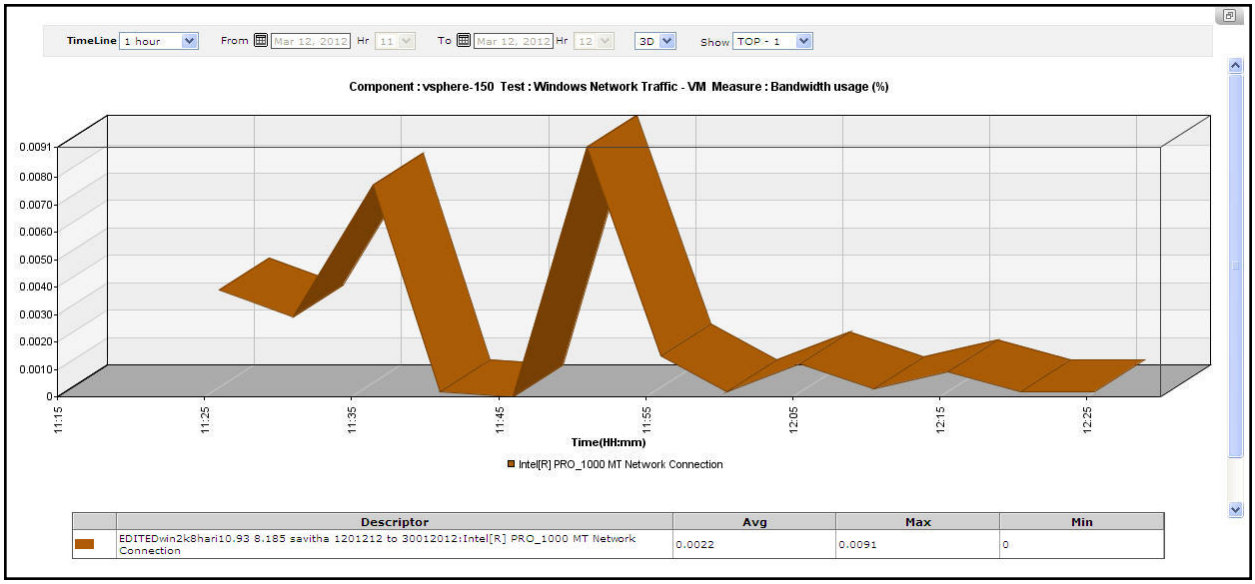


Figure 3.109: An enlarged network traffic graph

2. Network

To focus on network performance, select **Network** as the **Subsystem**. Figure 3.110 will then appear displaying time-of-day graphs that enable you to perform the following with ease:

- Track the variations in network traffic to and from the VM over the default period of 1 hour;
- Isolate network delays by observing the changes in the length of the output queue during the last 1 hour (by default);
- Detect when during the last hour (by default) packet errors occurred and why;

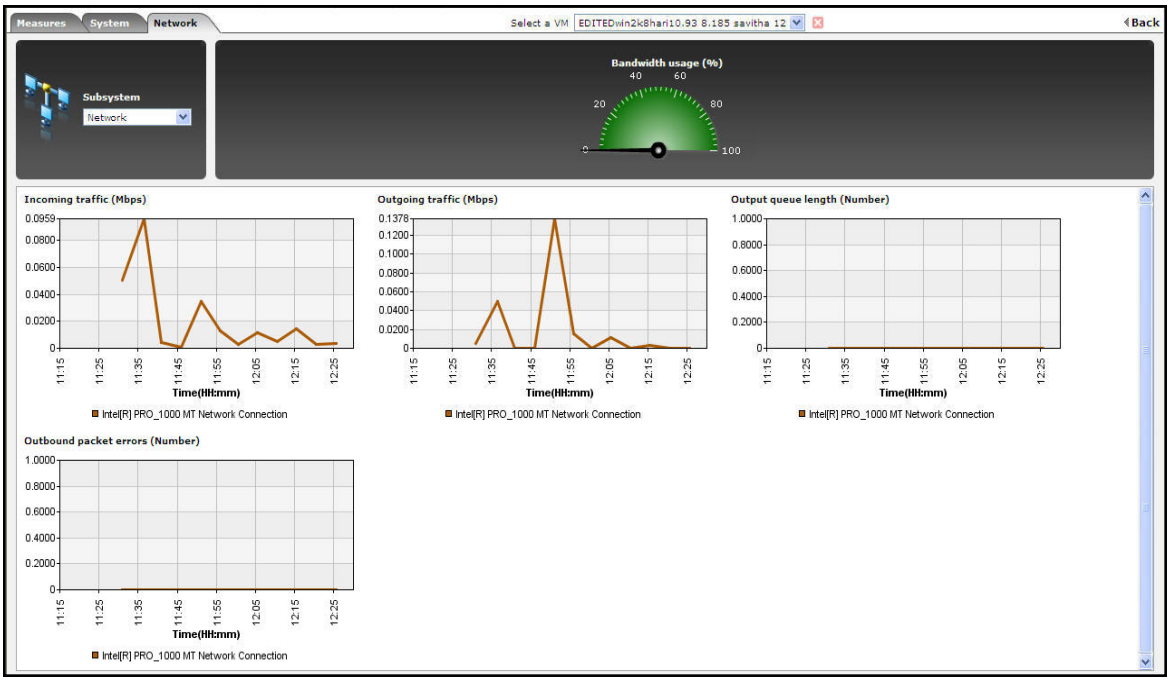


Figure 3.110: Dashboard of the Network Subsystem

You can click on any of the graphs here to enlarge it, and can even alter the **Timeline** of that graph in its magnified mode.

3. **Tcp**

Selecting **Tcp** as the **Subsystem** results in the display of time-of-day graphs on the Tcp connectivity of the chosen VM; these graphs, which are plotted for a default period of 1 hour, reveal the following:

- How has the Tcp load on the VM varied during the last 1 hour (by default)? Were there sporadic surges in load during the given period - if so, when did these spikes occur? If not, was a more consistent increase in load noticed during the last hour?
- Were too many Tcp connections suddenly dropped from the listen queue? Could it because of a Tcp overload on the VM?
- Did any Tcp connections fail during the default period? If so, when?
- Was the Tcp retransmit ratio optimal during the last 1 hour?



Figure 3.111: Dashboard of the Tcp Subsystem

You can click on any of the graphs here to enlarge it, and can even alter the **Timeline** of that graph in its magnified mode.

4

Monitoring vSphere/ESX Servers with VMs Hosting Desktop Applications

In some environments, the virtual guests hosted on vSphere/EXS servers may be used to support desktop applications. Administrators of such virtual environments would want to know the following:

- How many desktops are powered on simultaneously on the ESX Server?
- Which users are logged on and when did each user login?
- How much CPU, memory, disk and network resources is each desktop taking?
- What is the typical duration of a user session?
- Who has the peak usage times?
- What applications are running on each desktop?
- Which ESX Server is a virtual guest running on?
- When was a guest moved from an ESX Server? Which ESX Server was the guest moved to?
- Why was the guest migrated? What activities on the ESX host caused the migration?

Using the *VMware vSphere VDI* model (see Figure 4.1), administrators can find quick and accurate answers to all the queries above, and also receive a complete 'desktop view', which allows them to get up, close with the performance of every guest OS hosted by the ESX server and detect anomalies (if any) in its functioning.

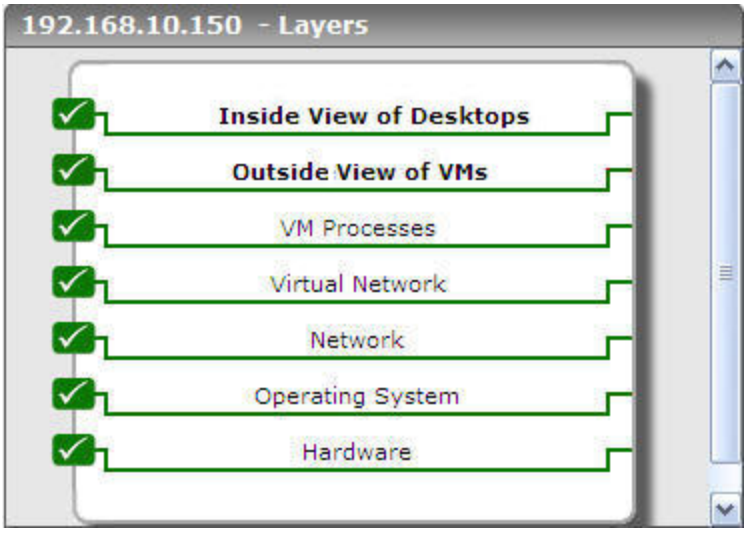


Figure 4.1: The layer model of a VMware vSphere VDI server

Each of the layers depicted by Figure 4.1 and the tests associated with the layers are discussed in detail in the sections that follow. Since the last 2 layers of the model have already been dealt with in the previous chapter, this chapter will discuss the **Virtual Machines** and the **Virtual Desktop** layer only.

4.1 The Outside View of VMs layer

The **Outside View of VMs** provides the host operating system's view of the resource usage levels of each of the virtual guests hosted on it. Using the information reported by this test, administrators can:

- Determine which of the guests is taking up more resources (CPU, memory, network, or disk) than the others. This information can help with load balancing or capacity planning. For example, if one of the guests is receiving a very high rate of requests compared to the others, this guest may be a candidate for migration to another ESX server, so as to minimize the impact it has on the other guests on the current ESX server.
- Determine times when sudden or steady spikes in the physical resource utilization are caused by the guest machines
- Know which guest systems at what times experienced heavy session loads or unexpected session logouts

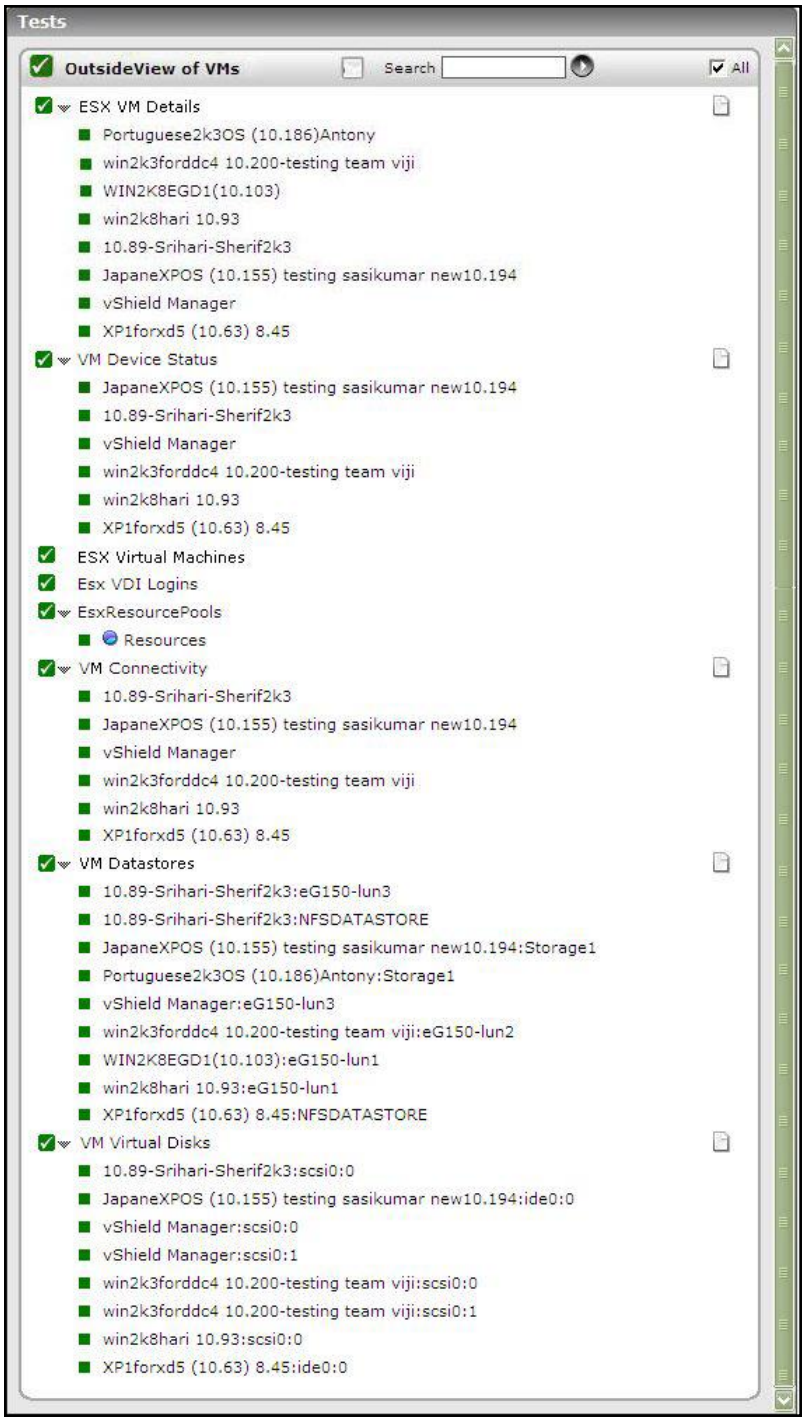


Figure 4.2: The tests associated with the Outside View of VMs layer of a VMware vSphere VDI server

4.1.1 VM Details – ESX Test

This test monitors the amount of the physical server's resources that each guest on an ESX server is taking up. Using the metrics reported by this test, administrators can determine which virtual guest is taking up most CPU, which guest is generating the most network traffic, which guest is taking up the maximum memory

utilization, which guest has the maximum disk activity, etc. Note that the amount of resources taken up by a virtual guest will be limited by the resource allocations that have been made by administrators. For example, an administrator could cap the amount of memory that a specific guest may take. Also, virtual guests can be organized into resource pools, and allocation of resources can be made at the resource-pool level. In this case, virtual guests allocated to the same resource pool contend for the resources allocated to the resource pool.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every logged-in user/powered guest on the ESX server monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has “Read-only” privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the “Read-only” role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to Section 2.6 of this document.
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

   Note:

   While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
10. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
11. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

    If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

    On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1**

    On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has

at least 'Read-only' privileges to the target ESX server.

12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private

key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_ virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if theREPORT BY USER FLAGis set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
16. **REPORT POWERED ON** - You can set the **REPORT POWERED ON** status to **YES**, so that the test reports an additional measure, *Is VM powered on?*, revealing whether a guest OS is currently running or not. The default status of this flag is set to **Yes** for a VMware ESX server. However, the default status of this flag is **No** in the case of virtual desktop infrastructures (VDI); this is because, in such environments, the virtual desktops will be in the powered-off state most of the time.
17. **AGGREGATE USER SESSIONS** -This flag is closely related to the **REPORT BY USER** flag. Since the **REPORT BY USER** flag is set to No by default for a VMware ESX server, this test will, by default, ignore the status of the **AGGREGATE USER SESSIONS** flag while monitoring that server. In case of the VDI model on the other hand, the **REPORT BY USER** flag is set to **Yes** by default. Therefore, the status of the **AGGREGATE USER SESSIONS** flag gains significance in the case of the VDI server. By default, the **AGGREGATE USER SESSIONS** flag is set to **No**. This implies that if a single user is currently logged into multiple guests, then this test, by default, will report a set of measures for every username on guestname. On the other hand, if the status of this flag is changed to **Yes**, then, this test will report a set of (aggregated) measures for every distinct user to the virtual desktop environment. In other words,

this test will report measures that are aggregated across all the currently active sessions for a user, spanning multiple VMs.

18. **ALERT FOR STATE CHANGE** - By default, the **ALERT FOR STATE CHANGE** flag is set to **False**, indicating that alerts will not be generated when the state of a VM changes from *Powered on* to *Powered off*. If this flag is set to **True**, then when the eG agents starts collecting metrics, alerts will be generated only for that particular VM whose state is changed from *Powered on* to *Powered off*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current sessions:	This measure is relevant only for monitoring of virtual desktops (i.e., for VMware VDI or VMware vSphere VDI servers). When reporting metrics for specific users, this metric indicates the number of sessions that each user has currently logged into; this measure will be available only if the test reports measures per currently logged in user.	Number	This is a good indicator of how busy the user is. The detailed diagnosis of this measure, if enabled, reveals the guests to which the user is currently logged on to.
VM power-on state:	Whether the virtual machine is currently running on the ESX server host or not.	Boolean	The table below displays the **States** that can be reported by this measure, and their numeric equivalents: **State** / **Value** Off / 0 On / 1 Suspended / 2 Unknown / 3 Standby / 4 Shuttingdown / 5 Resetting / 6 Stuck / 7 **Note:**

Measurement	Description	Measurement Unit	Interpretation
			By default, this measure reports one of the **States** listed in the table above. The graph of this measure however will represent the VM status using the numeric equivalents - '0' to '7'.
Physical CPU utilization:	Indicates the percentage of physical CPU used by the guest.	Percent	A high value for this measure indicates a virtual machine that is using a lot of the processor - possibly because one or more processes on this VM are taking a lot of CPU. By correlating the values of these metrics with the VM CPU ready metric will indicate if the VM is getting sufficient processor time or not. If the values of these metrics are high and VM CPU ready value is low, this is indicative of a VM that is getting enough processor time.
Physical CPU used:	Indicates the CPU used by this guest (in Mhz)	Mhz	The percent CPU usage measure serves as an effective indicator of how resource-intensive a particular VM is on a specific ESX server. However, for performing capacity planning or what-if analysis, the CPU usage of a VM measured in absolute terms would be more useful. For instance, the physical CPU usage of a VM could be 30% on a particular ESX server - this means that the VM is consuming 30% of the total physical CPU capacity of that ESX server. If you are planning to migrate the VM to another ESX server, then it would be unwise to assume that the VM will only consume 30% of CPU on the other ESX server as well, as the percentage will vary depending upon the physical CPU resources that are available to the other ESX server. The absolute measure however, will remain unchanged across

Measurement	Description	Measurement Unit	Interpretation
			ESX servers. Therefore, to decide which ESX server the VM is to be moved to, and to analyze the impact of this movement on the CPU resources of the new ESX host, you would require an absolute measure of CPU usage.
System usage of physical CPU:	Indicates the percentage of time this guest spent in the ESX server VMKernel to process interrupts and to perform other system activities.	Percent	An unusually high value indicates a problem and may be due to too many system- level tasks executing simultaneously.
VM CPU waits:	Indicates the percentage of time the guest spent in the wait state.	Percent	While the VM CPU ready metric denotes the time when the VM is waiting for CPU to execute, the busy wait state represents the time when the VM is waiting for some event to happen before it is ready to execute.
VM CPU extra:	Indicates the percentage of extra CPU used up by the virtual guests.	Percent	Ideally, the value of this measure should be very low. A high value indicates that one/more processes on the VM are consuming more CPU than necessary. Typically, screen savers, animations, and X servers all consume extra CPU, which can affect performance of other virtual machines and consolidation ratios. The non-trivial consumption of CPU by idling virtual machines can also have an adverse impact on Dynamic Resource Scheduling (DRS) decisions. You might want to disable screen savers and Window animations in the virtual machine to avoid the erosion of CPU resources. On Linux, if using an X server is not required, disable it.

Measurement	Description	Measurement Unit	Interpretation
VM CPU ready:	Indicates the percentage of time the guest was ready to run (i.e., it had instructions to execute) but was not able to because of processor contention.	Percentage	This metric should typically be low - generally 5% or less. The more time a VM spends waiting to run, the more lag time there is in responsiveness within the VM.
CPU guaranteed to VM:	Indicates the percentage of CPU guaranteed on the guest.	Percent	In some situations, system administrators want to know that a certain amount of memory for a virtual machine comes directly from the physical resources of the ESX Server machine. Similarly, the administrator might want to guarantee that a certain virtual machine always receives a higher percentage of the physical resources than other virtual machines. To ensure this, you can reserve physical resources of the host for specific virtual machines. Reservation specifies the guaranteed reservation for a virtual machine. The server allows you to power on a virtual machine only if the CPU and memory reservation is available. The server guarantees that amount even when the physical server is heavily loaded.
Physical CPU throttled:	Indicates the percentage of time the ESX host deliberately did not run the VM because that would violate the VM's limit setting.	Percent	A Limit refers to the maximum CPU the host can make available to this VM. Even though the VM is ready to run, if it is prevented from running owing to a probable limit violation, then the value of this measure will not be included in the value of the VM CPU ready measure. A high value for this metric indicates that the VM has been trying to get additional CPU resources but has not been able to

Measurement	Description	Measurement Unit	Interpretation
			do so because the CPU usage for this VM or its resource pool (if it is a part of a resource pool) has reached the allocated limit.
Memory overhead:	Indicates the memory overhead incurred by the virtual machine.	MB	ESX Server virtual machines can incur two kinds of memory overhead: • The additional time to access memory within a virtual machine. • The extra space needed by the ESX Server host for its own code and data structures, beyond the memory allocated to each virtual machine. The ESX server's memory virtualization ensures that the time overhead to memory accesses is minimal. The memory space overhead, on the other hand, is composed of two other components: • A fixed system-wide overhead for the service console (in the case of ESX 3/3.5) and the VMkernel. • Additional overhead for each virtual machine In addition, the space reserved for the virtual machine frame buffer and various virtualization data structures, also add to the memory overhead. Addition of more virtual CPUs, allocation of more memory to the guests, and choosing to configure 32- bit guest operating systems instead of 64-bit ones, can go a long way in reducing this overhead. The ESX server also provides optimizations such as memory sharing to save up memory.

Measurement	Description	Measurement Unit	Interpretation
Memory swapped out:	Indicates the rate at which memory is being swapped to disk by the ESX Server.	MB	ESX Server hosts use swapping to forcibly reclaim memory from a virtual machine when no vmmemctl driver is available because the vmmemctl driver: • Was never installed • Has been explicitly disabled • Is not running (for example, while the guest operating system is booting) • Is temporarily unable to reclaim memory quickly enough to satisfy current system demands Standard demand- paging techniques swap pages back in when the virtual machine needs them. Swap space must be reserved on disk for any unreserved virtual machine memory. This swap reservation is required to ensure the system is able to preserve virtual machine memory under any circumstances. In practice, only a small fraction of the swap space may actually be used. Typically, swap space usage for each VM should be low. Since access from RAM is much faster than access from physical disk, excessive usage of swap memory will slow down the performance of a VM. Watch for VMs that are seeing higher swap usage and more swap reads and writes.
Current swap memory:	Indicates the amount of memory that is currently swapped.	MB	
Memory swapped in:	Indicates the amount of memory that is being swapped in by the ESX Server from the disk for the guest.	MB	
Swap memory target:	Indicates the amount of memory that can be swapped.	MB	
Physical memory consumed:	Indicates the amount of physical memory consumed by the guest operating system.	MB	High memory consumption over long periods can deplete the free memory on the host, causing prolonged delays in the execution of the virtual machines. Comparison of the active memory usage across guests indicates the guest(s) that

Measurement	Description	Measurement Unit	Interpretation
			could be causing a memory bottleneck on the host.
Shared memory:	Indicates the current amount of shared guest operating system memory.	MB	
Balloon memory:	Indicates the total amount of physical memory currently reclaimed from this guest using the vmmemctl modules.	MB	The vmmectl driver that is installed on a virtual machine, emulates an increase or decrease in memory pressure on the guest operating system; this way, it forces the guest OS to place memory pages into its local swap file. This driver differs from the VMware swap file method as it forces the operating system to determine what memory it wishes to page. Once the memory is paged locally on the guest operating system, the free physical pages of memory may be reallocated to other guests. As the ESX hosts sees that memory demand has been reduced, it will instruct vmmemctl to “deflate” the balloon and reduce pressure on the guest OS to page memory. The maximum amount of memory that can be reclaimed from a guest may be configured by modifying the “ *sched.mem.maxmemctl* ” advanced option. If the memory reclaimed from a guest (i.e., the value of this measure) is very low, it indicates excessive memory usage by the guest. Under such circumstances, you might want to consider allocating more memory to the

Measurement	Description	Measurement Unit	Interpretation
			guest.
Balloon memory target:	Indicates the total amount of physical memory the ESX Server host attempts to reclaim using the vmmemctl module.		
Zero memory:	Indicates the amount of memory that is zeroed out.	MB	The “Memory Zero” amount will fluctuate as memory is over allocated. ESX will zero out the VM’s memory to use with other VM’s.
Memory granted:	Indicates the amount of memory that the Vmkernel has granted to the virtual machine.	MB	
Active memory:	Indicates the amount of memory that is actively used by the guest.	MB	
Memory usage:	Indicates the percentage of active memory that is currently in use.	Percent	High memory consumption over long periods can deplete the free memory on a VM, causing prolonged delays in the execution of the applications hosted by the virtual machine.
Configured memory:	Indicates the amount of configured memory.	MB	
Disk reads:	Indicates the rate at which read commands were issued to the storage adapters of the guest.	Reads/Sec	
Disk writes:	Indicates the rate at which write commands were issued to the storage	Writes/Sec	

Measurement	Description	Measurement Unit	Interpretation
	adapters of the guest.		
Issued disk commands:	Indicates the number of commands issued per second.	Commands/Sec	This measure is a good indicator of the level of activity on the storage adapters of the guest.
Disk commands aborted:	Indicates the rate at which commands aborted.	Aborts/Sec	A high rate is indicative of a problem condition that requires investigation.
Data writes to disk:	Indicates the rate at which data was written to the disk.	MB/Sec	
Data reads from disk:	Indicates the rate at which data was read from the disk.	MB/Sec	
Disk capacity:	Indicates the total disk capacity of the virtual machine.	MB	One of the key problems faced by VMware administrators is VM sprawl. Since VMs are easy to create and deploy, many a time an administrator might be faced with scenarios where many VMs are created on an ESX server, but very few are actively used. A VM, whether powered on or off, consumes disk space on an ESX server. When the ESX server hosting runs low on disk space, administrators might want to know which VM is taking up maximum disk space. This measure reveals the disk capacity of a VM, regardless of its on/off state. A quick comparison of the capacity across VMs can enable administrators to accurately identify the VM that is taking up maximum disk space.
Network packets transmitted:	Indicates the number of packets the guest	Packets/Sec	

Measurement	Description	Measurement Unit	Interpretation
	transmitted per second.		
Network data transmitted:	Indicates the amount of data the guest transmitted per second.	Mbps	Comparing the data transmitted across all the virtual guests provides an indicator of the guest that is generating most out-bound network traffic.
Network packets received:	Indicates the number of packets received per second by the guest.	Packets/Sec	
Network data received:	Indicates the rate at which data was received by the guest.	Mbps	Comparing the data received across all the virtual guests provides an indicator of the guest that has the most in-bound network traffic.
Virtual CPU utilization:	Indicates the percentage of allocated CPU resources that this VM is currently using.	Percent	Comparing the value of this measure across VMs will enable you to accurately identify the VMs on which CPU-intensive applications are executing.

4.1.2 Virtual Machines - ESX Test

This test enables administrators to determine how many guests have registered with the ESX server, and how many of these are currently running. In addition, the test also indicates whether any guests have migrated to or from the virtual server. VMware VMotion technology, unique to VMware, leverages the complete virtualization of servers, storage and networking to migrate an entire running virtual machine instantaneously from a failing/underperforming server to a healthy one, to ensure high availability of the virtual machine. The entire state of a virtual machine is encapsulated by a set of files stored on shared storage, and the VMware VMFS® cluster file system allows both the source and the target ESX Server to access these virtual machine files concurrently. The active memory and precise execution state of a virtual machine can then be rapidly transmitted over a high speed network. Since the network is also virtualized by ESX Server, the virtual machine retains its network identity and connections, ensuring a seamless migration process.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the ESX server monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ISVDISERVER** - By default, this flag is set to **NO** indicating that the test is being configured to execute for a *VMware vSphere VDI* model and not a *VMware vSphere ESX* model. This distinction needs to be made, since this test reports 2 additional measures for a *VMware vSphere VDI* server - namely, *Guests with users* and *Guests without users*.
5. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

6. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
7. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
8. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
9. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

10. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this

case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

12. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
13. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing

spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

Note:

While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

14. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
15. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *VMware vSphere ESX* server, this is set to *1:1* by default. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. It is recommended that you do not change the default setting of this parameter. This is because, eG Enterprise can discover the IP addresses of the guest operating systems on an ESX host, only while generating the detailed measures for this test. The automatic discovery of the guest IPs, in turn, enables eG Enterprise to perform **AutoVirtualMapping**.
16. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Registered guests:	Indicates the total number of virtual machines that have been registered with the ESX server.	Number	
Guests powered on:	Indicates the number of guests that are currently powered on.	Number	To know which are the guests that are powered on, use the detailed diagnosis capability of this measure (if enabled).
Guests with users:	Indicates the number of	Number	To know which guests the users have

Measurement	Description	Measurement Unit	Interpretation
	powered on guests with users logged in.		logged into, use the detailed diagnosis capability of this measure (if enabled). Note that this measure will not be available for a VMware vSphere ESX server.
Guests without users:	Indicates the number of powered on guests without any users logged in.	Number	Note that this measure will not be available for a VMware vSphere ESX server.
Added guests:	Indicates the number of guests that were newly added to the ESX server during this measurement period.	Number	The detailed diagnosis of these measures, if enabled, lists the virtual machines that were migrated by VMotion to or from (as the case may be) the ESX server.
Removed guests:	Indicates the number of guests that were newly removed from the ESX server during this measurement period.	Number	
VM templates:	Indicates the number of template VMs currently on the ESX server.	Number	A template is a “golden” copy of a virtual machine (VM) organized by folders and managed with permissions. They are useful because they act as a protected version of a model VM which can be used to create new VMs. As a template is the original and perfect image of a particular VM, it cannot be powered on or run.
Orphaned VMs:	Indicates the number of VMs that are in the orphaned state on the VMware vSphere ESX server.	Number	An orphan virtual machine is one that exists in the vCenter database but is no longer present on the ESX Server host. A virtual machine also shows as orphaned if it exists on a different ESX Server host than the ESX Server host expected by vCenter.

Measurement	Description	Measurement Unit	Interpretation
			A virtual machine can become orphaned, in any of the following situations: • After a VMotion or VMware DRS Migration; • After a VMware HA host failure occurs or after the ESX Server host comes out of maintenance mode; • If you delete a virtual machine outside of vCenter - say, through the VMware Management Interface while VC is down, or through the Virtual Infrastructure (VI) client directly connected to an ESX server host; • If vCenter is restarted while a migration is in progress; this is a temporary situation though; • If you schedule too many virtual machines to be relocated at the same time;
Other VMs:	Indicates the number of VMs which are in disconnected or invalid or inaccessible state on the vSphere/ESXi server.	Number	

The detailed diagnosis of the *Guests powered on* measure provides the details of the guests which are currently powered on. The details include the name of the guest, the IP address of the guest, and the operating system on which the guest has been mounted (see Figure 4.3).

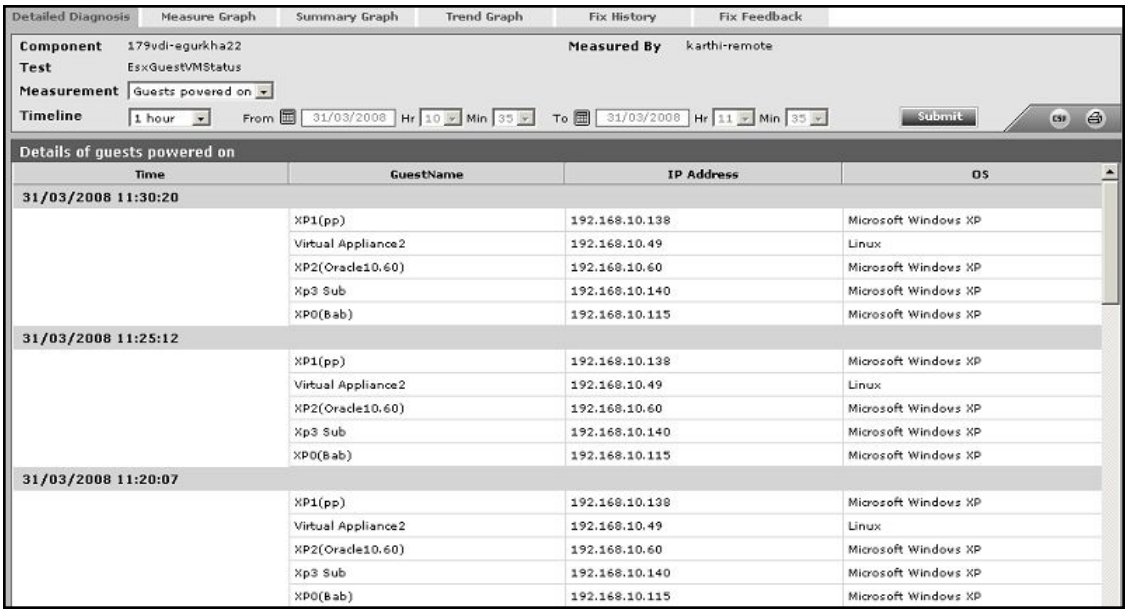


Figure 4.3: The detailed diagnosis of the Guests powered on measure

Similarly, the *Guests with users* measure provides detailed diagnosis information that reveals the guests to which users have logged in. Here again, the details include the name of the guest, the IP address of the guest, and the operating system on which the guest has been mounted.

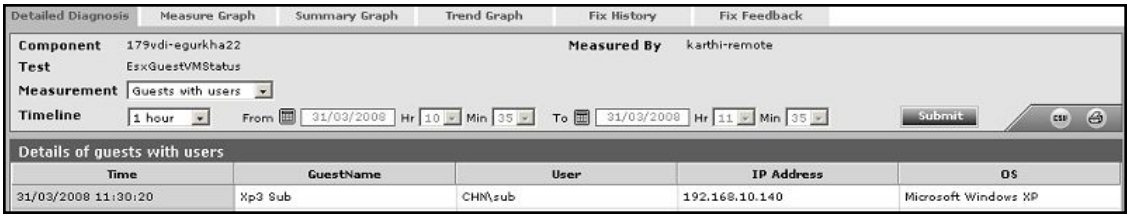


Figure 4.4: The detailed diagnosis of the Guests with users measure

4.1.3 VDI Logins – ESX Test

This test monitors the user logins to guests and reports the total count of logins and logouts.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the ESX server monitored.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.

4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .

   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** of this document to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

   Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this

was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the **SSL** flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section

3.1.1.1 .

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>**

(on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
12. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
13. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

14. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current sessions:	Indicates the number of user sessions that are currently active across all guests	Number	This is a good indicator of the session load on the guests.
New logins:	Indicates the number of new logins to the guests.	Number	A consistent zero value could indicate a connection issue. Use the detailed diagnosis of this measure to know which users logged in newly.
Percent new logins:	Indicates the percentage of current sessions that logged in during the last measurement period.	Percent	
Sessions logging out:	Indicates the number of sessions that logged out.	Number	If all the current sessions suddenly log out, it indicates a problem condition that requires investigation. The detailed diagnosis of this measure lists the sessions that logged out.

4.1.4 VDI Applications Test

This test discovers the applications executing on the virtual desktops and reports the availability and resource-usage of each of the desktop applications.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of outputs for every distinct application executing on the virtual desktops.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1**

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that

domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

- **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **4.1.4** .

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
12. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not

have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

Note:

While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

13. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to Yes (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their virtual machine name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **IS_SHOW_ALL_APPS** - To ensure that the test monitors only specific applications executing on the desktops and not all of them, set the **IS_SHOW_ALL_APPS** flag to **No**. Once this is done, then, you need to configure those applications that you want to exclude from the monitoring scope of this test. For this purpose, follow the steps given below:

    - Edit the **eg_tests.ini** file (in the {**EG_INSTALL_DIR}\manager\config** directory).
    - In the **[EXCLUDE_APPLICATIONS]** section of the file, you will find an entry of the following format:

      VmgApplicationTest={Comma-separated list of applications to be excluded}
    - To the comma-separated application list that pre-exists, append the applications that you want to monitor. For instance, if your test need not monitor *notepad.exe*, and *powerpnt.exe*, then, your entry should be:

      VmgApplicationTest=.....,notepad.exe,powerpnt.exe
    - Note that the exact application names should be provided, but the extensions (for instance,

.exe) can be dispensed with.

- Finally, save the file.

On the other hand, if you want to monitor all the applications, then, set the IS_SHOW_ALL_APPS flag to Yes.

17. **SHOW USER APPS ONLY** – By default, this flag is set to **Yes**. Accordingly, this test will monitor only those applications/processes that are running in the user's account. To monitor all applications/processes running in the virtual desktops, regardless of the user account using which they are running, set this flag to **No**.
18. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Processes running:	Indicates the number of instances of this application that is currently executing across all virtual desktops on the target host operating system.	Number	
CPU usage:	Indicates the percentage of physical CPU resources utilized by this application across the guest VMs.	Percent	A very high value of this measure is a cause for concern, as it indicates excessive CPU usage by a single application. This in turn would cause other desktop applications to contend for limited physical resources, thus degrading the performance of those applications and that of the virtual server as a whole.‘

Measurement	Description	Measurement Unit	Interpretation
Memory usage:	Indicates the percentage of physical memory resources utilized by this application across the guest VMs.	Percent	A very high value of this measure is a cause for concern, as it indicates excessive memory usage by a single application. This in turn would cause other desktop applications to contend for limited physical memory resources, thus degrading the performance of those applications and that of the virtual server as a whole.
CPU used:	Indicates the physical CPU (in Mhz) used up by this application.	Mhz	
I/O reads:	Indicates the rate at which this application reads data from the virtual disk.	KB/Sec	Compare the value of these measures across applications to identify the I/O-intensive application.
I/O writes:	Indicates the rate at which this application writes data to the virtual disk.	KB/Sec	
I/O read operations:	Indicates the rate at which this application performs read operations.	Operations/Sec	Compare the value of these measures across applications to identify the I/O-intensive application.
I/O write operations:	Indicates the rate at which this application performs write operations.	Operations/Sec	
Total I/O operations:	Indicates the rate at which this application performs I/O operations.	Operations/Sec	

Since the remaining tests mapped to the **Outside View of VMs** layer - namely, the **VM Connectivity** test and **VM Datastores** test - have been discussed already, let us move to the **Virtual Desktop** layer.

4.2 The Inside View of Desktops Layer

The **Outside View of VMs** layer provides an "external" view of the different VM guests - the metrics reported at this layer are based on what the VMware host is seeing about the performance of the individual guests. However, an external view of the VM guest operating system and its applications may not be sufficient. For instance, suppose one of the disk partitions of the guest operating system has reached capacity. This information cannot be gleaned from host operating system. Likewise, bottlenecks such as a longer process run queue or a higher disk queue length are more visible using an internal monitor. Internal monitoring (from within the guest operating system) also provides details about the resource utilization of different application (s) or processes.

The tests mapped to the **Virtual Desktop** layer provide an "internal" view of the workings of each of the guests - these tests send probes into each of the guest operating systems to analyze how well each guest utilizes the resources that are allocated to it, and how well it handles user sessions, TCP traffic, and network loading.

By default however, clicking on the **Virtual Desktop** layer, does not list the associated tests. Instead, Figure 4.5 appears, which displays the current state of all virtual desktops that have been configured on the monitored ESX host.

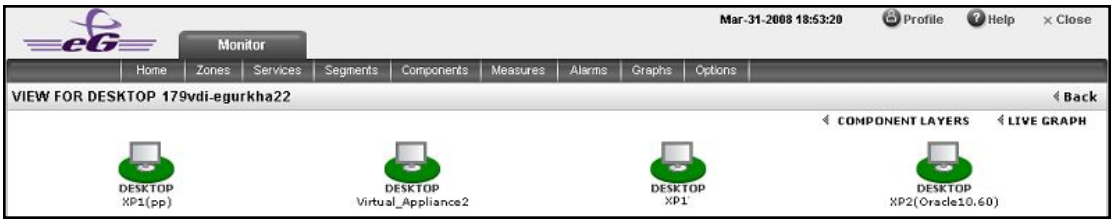


Figure 4.5: The current state of the desktops configured on the ESX VDI (i) server host that is monitored

To return to the layer model of the *VMware VDI* server and view the tests mapped to the **Virtual Desktop** layer, click on the **COMPONENT LAYERS** link in Figure 4.5. The tests depicted by Figure 4.6 then appears.

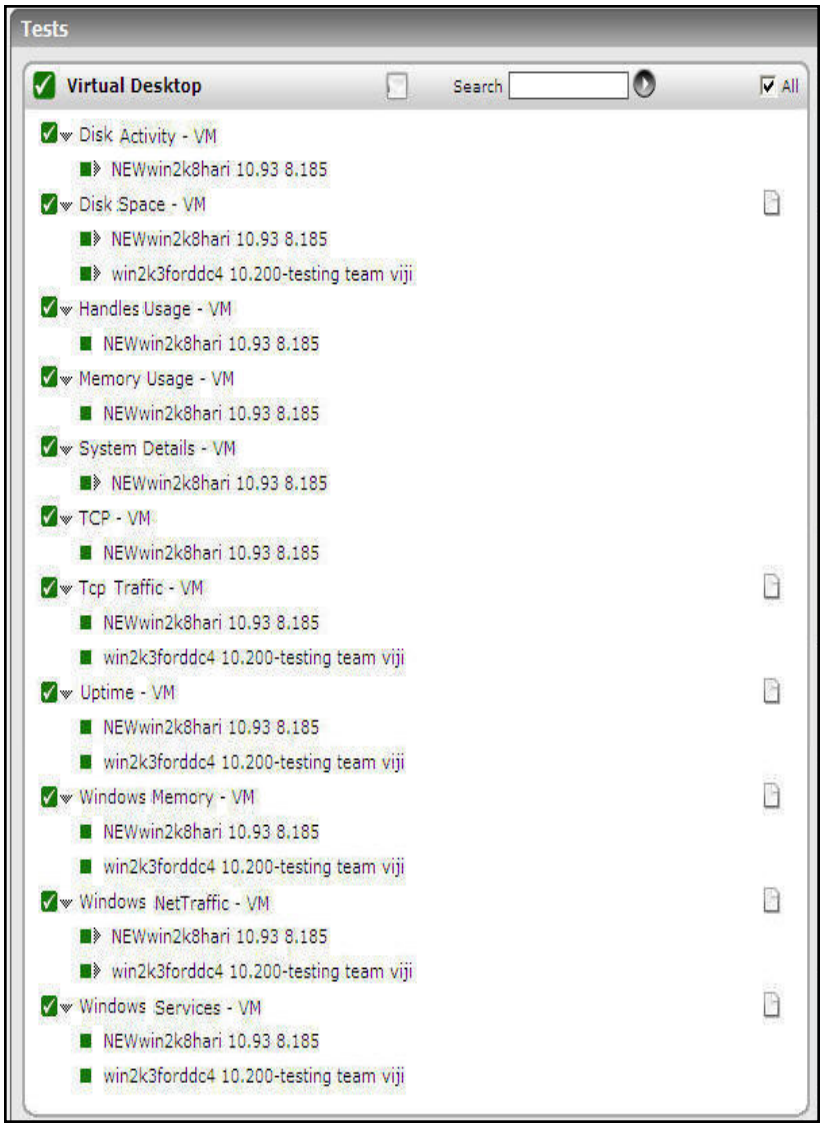


Figure 4.6: The tests associated with the Virtual Desktop layer of a VMware vSphere VDI

Almost all the tests depicted by Figure 4.6 have already been dealt with in the section on the *vSphere ESX server* model. The two tests that are specific to the *vSphere VDI Server* model are the Terminal to Desktop Connection test and the Desktop ICA Channel Test.

4.2.1 Desktop's HDX Channel Test

The key factors influencing user experience in a virtual desktop infrastructure are the latencies experienced by the user while connecting to the desktop via ICA and the bandwidth used when a user interacts with a virtual desktop. High latency and excessive bandwidth consumption can often slowdown access to desktops, thereby significantly delaying subsequent user operations. Hence, monitoring the latency and bandwidth usage of the ICA communication channel between the user terminal and the virtual desktops is essential.

The Desktop's HDX Channel test auto-discovers the virtual desktops on the ESX host and the users who are currently connected to each desktop. For each such user, the test monitors the communication between a user and the virtual desktop, and reports the following:

- The latency experienced by each user session;
- The bandwidth used by the incoming and outgoing data/audio/multimedia traffic transacted by the ICA communication channel between each user and virtual desktop;

Using this test, an administrator can identify user sessions that are being impacted by high latency and abnormal bandwidth usage. In addition, the test also reveals the type of traffic that is causing excessive bandwidth usage, thereby providing pointers to how the client configuration can be fine-tuned in order to reduce bandwidth consumption and improve performance.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence: Agents -> Tests -> Enable/Disable, pick *VMware vSphere ESX* as the **Component type**, set *Performance* as the **Test type**, choose this test from the disabled tests list, and click on the **>>** button to move the test to the **ENABLED TESTS** list.

Note:

This test will report metrics only if the following conditions are fulfilled:

- The test is applicable to **Windows VMs** only.
- The VMs being monitored should be managed by XenDesktop Broker.
- The Virtual Desktop Agent software should have been installed on the VMs.
- The **ICA Session** performance object should be enabled on the VMs.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every user who is connected to a virtual desktop via ICA.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has

"Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this

case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.
9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test

should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **4.2.1**.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
12. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
13. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

If the **REPORT POWERED OS** flag is set to Yes (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their virtual machine name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Session average latency:	Indicates the average client latency over the lifetime of this session.	Secs	Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when connecting to a virtual desktop.
Session deviation latency:	Indicates the difference between the minimum and maximum measured latency values for this session.	Secs	
Audio bandwidth input:	Indicates the bandwidth used while transmitting sound/audio to this user.	Kbps	Comparing these values across users will reveal which user is sending/receiving bandwidth- intensive sound/audio files over the ICA channel. To minimize bandwidth consumption, you may want to consider disabling client audio mapping.
Audio bandwidth output:	Indicates the bandwidth used while receiving sound/audio from this user.	Kbps	
COM bandwidth input:	Indicates the bandwidth used when sending data to this user's COM port.	Kbps	Comparing these values across users will reveal which user's COM port is sending/receiving bandwidth- intensive data over the ICA channel.
COM bandwidth ouput:	Indicates the bandwidth used when receiving data from this user's COM port.	Kbps	
Drive bandwidth	Indicates the bandwidth	Kbps	Comparing the values of these measures

Measurement	Description	Measurement Unit	Interpretation
input:	used when this user performs file operations on the mapped drive on the virtual desktop.		across users will reveal which user is performing bandwidth- intensive file operations over the ICA channel. If bandwidth consumption is too high, you may want to consider disabling client drive mapping on the client device. Client drive mapping allows users logged on to a virtual desktop from a client device to access their local drives transparently from the ICA session. Alternatively, you can conserve bandwidth by even refraining from accessing large files with client drive mapping over the ICA connection.
Drive bandwidth output:	Indicates the bandwidth used when the virtual desktop performs file operations on the client's drive.	Kbps	
Printer bandwidth input:	Indicates the bandwidth used when this user prints to a desktop printer over the ICA channel.	Kbps	Comparing the values of these measures across users will reveal which user is issuing bandwidth- intensive print commands over the ICA channel. If bandwidth consumption is too high, you may want to consider disabling printing. Alternatively, you can avoid printing large documents over the ICA connection.
Printer bandwidth output:	Indicates the bandwidth used when the desktop responds to print jobs issued by this user.	Kbps	
Session bandwidth input:	Indicates the bandwidth used from this user to the virtual desktop for a session	Kbps	Comparing the values of these measures across users will reveal which user and which virtual desktop is performing bandwidth- intensive operations for a session.
Session bandwidth output:	Indicates the bandwidth used from the virtual desktop to this user for a session.	Kbps	
Session compression input:	Indicates the compression ratio used from this user to the virtual desktop for a session.	Number	Compression reduces the size of the data that is transacted over the ICA channel.

Measurement	Description	Measurement Unit	Interpretation
			Comparing the values of these measures across users will reveal which client has been configured with a very low and a very high compression ratio. In the event of high bandwidth usage over an ICA channel, you can set a higher compression ratio for the corresponding client and thus reduce bandwidth consumption.
Session compression output:	Indicates the compression ratio used from the virtual desktop to this user for a session.	Number	
Speed screen data channel bandwidth input:	Indicates the bandwidth used from this user to the virtual desktop for data channel traffic.	Kbps	Comparing the values of these measures across users will reveal which user has been transmitting/receiving bandwidth-intensive data channel traffic.
Speed screen data channel bandwidth output:	Indicates the bandwidth used from virtual desktop to this user for data channel traffic.	Kbps	
Speed screen multimedia acceleration bandwidth input:	Indicates the bandwidth used from this user to virtual desktop for multimedia traffic.	Kbps	Comparing the values of these measures across users will reveal which user has been transmitting/receiving bandwidth-intensive multimedia traffic.
Speed screen multimedia acceleration bandwidth output:	Indicates the bandwidth used from the virtual desktop to this user for multimedia traffic	Kbps	
HDX media stream for flash data bandwidth input:	Indicates the bandwidth used from this user to virtual desktop for flash data traffic.	Kbps	Comparing the values of these measures across users will reveal which user has been transmitting/receiving bandwidth-intensive flash data.

Measurement	Description	Measurement Unit	Interpretation
HDX media stream for flash data bandwidth output:	Indicates the bandwidth used from the virtual desktop to this user for flash data traffic	Kbps	
USB bandwidth input:	Indicates the bandwidth used from this user to the virtual desktop for the USB port-related traffic.	Kbps	Comparing the values of these measures across users will reveal which user has been transmitting/receiving bandwidth-intensive USB traffic.
USB bandwidth output:	Indicates the bandwidth used from the virtual desktop to this user for the USB port-related traffic.	Kbps	
Input line speed:	Indicates the average line speed of all the sessions of this user to the desktop.	KB/Sec	
Output line speed:	Indicates the average line speed from the desktop to this user.	KB/Sec	
Bandwidth usage:	Indicates the percentage HDX bandwidth consumption of this user.	Percent	Compare the value of this measure across users to know which user is consuming the maximum HDX bandwidth.

4.2.2 Virtual Desktop Client's Network Connection Test

A Virtual Desktop Infrastructure (VDI) is a shared environment in which multiple users connect to desktops hosted by virtual machines executing on an ESX host from remote terminals using the Remote Desktop Protocol (RDP). One of the key factors influencing user experience in such an environment is the latency seen by the users when connecting to a virtual desktop. High network latencies or packet losses during transmission can cause significant slow-downs in request processing by the desktop. Hence, monitoring latencies between the virtual desktop and individual client terminals is important.

The Virtual Desktop Client's Network Connection test is executed by the eG agent on an ESX host. This test auto-discovers the virtual desktops on the ESX host, the users who are currently logged on to each of the

virtual desktops, and the IP address from which they are connecting to the virtual desktops. For each user, the test monitors the quality of the link between the client and the virtual desktop.

Using this test, an administrator can identify user sessions that are being impacted by high latencies or by excessive packet drops. In some cases, a virtual desktop may regard a user session as active, even though the network link connecting the user terminal to the virtual desktop has failed. The **Virtual Desktop Client's Network Connection** test alerts administrators to such situations.

This test will work only on Windows VMs.

Target of the test : An ESX host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of outputs for every user currently connected to the virtual desktop.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be

created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In

some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how

they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

- **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
- **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none*

indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

12. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

13. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to Yes (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their virtual machine name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **REPORTUNAVAILABILITY** – By default, this flag is set to **No**. This implies that, by default, the test will not report the unavailability of network connection between a user terminal and a virtual desktop. In other words, if the Packet loss measure of this test registers the value *100%* for any user, then, by default, this test will not report any measure for that user; under such circumstances, the corresponding user name will not appear as a descriptor of this test. You can set this flag to **Yes**, if you want the test to report and alert you to the unavailability of network connection between a user terminal and a virtual desktop.

17. **PACKETSIZE** - The size of packets used for the test (in bytes)

18. **PACKETCOUNT** - The number of packets exchanged between the virtual desktop and the user terminal during the test

19. **TIMEOUT** - How long after transmission should a packet be deemed lost (in seconds)
20. **PACKETINTERVAL** - Represents the interval (in milliseconds) between successive packet transmissions during the execution of this test.
21. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Number of sessions:	Indicates the current number of sessions for a particular user	Number	The value 0 indicates that the user is not currently connected to the virtual desktop. The detailed diagnosis of this measure lists the IP addresses of the clients from which this user connected to the virtual desktop. This will tell you if the same user launched session from multiple sessions. Moreover, if a user is unable to establish a network connection with a virtual desktop, administrators can use the detailed diagnosis to accurately identify the network connection – i.e., the connection from which client terminal to the virtual desktop - that is experiencing issues.
Average delay:	Indicates the average delay between transmission of a request by the agent on a virtual desktop and receipt of the response back from	Secs	Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when connecting to a virtual desktop.

Measurement	Description	Measurement Unit	Interpretation
	the user terminal.		
Minimum delay:	Indicates the minimum delay between transmission of a request by the agent on a virtual desktop and receipt of the response back from the user terminal.	Secs	A significant increase in the minimum round-trip time is often a sure sign of a poor link between the desktop and a user's terminal.
Packet loss:	Indicates the percentage of packets lost during data exchange between the virtual desktop and the user terminal.	Percent	Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing slowdowns because of poor performance on the network links between their terminals and the virtual desktop.

Note:

- If the same user is connecting to the virtual desktop from multiple client terminals, the value of the *Number of sessions*, *Average delay*, and *Packet loss* measures will be averaged across all the sessions of that user. The *Minimum delay* measure, on the other hand, will display the least value reported for *Minimum delay* across all the sessions of that user.
- When a user logs out, the number of sessions will be reduced by 1. If the number of user sessions becomes 0, the corresponding entry for that user in the eG user interface will be removed after a short period of time.

4.2.3 PCoIP Session - VM Test

PCoIP - PC over IP - is a proprietary protocol for remote workstation and desktop resolution. VMware View supports PCoIP to deliver virtual desktops to users connecting to the VDI. Since PCoIP recognizes different types of content and then uses different compression algorithms based on the content type, it is often considered ideal to deliver on the VDI promise of a rich user experience.

The key factors influencing user experience in such cases are the latencies experienced by the user while connecting to the desktop via PCoIP and the bandwidth used when a user interacts with a virtual desktop. High latency and excessive bandwidth consumption can often slowdown access to desktops, thereby significantly delaying subsequent user operations. Hence, monitoring the latency and bandwidth usage of the PCoIP communication channel between the user terminal and the virtual desktops is essential.

The PCoIP Session - VM test auto-discovers the virtual desktops on the ESX host and the users who are currently connected to each desktop via PCoIP. For each such user, the test monitors the communication between a user and the virtual desktop, and reports the following:

- The latency experienced by each user session;
- The bandwidth used by the incoming and outgoing data/audio/multimedia traffic transacted by the PCoIP communication channel between each user and virtual desktop;

Using this test, an administrator can identify user sessions that are being impacted by high latency and abnormal bandwidth usage. In addition, the test also reveals the type of traffic that is causing excessive bandwidth usage, thereby providing pointers to how the client configuration can be fine-tuned in order to reduce bandwidth consumption and improve performance.

This test is relevant only where VMware View is used to broker connections between the user and the desktops. Hence, this test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence: Agents -> Tests -> Enable/Disable, pick *VMware vSphere ESX* as the **Component type**, set *Performance* as the **Test type**, choose this test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every user who is connected to a virtual desktop via PCoIP.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called

the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the

vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
    - **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **4.2.3** .
    - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of

their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

12. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

13. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to Yes (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their virtual machine name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Round trip time:	Indicates the round trip latency between the virtual desktop and this user terminal.	Secs	Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when connecting to a virtual desktop.
Data received rate:	Indicates the rate at which data was received by this user from the virtual desktop.	Kbit/Sec	Comparing the value of each of these measures across users will enable administrators to quickly and accurately identify users who are consuming the maximum bandwidth. Once you zero-in on the user, you can compare the Data received rate of that user with the Data sent rate to know when the user consumed more bandwidth - when receiving data or while sending data?
Data sent rate:	Indicates the rate at which data was sent by this user to the virtual desktop.	Kbit/Sec	
Audio data received rate:	Indicates the bandwidth used while transmitting sound/audio to this user.	Kbit/Sec	Comparing these values across users will reveal which user is sending/receiving bandwidth- intensive sound/audio files over PCoIP.
Audio data sent rate:	Indicates the bandwidth used while receiving sound/audio from this user.	Kbit/Sec	
Imaging data received rate:	Indicates the bandwidth used when sending imaging data to this user.	Kbit/Sec	Comparing these values across users will reveal which user is sending/receiving bandwidth- intensive images over PCoIP.
Imaging data sent rate:	Indicates the bandwidth used when receiving imaging data from this user.	Kbit/Sec	
Imaging decoder capability rate:	Indicates the currrent estimate of the decoder processing capability.	Kbit/Sec	

Measurement	Description	Measurement Unit	Interpretation
Incoming bandwidth rate:	Indicates the overall bandwidth used by incoming PCoIP packets.	Kbit/Sec	Comparing the values of these measures across users will reveal which user is performing bandwidth-intensive operations over the PCoIP channel.
Outgoing bandwidth rate:	Indicates the overall bandwidth used by outgoing PCoIP packets.	Kbit/Sec	
USB data received:	Indicates the bandwidth used when this user received USB data over the PCoIP channel.	Kbit/Sec	Comparing the values of these measures across users will reveal which user is sending/receiving bandwidth-intensive USB data over the PCoIP channel.
USB data sent:	Indicates the bandwidth used when this user sent USB data over the PCoIP channel.	Kbit/Sec	
Received packets lost:	Indicates the percentage of packets received by this user that were lost.	Percent	A high value for these measures is indicative of a bad network connection between the user terminal and the virtual desktop.
Transmitted packets lost:	Indicates the percentage of packets transmitted by this user that were lost.	Percent	
Imaging encoded frames:	Indicates the number of imaging frames that were encoded per second.	Frames/Sec	

4.2.4 Blast Session - VM Test

The BLAST Session protocol allows files on a remote computer to be manipulated - i.e., to be deleted, renamed, or printed on the remote. Owing to the improved remote file management capabilities that Blast offers, VMware now supports the Blast protocol too for user communication with virtual desktops. Because of this, there is a need to know which users have connected to virtual desktops via Blast and how the experience of each user is. This is why, the Blast Session - VM test auto-discovers the users who are connected to virtual desktops via the Blast protocol, and measures the bandwidth usage, frames processing ability, throughput, and time taken to establish each user's session. In the process, bandwidth-hungry, latent sessions can be accurately isolated.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every user who is connected to a virtual desktop via the Blast protocol

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5**.

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6.
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with

'Read-only' privileges.

Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**.

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically

displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1**.

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the

given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.

- **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1**.

- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **4.2.4**.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

12. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

13. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to Yes (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their virtual machine name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Round trip time:	Indicates the round trip latency between the virtual desktop and this user terminal.	Seconds	Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when connecting to a virtual desktop. via Blast.
Bandwidth	Indicates the bandwidth	Kbps	Compare the value of this measure

Measurement	Description	Measurement Unit	Interpretation
	usage of all sessions of this user.		across users to know which user is consuming the maximum bandwidth.
Frame rate	Indicates the rate at which frames are processed during this user's sessions.	Frames/Sec	FPS is how fast your graphics card can output individual frames each second. It is the most time- tested and ideal measure of performance of a GPU. Higher the value of this measure, healthier is the GPU.
Throughput	Indicates the network throughput of this user's sessions.	Kbps	A poor network throughput could cause latencies in network communication.

4.2.5 User Profile Management - VM

User logon is a complex and resource intensive process in a VDI environment, and is a key determinant of the quality of a user's experience with the VDI service. This process is initiated when a desktop broker's load balancing algorithm selects the virtual desktop where a published application or desktop, which a user has selected, will be started and ends when the application or desktop is running and the user is able to interact with it.

Delays in the user logon process can therefore serve as key spoilers of a user's experience with the desktop service, causing significant loss of revenue and reputation in mission-critical VDI environments.

One of the common causes for delays in user logons is a delay in the loading of user profiles. To reduce the time taken to load profiles and thus minimize the user logon time, VDI environments where user connections are brokered through the Citrix XenDesktop Broker, use the Citrix Profile Management solution. Citrix Profile Management is a profile type that supersedes all other profiles for the user.

During logon, the Profile management service manages the user settings in a user profile. This service helps minimize the user logon time by enabling administrators to exclude (and include) certain files and folders in order to prevent extraneous settings from needlessly being copied with the profile. For example, some applications may create folders and files that account for tens or hundreds of megabytes - data that is really not required. By excluding these items, the profile is thus smaller, and smaller profiles load faster. Alternatively, you could elect to only include specific files and folders, thus keeping to a minimum the amount of profile data being managed within the user‘s profile.

Also, upon logoff, the Profile management service merges back only changed user settings to the centrally stored user settings (user‘s store).

In environments where the Citrix Profile Management service is utilized therefore, the user experience with the VDI service greatly depends upon how efficient the service is.

Note:

This test is relevant only where the Citrix XenDesktop Broker is used to broker connections between the user and the desktops.

To ascertain the efficiency of the Citrix Profile Management service, VDI administrators may have to periodically track the logon/logoff duration and profile size of each user to the virtual desktops operating on a target virtual host. Doing so will enable these administrators to determine whether/not the Profile management service has succeeded in minimizing both user logon times and profile sizes. The **User Profile Management - VM** test helps administrators perform this check at pre-configured intervals. The 'per-user' performance results reported by this test will not only enable administrators to judge the effectiveness of the Profile management service in its entirety, but will also shed light on those user logons/logoffs that are still experiencing delays; this provides insights into how the service can be fine-tuned to enhance the VDI experience of such users.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence: Agents -> Tests -> Enable/Disable, pick *VMware vSphere VDI* as the **Component type**, set *Performance* as the **Test type**, choose this test from the disabled tests list, and click on the **>>** button to move the test to the **ENABLED TESTS** list.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every user who is connected to a virtual desktop via ICA.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5**.

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it

will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .

- **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section Section **2.2** of this document to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**.

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1**

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is

obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1**.
    - **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **4.2.5**.
    - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**,

and **ADMIN PASSWORD** parameters to *none*.

11. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
12. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
13. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to Yes (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their virtual machine name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Logon Duration:	Indicates the duration of logon processing for this user.	Secs	This value helps to measure the reduction in logon times when the Profile Management service 'streams' the profile. Ideally therefore, this value should be low. A high value or a consistent increase in the value of this measure could indicate that profile loading still takes a lot of time at logon - this could be owing to a large profile size. You can then check the value reported by the Logon Bytes measure to know the profile size at logon. If profile sizes continue to grow at logon despite the use of Profile management, it is indicative of the ineffectiveness of profile management. You may then have to fine-tune the feature to further reduce the profile size by excluding more unnecessary files from the profile, or you may have to explore other options such as roaming profiles, mandatory profiles, etc.
Logon Bytes:	Indicates the size of this user's profile when it is retrieved from the user's store at logon.	MB	Ideally, the value of this measure should be low. A low profile size could result in faster profile loading at logon, lesser time to login, and consequently, a richer user experience with the VDI service. If profile sizes continue to grow despite the use of Profile management, it is indicative of the ineffectiveness of profile management. You may then have to fine-tune the feature to further reduce the profile size by excluding more unnecessary files from the profile.
Logoff Duration:	Indicates the duration of logoff processing for this user.	Secs	A low value is desired for this measure. A high value could indicate that the profile management service takes too

Measurement	Description	Measurement Unit	Interpretation
			long to update the user's store with changes in the user settings. This could be because of a bad network connection between the virtual desktop and the user's store, or because too many changes are waiting to be written to the user store.
Logoff Bytes:	Indicates the size of this user's profile when it is copied to the user store at logoff.	MB	This measure provides a fair idea of the volume of changes that were copied to the user's store at logoff.
Local Profile Setup Duration:	Indicates the time taken to create or prepare this user's profile on the local computer.	Secs	A low value is desired for these measures. If a user complaints of delays during logon, you can use the value of these measures to determine where the VDI service is spending too much time - is it when setting up the local profile? or is it when deleting the local profile?
Delete Local Profile Duration:	Indicates the time spent deleting this user's local profiles during the initial migration.	Secs	
Processed Logon Files - Under 1KB:	Indicates the number of locally copied files for this user's profile that are synchronized during logon and categorized by the file size of 1KB.	Number	All the *Processed Logon Files* measures help VDI administrators to understand whether/not 'profile streaming' (performed by the Profile Management service) has helped in reducing the number of locally copied files during logon. All the *Processed Logoff Files* measures help VDI administrators to understand how many files changed when the user session was in progress.
Processed Logoff Files - Under 1KB:	Indicates the number of locally copied file for this user's profile that are synchronized during logoff and categorized by the file size of 1KB.	Number	
Processed Logon Files from 1KB to 10KB:	Indicates the number of locally copied files for this	Number	

Measurement	Description	Measurement Unit	Interpretation
	user's profile that are synchronized during logon and categorized by the file size ranging from 1KB to 10KB.		
Processed Logoff Files from 1KB to 10KB:	Indicates the number of locally copied files for this user's profile that are synchronized during logoff and categorized by the file size ranging from 1KB to 10KB.	Number	
Processed Logon Files from 10KB to 100KB:	Indicates the number of locally copied files for this user's profile that are synchronized during logon and categorized by the file size ranging from 10KB to 100KB.	Number	
Processed Logoff Files from 10KB to 100KB:	Indicates the number of locally copied files for this user's profile that are synchronized during logoff and categorized by the file size ranging from 1KB to 10KB.	Number	All the *Processed Logon Files* measures help VDI administrators to understand whether/not 'profile streaming' (performed by the Profile Management service) has helped in reducing the number of locally copied files during logon. All the *Processed Logoff Files* measures help VDI administrators to understand how many files changed when the user session was in progress.
Processed Logon Files from 100KB to 1MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logon and categorized by the file size ranging from 100KB to 1MB.	Number	
Processed Logoff Files from 100KB to	Indicates the number of	Number	

Measurement	Description	Measurement Unit	Interpretation
1MB:	locally copied files for this user's profile that are synchronized during logoff and categorized by the file size ranging from 100KB to 1MB.		
Processed Logon Files from 1MB to 5MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logon and categorized by the file size ranging from 1MB to 5MB.	Number	
Processed Logoff Files from 1MB to 5MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logoff and categorized by the file size ranging from 1MB to 5MB.	Number	All the *Processed Logon Files* measures help VDI administrators to understand whether/not 'profile streaming' (performed by the Profile Management service) has helped in reducing the number of locally copied files during logon. All the *Processed Logoff Files* measures help VDI administrators to understand how many files changed when the user session was in progress.
Processed Logon Files Above 5MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logon and categorized by the file size above 5MB.	Number	
Processed Logoff Files Above 5MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logoff and categorized by the file size above 5MB.	Number	

4.2.6 Domain Time Sync – VM Test

Time synchronization is one of the most important dependencies of windows. A time protocol is responsible for determining the best available time information and converging the clocks to ensure that a consistent time is maintained across systems. By default, windows support a tolerance of plus or minus five minutes for clocks. If the time variance exceeds this setting, clients will be unable to authenticate and in the case of domain controllers, replication will not occur. It implements a time synchronization system based on Network Time Protocol (NTP).

NTP is a fault-tolerant, highly scalable time protocol and it is used for synchronizing computer clocks by using a designated reference clock. A reference clock is some device or machinery that spits out the current time. The special thing about these things is accuracy. Reference clocks must be accurately following some time standard. NTP will compute some additional statistical values based on the current time reported by the reference clock, which will describe the quality of time it sees. Among these values are: offset (or phase), jitter (or dispersion), frequency error, and stability. Thus each NTP server will maintain an estimate of the quality of its reference clocks and of itself.

This test reports the time difference between the reference clock and that of the target environment, and thus helps assess the quality of time seen by the Windows VM. With the help of this test, you can also easily determine whether the reference time changed recently.

Note:

This test reports metrics for Windows VMs only.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence: Agents -> Tests -> Enable/Disable, pick *VMware vSphere ESX* as the **Component type**, set *Performance* as the **Test type**, choose this test from the disabled tests list, and click on the **>>** button to move the test to the **ENABLED TESTS** list.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every user to each Windows virtual desktop on the vSphere/ESX server being monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been

discussed hereunder:

- **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

  ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for

the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case,

the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.
11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.
13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication

with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

- **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
- If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

  Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

  If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .
- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other

words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
NTP offset:	Indicates the time difference between the local clock and the designated reference clock.	Secs	For a tiny offset, NTP will adjust the local clock; for small and larger offsets, NTP will reject the reference time for a while. In the latter case, the operating system's clock will continue with the last corrections effective while the new reference time is being rejected. After some time, small offsets (significantly less than a second) will be slewed (adjusted slowly), while larger offsets will cause the clock to be stepped (set anew). Huge offsets are rejected, and NTP will terminate itself, believing something very strange must have happened.

4.2.7 Browser Activity – VM Test

When a user complains of a virtual desktop slowdown, administrators will have to instantly figure out if that VM is experiencing a resource crunch, and if so, which process/application on the desktop is contributing to it. One of the common reasons for CPU/memory contentions and handle leaks on a virtual desktop is web browsing! If a user to a virtual desktop browses resource-intensive web sites, it is bound to result in over-usage of the resources allocated to that VM, which in turn degrades the performance of not just that VM but even the other VMs on that host. While the **System Details – VM** test can lead administrators to the exact browser application that is consuming the CPU/memory resources of the VM excessively, it does not provide visibility into the precise websites that were been browsed when the resource contention occurred. This is where the **Browser Activity – VM** test helps. For each web browser that is being accessed by a user per virtual desktop, this test reports how every browser uses the allocated CPU, memory, and disk resources and

reveals the number and URLs of the web sites that are being accessed using each browser. This way, the test not only points administrators to resource-hungry browsers, but also indicates which web sites were being accessed using that browser.

Note:

- This test will report metrics only if the Windows VM being monitored uses the .Net framework v3.0 (or above).
- This test will not be able to monitor the Microsoft Edge browser on Windows 10 VMs.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every user to each Windows virtual desktop on the vSphere/ESX server being monitored.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5**.

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6**.

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it

requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG

agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent

obtains the inside view of all VMs on an ESX host by default.

Note:

While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>**

(on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1**.

- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1**.
- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
16. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
17. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the

capability, click on the **Off** option.

The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

- The eG manager license should allow the detailed diagnosis capability
- Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Running browser instances:	Indicates the number of instances of this browser currently running on this virtual desktop.	Number	Use the detailed diagnosis of this measure to know how much resources were utilized by each instance of a browser, so that the resource-hungry instance can be isolated.
Recent web sites:	Indicates the number of websites that were accessed using this browser on this virtual desktop during the last measurement period.	Number	Use the detailed diagnosis of this measure to know which web sites are being accessed using a browser.
CPU utilization:	Indicates the percentage CPU usage of this browser on this virtual desktop.	Percent	Compare the value of this measure across browsers to know which browser consumed the maximum CPU on a desktop. If the value of this measure is close to 100% on that desktop, it indicates excessive CPU usage by the browser. You may then want to use the detailed diagnosis of the Recent web sites measure to know which web sites are being accessed using that browser, which caused CPU usage to soar.
Memory used:	Indicates the percent usage of memory by this browser on this virtual desktop.	Percent	Compare the value of this measure across browsers to know which browser consumed the maximum

Measurement	Description	Measurement Unit	Interpretation
			memory on a desktop. If the value of this measure is close to 100% on that desktop, it indicates excessive memory usage by the browser. You may then want to use the detailed diagnosis of the Recent web sites measure to know which web sites are being accessed using that browser, which caused CPU usage to soar.
Handles used:	Indicates the number of handles opened by this browser on this virtual desktop.	Number	Compare the value of this measure across browsers to know which browser opened the maximum number of handles on a desktop. If the value of this measure consistently increases on that desktop, it indicates that the corresponding browser is leaking memory. You may then want to use the detailed diagnosis of the Recent web sites measure to know which web sites are being accessed using that browser, which caused the memory leak.
Disk reads:	Indicates the rate at which this browser read from the disks supported by this virtual desktop.	KB/Sec	A high value for these measures indicates that the browser is generating high disk I/O. You may then want to use the detailed diagnosis of the Recent web sites measure of this browser to know which web sites on the browser are responsible for the high disk I/O.
Disk writes:	Indicates the rate at which this browser read from the disks of this virtual desktop.	KB/Sec	
Disk IOPS:	Indicates the rate of read and write operations performed by this browser on the disks of this virtual desktop.	Operations/Sec	A high value for this measure indicates that the browser is generating high disk I/O. You may then want to use the detailed diagnosis of the Recent web sites measure of this browser to know which web sites on the browser are responsible for the high disk I/O.

Measurement	Description	Measurement Unit	Interpretation
Page faults:	Indicates the rate at which page faults by the threads executing in this browser are occurring on this virtual desktop.	Faults/Sec	Ideally, the value of this measure should be low. A high value for a browser is a cause for concern. You may then want to use the detailed diagnosis of the Recent web sites measure of this browser to know which web sites on the browser are responsible for page faults.

The detailed diagnosis of the *Running browser instances* measure reveals the process ID of each browser instance that is currently running on the virtual desktop and the resource usage of each instance. This way, you can easily and accurately identify the instance that is consuming resources excessively.

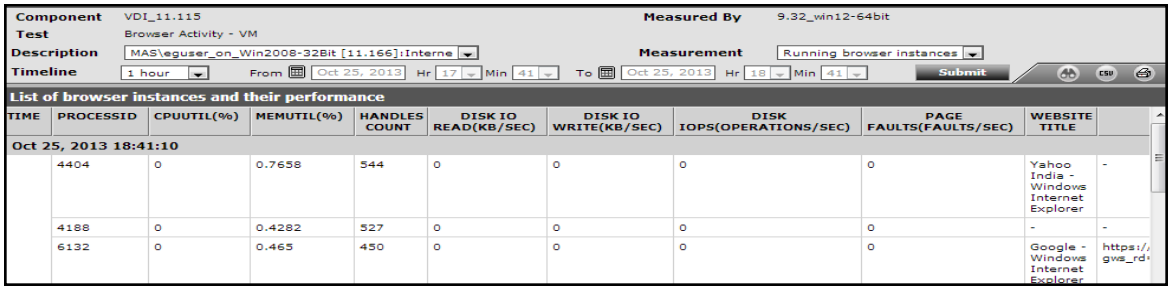


Figure 4.7: The detailed diagnosis of the Running browser instances measure

The detailed diagnosis of the *Recent web sites* measure reveals the names and URLs of the web sites that are being accessed using a browser.

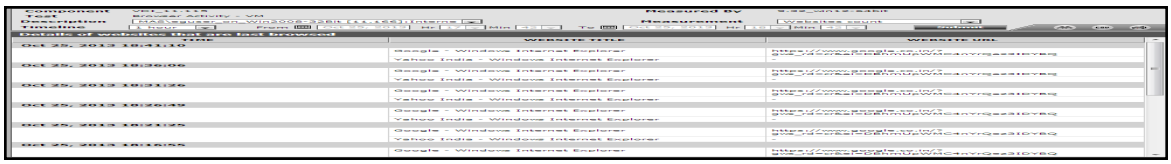


Figure 4.8: The detailed diagnosis of the Recent web sites measure

4.2.8 Personal vDisk – VM Test

The personal vDisk retains the single image management of pooled and streamed desktops while allowing people to install applications and change their desktop settings.

Unlike traditional Virtual Desktop Infrastructure (VDI) deployments involving pooled desktops, where users lose their customizations and personal applications when the administrator alters the base virtual machine (VM), deployments using personal vDisks retain those changes. This means administrators can easily and centrally manage their base VMs while providing users with a customized and personalized desktop experience.

Personal vDisks provide this separation by redirecting all changes made on the user's VM to a separate disk (the personal vDisk) attached to the user's VM. The content of the personal vDisk is blended at runtime with the content from the base VM to provide a unified experience. In this way, users can still access applications provisioned by their administrator in the base VM.

But, what happens if a personal vDisk runs out of space? Simple! Users will no longer be able to hold on to their customizations, allowing them access to only the base VM and the applications installed therein! This outcome beats the entire purpose of having personal vDisks! If this is to be avoided, then administrators should continuously monitor the usage of the personal vDisks, proactively detect a potential space crunch, determine what is causing the rapid erosion of space on the personal vDisk, and fix the root-cause, before desktop users complain. This is where the **Personal vDisk – VM** test helps.

For each VM on a VMware vSphere server, this test tracks the status and space usage of its personal vDisk and promptly reports errors / abnormal space usage. This way, administrators can accurately identify personal vDisks with very limited space, which VM such personal vDisks are associated with, and what is consuming too much disk space – user profiles? Or user applications?

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every *user_on_VM*.

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it

will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

- **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3**. When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.

7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.
9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

16. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this

frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

17. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Personal vDisk service status:	Indicates whether Citrix Personal vDisk service is running or not on this VM.		The values that this measure can report and their corresponding numeric values have been discussed in the table below: Measure Value / Numeric Value Stopped – 0 Running – 1 Not installed – 2 **Note:** By default, this test reports the **Measures Value**s listed in the table above to indicate the status of the Personal vDisk service. In the graph of this measure however, the same will be represented using the numeric equivalents.
Recompose status:	Indicates the status of the initially provisioned disk or the updated image.	Number	Use the detailed diagnosis of this measure to know for which VM the initial personal vDisk provisioning or image update were unsuccessful and why. The VM can be in one of the following states:

Measurement	Description	Measurement Unit	Interpretation
			• **OK** – The initial provisioning or last image update was successful. • **Disk Init** – This is the first time that the personal vDisk has started or been resized. It is being initialized and partitioned by the service. • **Disk Format** – The personal vDisk is being formatted. • **Updating** – The initial provisioning or an image update is in progress. • **Error (Disk Discovery)** – An error state. An error occurred while discovering the personal vDisk. • **Error (Disk Init)** – An error state. An error occurred while partitioning or formatting the personal vDisk. • **Error (Sys Init)** – An error state. An error occurred while starting the Personal vDisk Service or configuring the personal vDisk. • **Error (Update)** – An error state. An error occurred during the initial provisioning or the last image update. • **Unknown** – An error state. An error occurred but the cause is unknown.
Space used by user applications:	Indicates the amount of space used by applications installed on the personal vDisk attached to this VM.	MB	Personal vDisks have two parts, which use different drive letters and are by default equally sized. One part comprises a Virtual Hard Disk file (a .vhd file). This contains items such as applications installed in C:\Program Files. By default, this part uses drive V: but is hidden from users. These measures indicate how much space has been allocated to this .vhd file

Measurement	Description	Measurement Unit	Interpretation
			and how much of the allocated space has been utilized by user applications contained in this file.
Space allocated for user applications:	Indicates the amount of space allocation for storing user applications on the personal vDisk attached to this VM.	MB	A high value for the *Space used by user applications* and *Space utilized by user applications* measures is indicative of excessive space used by user applications. You can compare the value of these measures across VMs to know which user to which VM has utilized too much space reserved for user applications on the personal vDisk. If the value of the Space utilized by user applications measure grows close to 100% for any VM, it implies that potentially, the user to that VM will not be able to install any applications on the personal vDisk; nor access any applications.
Space utilized by user applications:	Indicates the percentage of allocated space used by applications installed on the personal vDisk attached to this VM.	Percent	
Space used by user profiles:	Indicates the amount of space used for storing user profiles on the personal vDisk attached to this VM.	MB	Personal vDisks have two parts, which use different drive letters and are by default equally sized. One part comprises C:\Users (in Windows 7) or C:\Documents and Settings (in Windows XP). This contains user data, documents, and the user profile. By default this uses drive P:. These measures indicate how much space has been allocated to user profiles and how much of the allocated space has been utilized by user profiles. A high value for the *Space used by user profiles* and *Space utilized by user profiles* measures is indicative of excessive space used by user profiles. You can compare the value of these measures across VMs to know which

Measurement	Description	Measurement Unit	Interpretation
			VM's user profiles are consuming the maximum space on the personal vDisk. If the value of the Space utilized by user profiles measure grows close to 100% for any VM, it implies that potentially, the user to that VM will not be able to store/access any more documents or user data on the personal vDisk .
Space allocated for user profiles:	Indicates the amount of space allocated for storing user profiles on the personal vDisk attached to this VM.	MB	
Space utilized by user profiles:	Indicates the percentage of allocated space that has been used up by user profiles on the personal vDisk attached to this VM.	Percent	
Free space:	Indicates the amount of unused space on the personal vDisk attached to this VM.	MB	Ideally, the value of this measure should be high. You can compare the value of this measure across VMs to know which VM's personal vDisk has the least free space. You may then want to resize that personal vDisk to accommodate more data.
Total size:	Indicates the total size of the personal vDisk attached to this VM.	MB	The minimum size of a Personal vDisk is 3 GB, however a size of 10 GB is recommended.
Space utilized:	Indicates the percentage of space in the personal vDisk attached to this VM that is currently used.	Percent	A consistent increase in the value of this measure is a cause for concern, as it indicates a gradual erosion of free space in the personal vDisk of a VM. By comparing the value of this measure across VMs, you can identify which VM's personal vDisk is running out of space! Once the VM with the space-hungry vDisk is isolated, you may want to compare the value of the *Space utilized by user applications* and *Space utilized by user profiles* measures of that VM, to clearly understand what is occupying too much space in the

Measurement	Description	Measurement Unit	Interpretation
			personal vDisk – is it the user profiles? Or is it the user applications? Based on this inference, you can figure out which drive partition of the personal vDisk has limited free space, and can decide between freeing up space in that partition or allocating more space to the personal vDisk itself.

4.2.9 Virtual Desktop Session Start-up Details Test

Figure 4.9 depicts a typical user logon process to a virtual desktop via XenDesktop broker.

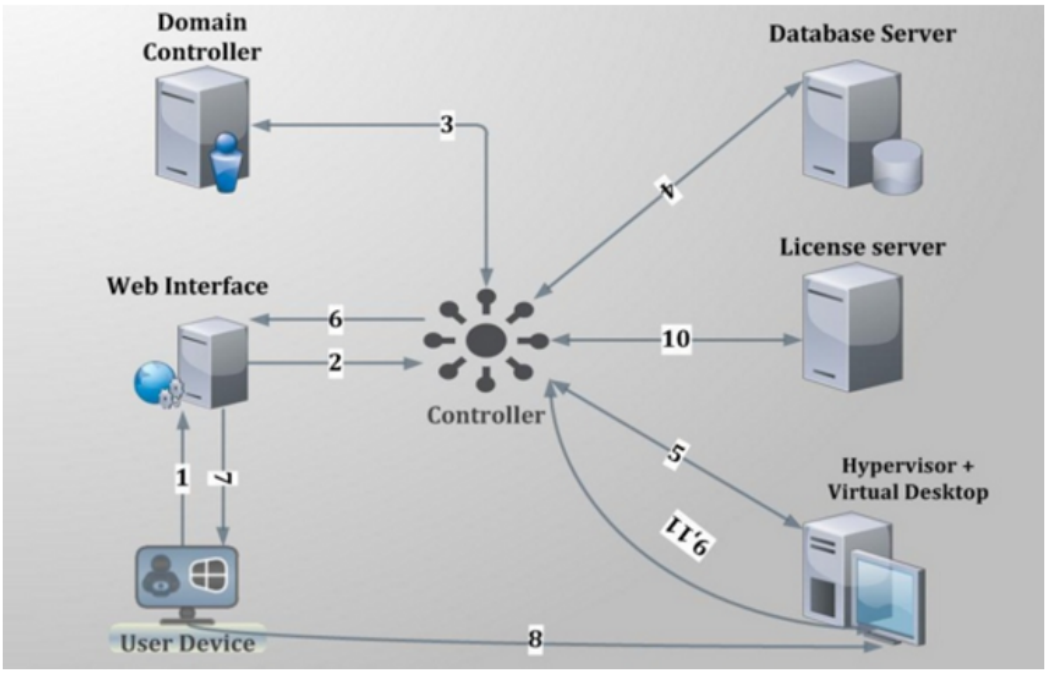


Figure 4.9: Citrix user logon process

The process depicted by Figure 4.9 above has been described below:

1. User provides his/her credentials to the web interface.
2. Web interface forwards the credentials to controller for verification process.
3. Delivery controller transfers these credentials to the domain controller to check if the user is present in the active directory.

4. Once it gets the successful confirmation from AD then controller communicates with site database to check what type of virtual desktop is available for current user.
5. Controller then interacts with the hypervisor layer to gather information about the availability of virtual desktop.
6. Controller then passes the ICA file for user and all the connection information is present inside ICA file so that client can establish the connection.
7. After all the process is complete, the user is assigned the virtual desktop.
8. The user then establishes a connection with the assigned virtual desktop.
9. The virtual desktop again communicates with controller for verification of licensing.
10. Controller checks for license from license server about what type of license is available for user in this current session. License server then communicates back with controller providing the licensing information.
11. Information obtained from license server is then passed to the virtual desktop.

From the discussion above, it can be inferred that login processing happens at two different places – at the delivery controller, and inside the virtual desktop. While login, authentication, and application brokering happen on the delivery controller, session creation and setup happens inside the virtual desktop. A problem in any of these places can result in a poor user experience. Inevitably, these issues result in service desk calls and complaints that "Citrix is slow." Diagnosing login problems has traditionally been a difficult, time-consuming, manual process due to the large number of steps involved. The key to resolving user experience issues therefore, lies in tracking each user's sessions end-to-end, ascertaining the time spent by the session at each step of the logon process – be it on the delivery controller or on the virtual desktop– and accurately identifying where and at what step of the logon process, the slowdown occurred.

To determine the time taken by the entire logon process of a user, isolate logon slowness, and understand where the process was bottlenecked – whether on the delivery controller or on the XenApp server – use the **User Logon Performance** test mapped to the Citrix XA/XD Site component. If the **User Logon Performance** test reveals a problem in session start-up on the virtual desktop, then use the **Virtual Desktop Session Start-up Details** test.

With the **Virtual Desktop Session Start-up Details** test, administrators can receive deep visibility into the virtual desktop end of the Citrix logon process. This test takes an administrator into the virtual desktop, reveals the users who are currently logged on to the virtual desktop, and accurately reports the average time it took for the sessions of each user to start inside the virtual desktop. This way, administrators can rapidly identify which user's sessions are experiencing undue start-up delays.

In addition, the test also provides a break-up of the session start-up duration. This way, the test precisely pinpoints where the delay occurred - when user credentials were obtained? when credentials were validated? during profile loading? during login script execution? when mapping drives or creating printers?

For this purpose, the test categorizes its metrics into *client start-up metrics* and *server start-up metrics*.

The *client start-up metrics* are concerned with timing the operations that occur from the point when the user requests for access to a virtual desktop to the point at which a connection to the virtual desktop is established. While onnection-brokering mechanisms involve components that are not on the physical client

device, the tasks these systems perform have a direct impact on the performance of the connection start-up and are recorded as part of the client-side process.

The *server start-up metrics* are concerned with timing the operations that occur when creating a new session on the virtual desktop. This includes user authentication, client device mapping, profile loading, login scripts execution, and finally, starting the user's desktop.

Note:

This test will report metrics for only those users who are accessing virtual desktops via a XenDesktop broker.

Target of the test : An ESX server host

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every *user_on_VM*.

Configurable parameters for the test

1. **TESTPERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for discovering the VMs on the target ESX server and collecting performance statistics from it. These monitoring/discovery methodologies and their corresponding configuration requirements have been discussed hereunder:

   - **VM discovery using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent discovers the guest operating systems on a physical ESX host. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for discovering the VMs, this test should be configured with an **ESX USER** who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and Terminate Sessions**. To know how to grant this permission to an ESX user, refer to Section **2.6** .

   - **VM discovery using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and discovers the VMs executing on it. While this approach scales well, it

requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to perform guest discovery using the VM information already available in vCenter. In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

Refer to Section **2.2** to know how to create a user on vCenter.

If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and Terminate Sessions** (from vCenter 4.1, this is called the **View and stop sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** . When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests.

This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default. On the other hand, if the eG agent has been configured to use the *VMPerl API* or *CLI* for monitoring (i.e., if the ESX USER parameter is set to *none*), then the status of the SSL flag is irrelevant.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring all the ESX servers in your environment, then you have to set the **SSL** flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled ESX server directly, the eG agent, by default, connects to port 443 of the ESX server to pull out metrics, and while monitoring a non-SSL-enabled ESX server, the eG agent connects to port 80. Similarly, while monitoring an ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG

agent connects to port 80 of vCenter. Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the **SSL** flag. In some environments however, the default ports 80 or 443 might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the ESX server or vCenter in your environment listens so that the eG agent communicates with that port.

8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to discover the guests on the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for guest discovery from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection, the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

   On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section 3.1.1.1 to know how to add a vCenter server using this page. Once the vCenter server is added, its IP address, **ESX USER**, and **ESX PASSWORD** will be displayed against the corresponding text boxes.

   On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for VM information - then set the **VIRTUAL CENTER** parameter to '**none**'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can either be set to **none** or can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to none indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.

10. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: *xp,*lin*,win*,vista. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent

obtains the inside view of all VMs on an ESX host by default.

Note:

While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

11. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

12. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

    Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

13. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.
    - If the guests do not belong to any domain (as in the case of Linux/Solaris guests): In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the ADMIN USER below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

      If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>**

(on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains**- In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- If the inside view using flag is set to 'eG VM Agent (Windows)' - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to **none**.

14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **Yes** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USERflag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtualmachine* name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
16. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
17. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the

capability, click on the **Off** option.

The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

- The eG manager license should allow the detailed diagnosis capability
- Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
User sessions:	Indicates the number of sessions currently open for this user on this virtual desktop.	Number	Use the detailed diagnosis of this measure to view the complete details of this user's session. Such details includes the name and IP address of the client from which the session was launched, when session creation started, and when it ended. With the help of this information, administrators can quickly understand if the session took too long to get created.
Session start- up duration:	Indicates the time taken by this user to complete session start-up inside this virtual desktop.	Secs	Compare the value of this measure across users to know which user's sessions took the longest to start on the virtual desktop. To know what is causing this 'slowness', compare the values reported by all the other 'duration' measures of this test for that user on that virtual desktop. This will quickly lead you to where that user's session start-up is spending the maximum time.
Profile load duration:	Indicates the time taken to load this user's profile.	Secs	If the user's Session start-up duration is high, you may want to compare the value of this measure with that of the other 'duration' measures reported for this user to figure out if a delay in profile loading is what is really ailing that user's logon experience with this virtual desktop.

Measurement	Description	Measurement Unit	Interpretation
			One of the common reasons for high profile load time is the large size of the user profile.
Group policy processing duration:	Indicates the time taken by this user's session to process group policies.	Secs	If a user's Session start-up duration is high, you may want to compare the value of this measure with that of the other 'duration' measures reported for this user to figure out if a delay in group policy processing is what is really ailing that user's logon experience with this virtual desktop. In such a case, you can also use the detailed diagnosis of this measure to figure out the names of the group policy client- side extensions (CSE), the time each CSE took to run, the status of every CSE, and errors (if any) encountered by each CSE. Using these in- depth metrics, Citrix administrators can accurately pinpoint which CSE is impeding speedy group policy processing. Note: Detailed diagnostics will be available for this measure only if the eG VM Agent is deployed on the virtual desktops and the inside view using parameter of this test is set to eG VM Agent. Logon performance improves when fewer Group Policies are applied. Merge GPOs when possible instead of having multiple GPOs.
Login script execution duration:	Indicates the time taken for the login script to execute for this user.	Secs	If a user complains of slowness, then, you can compare the value of this measure with that of the other 'duration' measures of that user to figure out what could have really caused the slowness.

Measurement	Description	Measurement Unit	Interpretation
Start- up client duration:	This is the high-level client-side connection start- up metric. It starts at the time of the request (mouse click) and ends when the connection between this user's client device and the virtual desktop has been established.	Secs	When any user complains of slowness when trying to logon to a virtual desktop, you may want to compare the value of this measure with that of the Session start-up server duration measure of that user to know whether a client-side issue or a server-side issue is responsible for the slowness he/she is experiencing with that virtual desktop. If this comparison reveals that the Start-up client duration of the user is high, it indicates a client- side issue that is causing long start times. In this case therefore, compare the value of the client start- up metrics such as the Application enumeration client duration, Configuration file download client duration, Credentials obtention client duration, ICA file download client duration, Launch page web server duration, Name resolution client duration, Name resolution web server duration, Session look- up client duration, Session creation client duration, and Ticket response web server duration to know what client-side issue is causing the Start- up client duration to be high.
Back-up URL client count:	**This measure is relevant when the Citrix Receiver is the session launch mechanism**. It records the number of back- up URL retries before a successful launch. Note that this is the only start-up metric that is a measure of attempts, rather than time duration.	Number	If this metric has a value higher than 1, it indicates that the Web Interface server is unavailable and the Citrix Receiver is attempting to connect to back-up Web Interface servers to launch the virtual desktop. A value of 2 means that the main Web Interface server was unavailable, but the Citrix Receiver managed to launch the virtual desktop successfully using

Measurement	Description	Measurement Unit	Interpretation
			the first back-up server that it tried. A value higher than 2 means that multiple Web Interface servers are unavailable. Probable reasons for the non-availability of the Web Interface servers include (in order of likelihood): • Network issues between the client and the server. So the administrator should make sure that the Web Interface server is on the network and accessible to the clients. • An overloaded Web Interface server that is not responding (or has crashed for another reason). Try to log on to the server and check the Windows Performance Monitor/Task Manager to see how much memory is in use and so on. Also, review the Event Logs to see if Windows logged any serious errors.
Application enumeration client duration:	**This measure is relevant when the Citrix Receiver is the session launch mechanism**. It measures the time needed by this user's session to retrieve the list of applications from the Web Interface service.	Secs	If the Start-up client duration measure reports a high value for a user, then compare the value of this measure with that of the other client-side metrics such as Configuration file download client duration, Credentials obtention client duration, ICA file download client duration, Launch page web server duration, Name resolution client duration, Name resolution web server duration, Session look-up client duration, Session creation client duration, and Ticket response web server duration to know whether/not slowness in application enumeration is the precise reason why it took the user a long time to establish a session with the virtual desktop.

Measurement	Description	Measurement Unit	Interpretation
Configuration file download client duration:	**This measure is relevant when the Citrix Receiver is the session launch mechanism**. It measures the time this user's session took to retrieve the configuration file from the XML broker.	Secs	If the *Start-up client duration* measure reports a high value for a user, then compare the value of this measure with that of the other client-side metrics such as *Application enumeration client duration, Credentials obtention client duration, ICA file download client duration, Launch page web server duration, Name resolution client duration, Name resolution web server duration, Session look- up client duration, Session creation client duration*, and *Ticket response web server duration* to know whether/not slowness in retrieving the configuration file from the XML server is the precise reason why it took the user a long time an ICA session with the XenApp server.
Credentials obtention client duration:	**This measure is relevant when the Citrix Receiver is the session launch mechanism**. It measures the time required by this user's session to obtain the user credentials.		Note that COCD is only measured when the credentials are entered manually by the user. Because this metric may be artificially inflated if a user fails to provide credentials in a timely manner, it is subtracted from the Start-up client duration. However, in the event that the user manually inputs the credentials, and the value of this measure is higher than that of all the other client start-up metrics that this test reports, it is a clear indicator that any connection delay that the user may have experienced is owing to slowness in obtaining user credentials.
ICA file download duration:	**This measure is relevant when the Citrix Receiver is the session launch mechanism**. This is the	Secs	The overall process here is: a. The user clicks on application icon. b. The user's browser requests the

Measurement	Description	Measurement Unit	Interpretation
	time it takes for this user's client to download the ICA file from the web server.		Web Interface launch page. c. The Web Interface launch page receives the request and starts to process the launch, communicating with the virtual desktop and potentially other components such as Secure Ticket Authority (STA). d. The Web Interface generates ICA file data. e. The Web Interface sends the ICA file data back to the user's browser. f. The browser passes ICA file data to the client. This measure represents the time it takes for the complete process (step 1 to 6). The measure stops counting time when the client receives the ICA file data. The Launch page web server duration measure on the other hand, covers the Web server portion of the process (that is, steps 3 and 4). If the ICA file download duration is high, but the Launch page web server duration is normal, it implies that the server-side processing of the launch was successful, but there were communication issues between the client device and the Web server. Often, this results from network trouble between the two machines, so investigate potential network issues first.
Launch page web server duration:	**This measure is relevant when the Web Interface is the session launch**	Secs	If the value of this measure is high, it indicates at a bottleneck on the Web Interface server.

Measurement	Description	Measurement Unit	Interpretation
	mechanism. It measures the time needed by this user's session to process the launch page (launch. aspx) on the Web Interface server.		Possible causes include: • High load on the Web Interface server. Try to identify the cause of the slow down by checking the Internet Information Services (IIS) logs and monitoring tools, Task Manager, Performance Monitor and so on. • Web Interface is having issues communicating with the other components. Check to see if the network connection between Web Interface and virtual desktop is slow. If the Web server seems okay, consider reviewing the virtual desktop for problems.
Name resolution client duration:	This is the time it takes the XML service to resolve the name of a published application to an IP address.	Secs	This metric is collected when a client device directly queries the XML Broker to retrieve published application information stored in IMA. This measure is only gathered for new sessions since session sharing occurs during startup if a session already exists. When this metric is high, it indicates the XML Broker is taking a lot of time to resolve the name of a published application to an IP address. Possible causes include a problem on the client, issues with the XML Broker, such as the XML Broker being overloaded, a problem with the network link between the two, or a problem in IMA. Begin by evaluating traffic on the network and the XML Broker.
Name resolution web server duration:	**This measure is relevant when the Citrix Receiver is the session launch**	Secs	When this metric is high, there could be an issue with the Web Interface server or the Citrix Receiver, the XML Service,

Measurement	Description	Measurement Unit	Interpretation
	mechanism. It is the time it takes the XML service to resolve the name of this virtual desktop to its IP address.		the network link between the two, or a problem in IMA. Like the *Name resolution client duration* measure, this metric indicates how long it takes the XML service to resolve the name of a virtual desktop to its IP address. However, this metric is collected when a Web Interface site is performing this process on behalf of a launch request it has received from either the Citrix Receiver or from a user clicking a Web Interface page icon.
Session look- up client duration:	Indicates the time this user's session takes to query every ICA session to host the requested published application.	MSecs	The check is performed on the client to determine whether the application launch request can be handled by an existing session. A different method is used depending on whether the session is new or shared.
Session creation client duration:	Indicates the new session creation time.	Secs	In the event of slowness, if the *Start-up client duration* of a user session is found to be higher than the *Session start-up server duration* , you may want to compare the value of this measure with all other client start- up measures to determine whether/not session creation is the process that is slowing down the application launch.
Ticket response web server duration:	**This measure is relevant when the Citrix Receiver or Web Interface is the desktop launch mechanism** . This is the time this user's sessions take to get a ticket (if required) from the STA server or XML service.	Secs	When this metric is high, it can indicate that the Secure Ticket Authority (STA) server or the XML Broker are overloaded.

Measurement	Description	Measurement Unit	Interpretation
Reconnect enumeration client duration:	**This measure is relevant when the Citrix Receiver or Web Interface is the desktop launch mechanism**. This is the time it takes this user's client to get a list of reconnections.	Secs	Compare the value of this measure with that of other client start-up metrics for a user to know what is the actual cause for the client start-up delay.
Reconnect enumeration web server duration:	**This measure is relevant when the Citrix Receiver or Web Interface is the desktop launch mechanism**. This is the time it takes the Web Interface to get the list of reconnections for this user from the XML service.	Secs	Compare the value of this measure with that of other client start-up metrics for a user to know what is the actual cause for the client start-up delay.
Session start- up server duration:	This is the high- level server- side connection start-up metric. It includes the time spent on this virtual desktop to perform the entire start- up operation.	Secs	When this metric is high, it indicates that there is a server- side issue increasing session start times. To zero-in on this issue, compare the values of the server start- up metrics such as Session creation server duration, *Credentials obtention server duration, Program neighbourhood credentials obtention server duration, Credentials obtention network server duration, Credentials authentication server duration, Profile load server duration, Login script execution server duration, Drive mapping server duration, Drive mapping server duration*, and *Printer creation server duration.*
Session creation server duration:	Indicates the time spent by this virtual desktop in creating the session for this user.	Secs	This duration starts when the ICA client connection has been opened and ends when authentication begins. This should not be confused with '*Session start-up*

Measurement	Description	Measurement Unit	Interpretation
			server duration'.
Credentials obtention server duration:	Indicates the time taken by this virtual desktop to obtain the credentials of this user.	Secs	This time is only likely to be a significant if manual login is being used and the server- side credentials dialog is displayed (or if a legal notice is displayed before login commences). Because this metric may be artificially inflated if a user fails to provide credentials in a timely manner, it is not included in the *Session start-up server duration*. However, in the event that the user manually inputs the credentials, and the value of this measure is higher than that of all the other client start-up metrics that this test reports, it is a clear indicator that any connection delay that the user may have experienced is owing to slowness in obtaining user credentials.
Credentials obtentions network server duration:	Indicates the time spent by this virtual desktop performing network operations to obtain credentials for this user.	Secs	This only applies to a Security Support Provider Interface login (a form of pass-through authentication where the client device is a member of the same domain as the server and Kerberos tickets are passed in place of manually entered credentials).
Program neighbourhood credentials obtention server duration:	Indicates the time needed for this virtual desktop to cause the Program Neighborhood instance running on the client ("Program Neighborhood Classic") to obtain this user's credentials.	Secs	As in the case of the Credentials obtention server duration metric, because this metric may be artificially inflated if a user fails to provide credentials in a timely manner, it is not included in the Session start-up server duration.

Measurement	Description	Measurement Unit	Interpretation
Credentials authentication server duration:	Indicates the time spent by this virtual desktop when authenticating the user's credentials against the authentication provider, which may be Kerberos, Active Directory® or a Security Support Provider Interface (SSPI).	Secs	Where server-side issues are causing user experience to deteriorate, you can compare the value of this measure with that of all the other server start-up metrics that this test reports – i.e., Session creation server duration, Credentials obtention server duration, Program neighbourhood credentials obtention server duration, Credentials obtention network server duration, Profile load server duration, Login script execution server duration, Drive mapping server duration, Drive mapping server duration, and Printer creation server duration – to know what is the root-cause of delays in server start-up.
Profile load server duration:	Indicates the time required by this virtual desktop to load this user's profile.	Secs	If this metric is high, consider your Terminal Services profile configuration. Citrix Consulting has found that when customers have logon times greater than 20 seconds, in most cases, this can be attributed to poor profile and policy design. Roaming profile size and location contribute to slow session starts. When a user logs onto a session where Terminal Services roaming profiles and home folders are enabled, the roaming profile contents and access to that folder are mapped during logon, which takes additional resources. In some cases, this can consume significant amounts of the CPU usage. Consider using the Terminal Services home folders with redirected personal folders to mitigate this problem. In general, consider using Citrix Profile management to manage user profiles in Citrix environments. This tool also provides logging capabilities to help

Measurement	Description	Measurement Unit	Interpretation
			isolate profile issues. If you are using Citrix profile management and have slow logon times, check to see if your antivirus software is blocking the Citrix profile management tool.
Login script execution server duration:	Indicates the time needed by this virtual desktop to run this user's login script (s).	Secs	If the value of this measure is abnormally high for any user, consider if you can streamline this user or group's login scripts. Also, consider if you can optimize any application compatibility scripts or use environment variables instead.
Drive mapping server duration:	Indicates the time needed for this virtual desktop to map this user's client drives, devices and ports.	Secs	Make sure that, when possible, your base policies include settings to disable unused virtual channels, such as audio or COM port mapping, to optimize the ICA protocol and improve overall session performance.
Printer creation server duration:	Indicates the time required for this virtual desktop to synchronously map this user's client printers.	Secs	If the configuration is set such that printer creation is performed asynchronously, no value is recorded for this measure as it is does not impact completion of the session start-up. On the other hand, if excessive time is spent mapping printers, it is often the result of the printer autocreation policy settings. The number of printers added locally on the users' client devices and your printing configuration can directly affect your session start times. When a session starts, the virtual desktop has to create every locally mapped printer on the client device. Consider reconfiguring your printing policies to reduce the number of printers that get

Measurement	Description	Measurement Unit	Interpretation
			created - especially if users have a lot of local printers.

As stated earlier, by default, clicking on the **Virtual Desktop** layer, leads you to a page displaying the current status of the individual desktops that have been configured on the ESX host. If you want to override this default setting - i.e., if you prefer to view the tests mapped to the **Virtual Desktop** layer first, and then proceed to focus on individual desktop performance, follow the steps given below:

- Edit the **eg_ui.ini** file in the **<EG_INSTALL_DIR>\manager\config** directory
- Set the **LAYERMODEL_LINK_TO_VIRTUAL** flag in the file to **false**; this is set to **true** by default.
- Save the **eg_ui.ini** file.

Doing so ensures that as soon as the **Virtual Desktop** layer is clicked, the list of tests mapped to that layer appears. If you now want the **Desktop view** of Figure 4.5, simply click on **Back** button in the layer model page.

From the desktop view, you can further drill-down to focus on the health of a particular desktop, by clicking on the icon representing the desktop in Figure 4.5. Figure 4.10 then appears displaying all the performance metrics extracted from that virtual desktop in real-time. You are thus enabled to cross-correlate across the various metrics, and quickly detect the root-cause of current/probable disturbances to the internal health of a desktop. To view the time-of-day variations in a measure, you can view its graph by clicking on that measure in Figure 4.10.

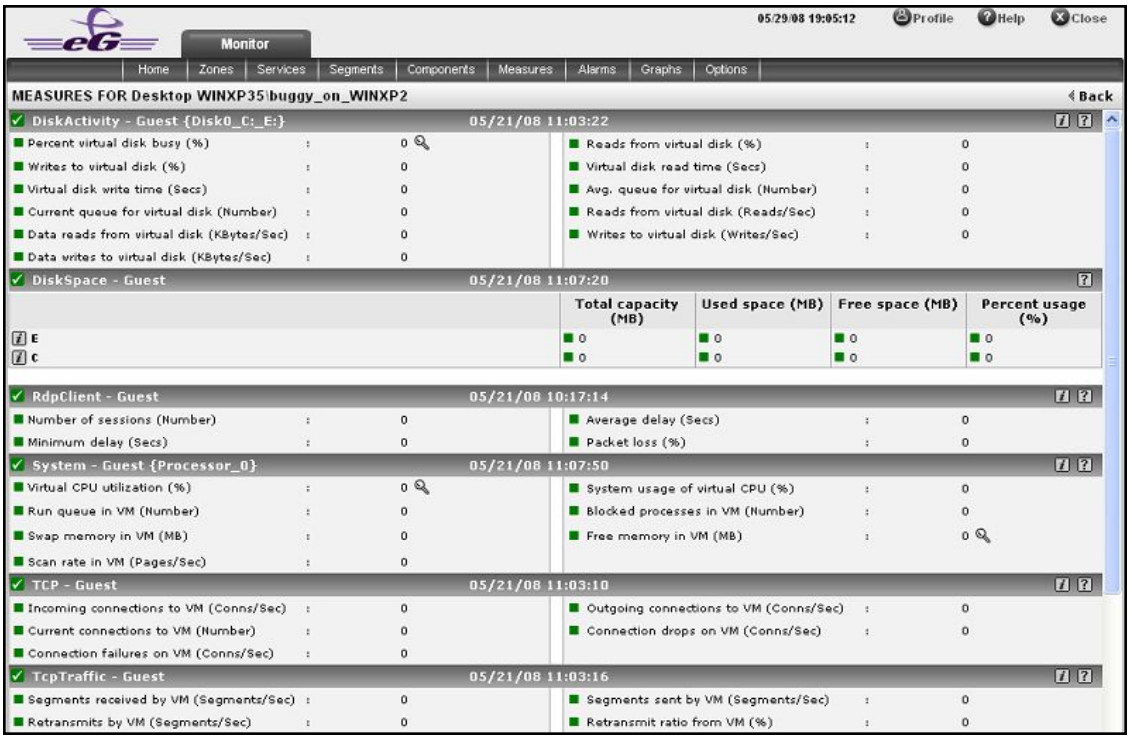


Figure 4.10: The measures pertaining to a particular desktop

You can also view live graphs of pre-configured measures pertaining to an ESX server host and the virtual desktops configured on it, by clicking on the **LIVE GRAPHS** link in Figure 4.5. Alternatively, you can click on the icon that appears in the **Tests** panel of the layer model page when the **Virtual Desktop** layer (see Figure 4.6) is clicked to view the live graph. The resulting graph display (see Figure 4.11) has been organized in such a way that next to every host-pertinent measure graph, the closely related guest-specific measure graph appears. For instance, next to the graph of the 'CPU usage' measure of the Cpu - Host test, you will find a graph of the 'Cpu used' measure of the EsxGuestDetails test. This way, you can easily compare and correlate how well the physical CPU resources are being utilized by both the ESX host and the guests. On the basis of this analysis, you can proactively isolate potential performance issues, and also determine the root-cause of the issue - is it the ESX host? or is it the virtual guest? If you access this page from the **LIVE GRAPH** link in Figure 4.5, then, by default, you will view live graphs pertaining to the *VMware vSphere VDI* server. However, you can select a different virtualized component-type and a different virtualized component using the **type** and **Component** lists (respectively) in Figure 4.11.

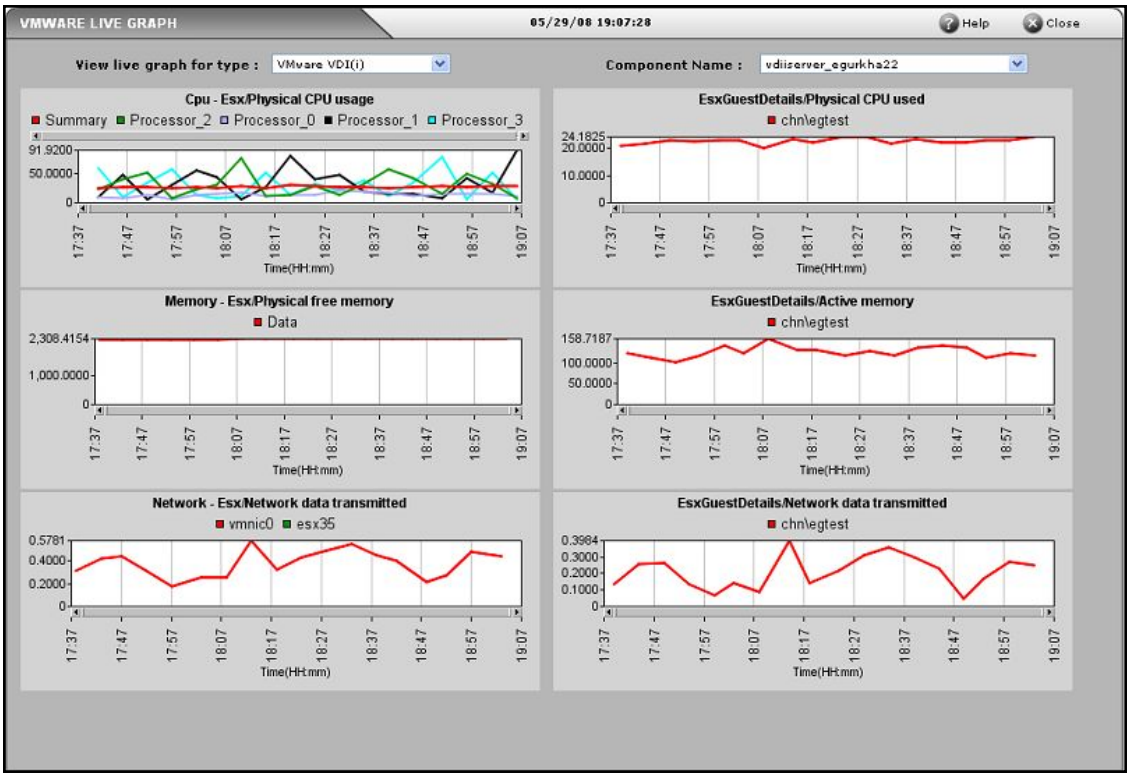


Figure 4.11: Live graph for VMware vSphere VDI server

4.2.10 Virtual Desktop Sessions Details Test

Every user who is logged into a virtual desktop, may not actively use that desktop. In a VDI infrastructure, it is common for users to just log into desktops, and leave them unused for long time periods. Such desktops are a huge resource drain, as they continue to consume resources, regardless of the level of activity on them. Moreover, with many organizations these days providing the work-from-home option to their employees, identification of idle users is paramount, as such users themselves are deemed unproductive resource consumers. Also, this is a good indicator to whether/not the work-from-home facility is effectively used by employees or is abused! Besides, since idle users unnecessarily hold on to desktops, users with genuine needs may not have any desktops to work with. If administrators can quickly identify these idle users and the desktops they are logged into, they can rapidly pull the desktops from such users and assign them to users who can use them effectively. The **Virtual Desktop Sessions Details** test turns the spotlight on these idle users. For each user session on a virtual desktop, this test reports the total duration of the session and the percentage of time for which the session was active. The test also reports the total idle time during the session. From these statistics, administrators can accurately identify those users who are wasting the desktops assigned and resources allocated to them.

Target of the test : An ESX server host

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every user who is currently logged into a virtual desktop

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the specified **HOST** listens. By default, this is NULL.
4. **ESX USER** and **ESX PASSWORD** - In order to enable the test to extract the desired metrics from a target ESX server, you need to configure the test with an **ESX USER** and **ESX PASSWORD**. The user credentials to be passed here depend upon the mechanism used by the eG agent for collecting performance statistics from the ESX server and its VMs. These monitoring methodologies and their corresponding configuration requirements have been discussed hereunder:
   - **Monitoring using the web services interface of the ESX server**: Starting with ESX server 3.0, a VMware ESX server offers a web service interface using which the eG agent collects metrics from the ESX server. The VMware VI SDK is used by the agent to implement the web services interface. To use this interface for monitoring, this test should be configured with an ESX USER who has "Read-only" privileges to the target ESX server. By default, the root user is authorized to execute the test. However, it is preferable that you create a new user on the target ESX host and assign the "Read-only" role to him/her. The steps for achieving this have been elaborately discussed in Section **2.5** .

     ESX servers terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until ESX times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of an ESX user with permissions to **View and stop sessions** (prior to vSphere/ESX server 4.1, this was called the **View and Terminate Sessions** privilege). To know how to grant this permission to an ESX user, refer to 2.6 .
   - **Monitoring using the vCenter in the target environment**: By default, the eG agent connects to each ESX server and collects metrics from it. While this approach scales well, it requires additional configuration for each server being monitored. For example, separate user accounts may need to be created on each server for read-only access to VM details. While monitoring large virtualized installations however, the agents can be optionally configured to monitor ESX servers using the statistics already available with different vCenter installations in the environment.

   In this case therefore, the **ESX USER** and **ESX PASSWORD** that you specify should be that of an *Administrator* or *Virtual Machine Administrator* in vCenter. However, if, owing to security constraints, you prefer not to use the credentials of such users, then, you can create a special role on vCenter with 'Read-only' privileges.

   Refer to Section **2.2** to know how to create a user on vCenter.

   If the ESX server for which this test is being configured had been discovered via vCenter, then the eG manager automatically populates the **ESX USER** and **ESX PASSWORD** text boxes with the vCenter user credentials using which the ESX discovery was performed.

Like ESX servers, vCenter servers too terminate user sessions based on timeout periods. The default timeout period is 30 mins. When you stop an agent, sessions currently in use by the agent will remain open for this timeout period until vCenter times out the session. If the agent is restarted within the timeout period, it will open a new set of sessions. If you want the eG agent to close already existing sessions before it opens new sessions, then you would have to configure all the tests with the credentials of a vCenter user with permissions to **View and stop sessions** (prior to vCenter 4.1, this was called the **View and Terminate Sessions** permission). To know how to grant this permission to a user to vCenter, refer to Section **2.3** .

When the eG agent is started/restarted, it first attempts to connect to the vCenter server and terminate all existing sessions for the user whose credentials have been provided for the tests. This is done to ensure that unnecessary sessions do not remain established in the vCenter server for the session timeout period. Ideally, you should create a separate user account with the required credentials and use this for the test configurations. If you provide the credentials for an existing user for the test configuration, when the eG agent starts/restarts, it will close all existing sessions for this user (including sessions you may have opened using the Virtual Infrastructure client). Hence, in this case, you may notice that your VI client sessions are terminated when the eG agent starts/restarts.

5. **CONFIRM PASSWORD** - Confirm the password by retyping it here.
6. **SSL** - By default, the ESX server is SSL-enabled. Accordingly, the SSL flag is set to **Yes** by default. This indicates that the eG agent will communicate with the ESX server via HTTPS by default.

   Like the ESX sever, the vCenter is also SSL-enabled by default. If you have chosen to use the vCenter for monitoring, then you have to set the SSL flag to **Yes**.
7. **WEBPORT** - By default, in most virtualized environments, the vSphere/ESX server and vCenter listen on port 80 (if not SSL-enabled) or on port 443 (if SSL-enabled). This implies that while monitoring an SSL-enabled vSphere/ESX server directly, the eG agent, by default, connects to port 443 of the vSphere/ESX server to pull out metrics, and while monitoring a non-SSL-enabled server, the eG agent connects to port 80. Similarly, while monitoring a vSphere/ESX server via an SSL-enabled vCenter, the eG agent connects to port 443 of vCenter to pull out the metrics, and while monitoring via a non-SSL-enabled vCenter, the eG agent connects to port 80 of vCenter.

   Accordingly, the **WEBPORT** parameter is set to *80* or *443* depending upon the status of the ssl flag. In some environments however, the default ports *80* or *443* might not apply. In such a case, against the **WEBPORT** parameter, you can specify the exact port at which the vSphere/ESX server or vCenter in your environment listens so that the eG agent communicates with that port.
8. **VIRTUAL CENTER** - If the eG manager had discovered the target ESX server by connecting to vCenter, then the IP address of the vCenter server used for discovering this ESX server would be automatically displayed against the **VIRTUAL CENTER** parameter; similarly, the **ESX USER** and **ESX PASSWORD** text boxes will be automatically populated with the vCenter user credentials, using which ESX discovery was performed.

   If this ESX server has not been discovered using vCenter, but you still want to monitor the ESX server via vCenter, then select the IP address of the vCenter host that you wish to use for monitoring the ESX server from the **VIRTUAL CENTER** list. By default, this list is populated with the IP address of all vCenter hosts that were added to the eG Enterprise system at the time of discovery. Upon selection,

the **ESX USER** and **ESX PASSWORD** that were pre-configured for that vCenter server will be automatically displayed against the respective text boxes.

On the other hand, if the IP address of the vCenter server of interest to you is not available in the list, then, you can add the details of the vCenter server on-the-fly, by selecting the **Other** option from the **VIRTUAL CENTER** list. This will invoke the **ADD VCENTER SERVER DETAILS** page. Refer to Section **3.1.1.1** .

On the other hand, if you want the eG agent to behave in the default manner -i.e., communicate with each ESX server for monitoring it - then set the **VIRTUAL CENTER** parameter to '*none*'. In this case, the **ESX USER** and **ESX PASSWORD** parameters can be configured with the credentials of a user who has at least 'Read-only' privileges to the target ESX server.

9. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section **2.7** for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the **INSIDE VIEW USING** flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the **DOMAIN** specification. Discussed below are the different values that the **DOMAIN** parameter can take, and how they impact the **ADMIN USER** and **ADMIN PASSWORD** specifications:

    - **If the VMs belong to a single domain**: If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.
    - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests)** : In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

      Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of

the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The **ADMIN PASSWORD** in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section **5.1.1** .

- **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORDS** would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section **3.6.1.1** .
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

11. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS S**pecification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the exclude vms text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs.
12. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the** '**inside view**' **of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on an ESX host by default.

    Note:

    While performing VM discovery, the eG agent will not discover the operating system of the VMs

configured in the **IGNORE VMS INSIDE VIEW** text box.

13. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.
14. **REPORT BY USER** - While monitoring a VMware ESX server, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the ESX server are identified using the hostname specified in the operating system. On the other hand, while monitoring VMware Desktop environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
15. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'**.

    If the **REPORT POWERED OS** flag is set to Yes (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their virtual machine name and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total time in session:	Indicates the time that has elapsed since this user logged into this desktop.	Mins	
Active time in last measure period:	Indicates the percentage of time in the last measurement period during which this user actively used this desktop.	Percent	Ideally, the value of this measure should be 100%. A low value for this measure denotes a high level of inactivity recently.
Time since last activity:	Indicates the time that has elapsed since this user performed an action on this desktop.	Mins	A high value for this measure indicates that the user has been idle for a long time. Compare the value of this measure across users to know which user has been idle for the longest time.
Total idle time in	Indicates the total time for	Mins	If the value of this measure is the

Measurement	Description	Measurement Unit	Interpretation
session:	which this user was idle during the session.		same as the value of the *Total time in session* measure for a user, it means that the user has been idle throughout the session. If the value of this measure is close to the value of the *Total time in session* measure for a user, it implies that the user has been idle for a long time. If the value of this measure is much lesser than the value of the *Total time in session* measure for a user, it means that the user has been active for most part of the session.

5

Troubleshooting

5.1 Troubleshooting the Failure of the eG Remote Agent to Obtain the 'Inside View' of a Windows VM

If the eG remote agent is unable to obtain the inside view of a Windows VM, then, first check whether the agent is able to connect to the problem VM. The steps for performing this check will vary depending upon the operating system on which the remote agent executes.

Note:

The steps discussed below apply only when the following conditions are fulfilled:

- The vSphere/ESX servers being monitored should be configured only with Windows VMs;
- All the Windows VMs should belong to a single domain only - i.e., the inside view tests for the target vSphere/ESX server should be configured with a single **ADMIN USER** and **ADMIN PASSWORD** only.

If the eG agent is operating on a Unix host, then the following steps will apply:

1. Login to the Unix host as eG install user.
2. Go to the shell prompt and switch to the **/opt/egurkha/lib** directory.
3. Set the **CLASSPATH** by issuing the following commands at the prompt:

   classpath=.:eg_agent.jar:eg_util.jar:$CLASSPATH

   export classpath
4. Next, set the JRE path by issuing the following commands:

   PATH=/opt/egurkha/jre/bin:$PATH

   export PATH
5. Finally, issue the following command to try connecting to the Windows VM:

   java EgWinConnect -user <username> -password <password> -domain <domainname> -IP <IP_address>

   In this command:

   - Substitute the **<username>** and **<password>** variables with the values that you have configured for the **ADMIN USER** and **ADMIN PASSWORD** parameters (respectively) for all inside view tests. While the **ADMIN USER** value can be easily obtained from the test configuration page in the eG administrative interface, the **ADMIN PASSWORD** cannot be so obtained, as it would be in an encrypted format in the user interface. To know the password therefore, do the following:
   - Open the **eg_agents.ini** file in the **/opt/egurkha/gent/config** directory on the agent host.

- Look for entries related to the inside view tests that the agent has executed on the other Windows VMs on the target vSphere/ESX server.
- These entries will typically include a **-admin_password** parameter, which will be followed by the decrypted **ADMIN PASSWORD**.
- Make a note of the decrypted password, and substitute the **<password>** variable in the **java EgWinConnect** command with it.
- Next, substitute the **<domainname>** variable with the value that you passed to the **DOMAIN** parameter in the test configuration page, and the **<IP_address>** with the IP address of the problematic Windows VM.
- If this command fails, it is a clear indication that the remote agent is unable to communicate with the Windows VM. You will then have to investigate the reasons for the same and fix them in order to ensure that the agent is able to obtain the "inside view" of that VM.

If the eG agent is operating on a Windows host, then the following steps will apply:

1. Login to the Windows host on which the eG agent is executing.
2. Go to the command prompt and switch to the **<EG_INSTALL_DIR>\lib** directory.
3. Issue the following command to set the path to the **<EG_INSTALL_DIR>\JRE\bin** directory.

   set path=<EG_INSTALL_DIR>\jre\bin

4. Set the **CLASSPATH** by issuing the following command at the prompt:

   set classpath=.<eg_install_dir>\lib\eg_agent.jar;<eg_install_dir>\lib\eg_util.jar

5. Finally, issue the following command to try connecting to the Windows VM:

   java EgWinConnect -user <username> -password <password> -domain <domainname> -IP <IP_address>

   In this command:

   - Substitute the **<username>** and **<password>** variables with the values that you have configured for the **ADMIN USER** and **ADMIN PASSWORD** parameters (respectively) for all inside view tests. While the **ADMIN USER** value can be easily obtained from the test configuration page in the eG administrative interface, the **ADMIN PASSWORD** cannot be so obtained, as it would be in an encrypted format in the user interface. To know the password therefore, do the following:
   - Open the **eg_agents.ini** file in the **<EG_INSTALL_DIR>\agent\config** directory on the agent host.
   - Look for entries related to the inside view tests that the agent has executed on the other Windows VMs on the target vSphere/ESX server.
   - These entries will typically include a **-admin_password** parameter, which will be followed by the decrypted **ADMIN PASSWORD**.
   - Make a note of the decrypted password, and substitute the **<password>** variable in the **java EgWinConnect** command with it.
   - Next, substitute the **<domainname>** variable with the value that you passed to the **DOMAIN** parameter in the test configuration page, and the **<IP_address>** with the IP address of the

problematic Windows VM.

A sample command is given below:

C:\eGurkha\lib>java EgWinConnect - user eguser - password C13120CB9E5D4B1423419897808BAE65 -domain mas -ip 192.168.10.216

- If the command is successful, the following output will appear:

```
Attempt to connect and execute for 192.168.10.216
 Output is [[Ok, , , , , Windows IP Configuration, , , , , , , , , Ethernet adapt
 er Local Area Connection 2:, , , , , ,    Connection-specific DNS Suffix  . : ,
 , ,    IP Address. . . . . . . . . . . . : 192.168.10.216, , ,    Subnet Mask .
 . . . . . . . . . . : 255.255.255.0, , ,    Default Gateway . . . . . . . . . :
 192.168.10.2, , , , EgDone 0(0x0)], []]
```

- The command may fail when it encounters one of the following errors. The reasons for these errors and the recommended resolution for the same have been provided below.

Error	Reason	Fix
Couldn't connect to \\<IP_of_ Windows_VM>\ADMIN$ **Logon failure: unknown user name or bad password**	Occurs if the inside-view tests have been configured with an incorrect **DOMAIN**, invalid **ADMIN USER** name, or a wrong **ADMIN PASSWORD**.	Reconfigure the tests with the valid credentials of a *domain administrator*
Couldn't connect to \\<IP_of_ Windows_VM>\ADMIN$ **The network path was not found.**	• Can occur if the **ADMIN$** share has not been enabled on the target Windows VM; • Can occur if the Windows firewall is blocking connection to the VM, or if File/Print Sharing has not been enabled yet.	• Enable the **ADMIN$** share on the target VM. • Provide domain administrator with full access to the **ADMIN$** share. • Try connecting to the Windows VM remotely; if the problem persists: ◦ Reconfigure the Windows firewall to allow communication between the remote agent and the Windows VM; ◦ Reconfigure the Windows firewall to allow File/Print Sharing
Couldn't connect to \\<IP_of_ Windows_	• Can occur if the **ADMIN$** share exists but the domain	Provide domain administrator with full access to the **ADMIN$** share.

Error	Reason	Fix
VM>\ADMIN$ **Access is denied.**	administrator does not have permission to access the shared folder; • Can occur if the **ADMIN$** share exists, but the user with full access to the shared folder is not the domain administrator.	

5.1.1 Troubleshooting the Failure of the eG Remote Agent to Connect to or Report Measures for Linux Guests

By default, the eG agent uses secure shell (SSH) to connect to Linux guests, and collect performance metrics from them. Password Authentication is the default method for SSH connections in eG Enterprise. If the eG agent fails to report measures for a Linux guest or is unable to connect to a guest, it could imply that the Linux guest does not support SSH or that password authentication is not supported by the SSH daemon running on the Linux guest. Under such circumstances, you can perform either of the following:

- Enable Password Authentication in the SSH daemon on the Linux guest; or,
- Implement Key-Based Authentication between eG agent and the SSH daemon of the Linux guest.

If you pick option (1), then follow the steps given below to enable password authentication:

- Login to the Linux guest to be monitored.
- Edit the **sshd_config** file in the **/etc/ssh** directory.
- Check whether the **Password Authentication** flag in the **sshd_config** file is set to **no**. If so, set it to **yes**.
- Then, save the file and restart/signal the SSH daemon (eg., using **kill -1 <sshd_config PID>**).

On the contrary, if you choose to enable key-based authentication [i.e, option (2)], then you will have to generate a public/private key pair. A public/private key pair is available in the **<EG_INSTALL_DIR>\agent\sshkeys** directory (on Windows; on Unix, this will be **/opt/egurkha/agent/sshkeys**) of the eG agent. While the private key is available in the file named **id_rsa**, the public key is contained within the file **authorized_keys**. You now have the option to proceed with the default keys or generate a different key pair. If you decide to go with the keys bundled with the eG agent, do the following:

- To enable key-based authentication, the private key should remain in the **<EG_INSTALL_DIR>\agent\sshkeys** directory (on Windows; on Unix, this will be **/opt/egurkha/agent/sshkeys**), and the public key should be copied to each of the Linux guests to be monitored. To achieve this, first login to the Linux guest to be monitored as the eG user.
- Create a directory named .**ssh** in the **<USER_HOME_DIR>** on the guest operating system, using the command: **mkdir ~/.ssh**.

- Next, copy the **authorized_keys** file from the **<EG_INSTALL_DIR>\agent\sshkeys** directory (on Windows; on Unix, this will be **/opt/egurkha/agent/sshkeys**) on the eG remote agent host to the **<USER_HOME_DIR>/.ssh** directory on the Linux guest.
- Make sure that the permission of the **.ssh** directory and the **authorized_keys** file is **700**.

Finally, on the eG manager host, edit the **<EG_INSTALL_DIR>\manager\config\eg_tests.ini** file. Against the **EgJavaSSHKeyFile** parameter, enter: **agent/sshkeys/id_rsa.pub**, and save the file.

On the other hand, if you want to generate a new key pair, then do the following:

- Login to any Linux host in your environment (even a Linux VM) as an eG user.
- From the **<USER_HOME_DIR>**, execute the command: **ssh-keygen -t rsa**. Upon executing the command, you will be requested to specify the full path to the file to which the key is to be saved. By default, a directory named **.ssh** will be created in the **<USER_HOME_DIR>**, to which the key pair will be saved. To go with the default location, simply press **Enter**.

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/egurkha/.ssh/id_rsa):
```

- Next, you will be prompted to provide a pass phrase. Provide any pass phrase of your choice.

```
Enter passphrase (empty for no passphrase): eginnovations
Enter same passphrase again: eginnovations
```

- If the key pair is created successfully, then the following messages will appear:

```
Your identification has been saved in /hom
e/egurkha/.ssh/id_rsa.
Your public key has been saved in /home/egurkha/.ssh/id_rsa.pub.
The key fingerprint is:
09:f4:02:3f:7d:00:4a:b4:6d:b9:2f:c1:cb:cf:0e:e1 dclements@sde4.freshwater.com
```

- The messages indicate that the private key has been saved to a file named **id_rsa** in the **<USER_HOME_DIR>/.ssh**, and the public key has been saved to a file named **id_rsa.pub** in the same directory. Now, to enable key-based authentication, login to the Linux guest to be monitored as the eG user.
- Create a directory named **.ssh** in the **<USER_HOME_DIR>** on the guest operating system, using the command: **mkdir ~/.ssh**.
- Next, copy the **id_rsa.pub** file from the **<USER_HOME_DIR>/.ssh** directory on the Linux host to the **<USER_HOME_DIR>/.ssh** directory on the Linux guest.
- Ensure that the **id_rsa.pub** file on the Linux guest is renamed as **authorized_keys**.
- Repeat this procedure on every Linux guest to be monitored.
- Then, lock the file permissions down to prevent other users from being able to read the key pair data, using the following commands:

  chmod go-w ~/

  chmod 700 ~/.ssh

chmod go-rwx ~/.ssh/*

- Finally, on the eG manager host, edit the **<EG_INSTALL_DIR>\manager\config\eg_tests.ini** file. Against the **EgJavaSSHKeyFile** parameter, enter: **agent/sshkeys/id_rsa.pub**, and save the file.

Instead of choosing between the authentication modes (Password or Key-based), you can also disable the usage of the Java SSH client, and use **plink** to connect to Linux guests. To achieve this, follow the steps given below:

- Edit the **eg_tests.ini** file in the **/opt/egurkha/manager/config** directory (on Unix; on Windows, this will be **<EG_INSTALL_DIR>\manager\config** directory).
- Set the **JavaSSHForVm** flag in the **[AGENT_SETTINGS]** section of the file to **false**; by default, this is set to **true** indicating that the eG agent uses Java SSH by default. By setting the flag to **false**, you can ensure that the eG agent does not use Java SSH, and instead uses the **plink** command to connect to Linux guests.
- The **plink** command exists in the **<EG_INSTALL_DIR>\lib\vmgfiles** directory (on Windows; on Unix, this will be **/opt/egurkha/lib/vmgfiles**) of the eG agent. To use the **plink** command, you need to explicitly configure the SSH keys, so that the eG agent is able to communicate with the Linux guests using SSH. To do this, follow the steps given below:
- Go to the command prompt and switch to the directory containing the **plink** command.
- Then, execute the **plink** command to connect to any of the Linux-based virtual machines on the vSphere host. The syntax for the **plink** command is as follows:

  plink -ssh <user>@<IP_of_target_host> <command>

  For example, assume that you want to connect to the virtual machine, **192.168.10.7**, as user **john** with password **john**, to know its hostname. The syntax of the **plink** command in this case will be:

  plink -ssh john@192.168.10.7 hostname, where **hostname** is the command to be executed on the remote host for extracting its hostname.
- To ensure that you do not connect to an imposter host, **SSH2.x** presents you with a unique host key fingerprint for that host, and requests your confirmation to save the displayed host key to the cache.

```
The server's host key is not cached in the registry. You have no guarantee that the
server is the computer you think it is.

The server's rsa2 key fingerprint is:<host key>

If you trust this host, enter "y" to add the key to PuTTY's cache and carry on
connecting.

If you want to carry on connecting just once, without adding the key to the cache,
enter "n".
 If you do not trust this host, press Return to abandon the connection.

Store key in cache? (y/n) y
```

Once you confirm the host key storage and provide the user's password to connect to the virtual guest, this message will not appear during your subsequent attempts to connect to any Linux-based virtual machine on the monitored vSphere/ESX host. In other words, the eG agent will be able to execute tests on all Linux guests on the target ESX host without any interruption. Therefore, press **y** to confirm key storage.

5.2 Troubleshooting the Failure of the eG Remote Agent to Connect to or Report Measures for Windows 2008 or Windows Vista VMs

The remote agent may not be able to connect to or collect inside view metrics from a VM running Windows 2008 or Windows Vista, if the **User Access Control** (UAC) feature is enabled on those VMs. In such a case, do the following on each of those VMs to enable the remote agent to connect to them:

1. Click **Start**
2. Type **REGEDIT**
3. Press **Enter**
4. In the left pane, browse to the following folder:

   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
5. Right-click on a blank area in the right pane.
6. Click **New**.
7. Click **DWORD Value**.
8. Type **LocalAccountTokenFilterPolicy**.
9. Double-click the item you just created,
10. Type **1** into the box that appears.
11. Click **OK**.
12. Restart the virtual machine.

6

How to Stop the 'Agentless Monitoring' of vSphere/ESX Servers and VMs?

This chapter briefly discusses the procedures to be followed to ensure that an eG agent on a remote Windows host no longer monitors the following:

- A particular VM on an ESX server (inside view only)
- All the VMs of an ESX server (inside view only)
- An ESX server and its VMs (inside and outside view only)

6.1 How to Stop the "Agentless Monitoring" of a Single VM on an ESX Server?

To achieve this, follow the steps given below:

1. If the VM being 'unmanaged' is a Linux VM, then disable SSH on that VM.
2. In case of a Windows VM however, first, login to the Windows VM as a user with full access to the **ADMIN$** share on that VM.
3. Go to the command prompt, and type the following command therein to remove the **ADMIN$** share:

   net share ADMIN$ /delete

4. Then, proceed to uninstall any eG software on the Windows VM. For this, first determine the mechanism using which the remote Windows agent obtains the *inside view* of the Windows VMs - whether it is using the eG VM Agent? or via NetBIOS? If an eG VM Agent has been deployed on the target Windows VM for obtaining the *inside view*, then uninstall the eG VM agent to make sure that such a VM is no longer monitored. To achieve this, follow the steps given below:

   - Login to the Windows VM.
   - Stop the eG VM agent using the menu sequence: Start -> Programs -> eG Monitoring Suite -> eG VM Agent -> Stop Agent.
   - Then, follow the menu sequence, Start -> Programs -> eG Monitoring Suite -> eG VM Agent -> Uninstall Agent, to begin uninstalling the VM agent.
   - Figure 6.1 then appears. Select the **Remove** option from Figure 6.1 and click the **Next** button therein to uninstall the eG VM agent.

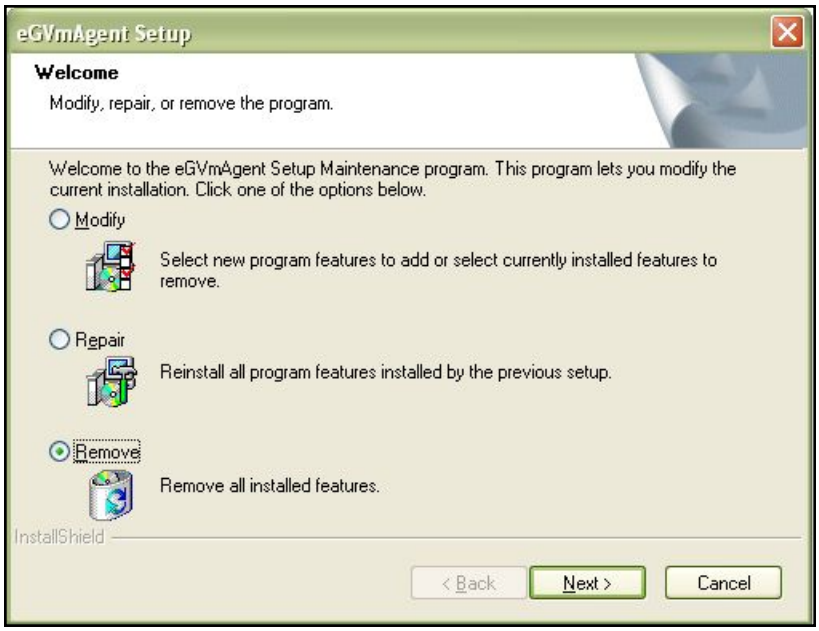


Figure 6.1: Selecting the Remove option

- Click the **OK** button in the message box of Figure 6.2 to resume the uninstallation.



Figure 6.2: Confirming the removal of the eG VM agent and its components

- Upon successful uninstallation, setup automatically exits.
- Repeat this procedure on every Windows VM on the ESX server.

5. On the other hand, if the remote Windows agent has obtained the *inside view* of the Windows VM **without using the eG VM Agent** - i.e., by remotely connecting to every Windows VM via NetBIOS - then, follow the steps given below to ensure that the Windows VM is no longer monitored:
   - The first step is to remove the eGRemote Execution service that the remote Windows agent creates on each of the Windows VMs it monitors - this service enables the remote agent to execute certain key scripts/executables on the Windows VM for collecting the *inside view* metrics. To uninstall this service, an executable named **ServiceInstaller.exe** should be made available on the Windows VM. Typically, every eG agent deployed on a Windows host is bundled with a **ServiceInstaller.exe**. Therefore, login to the Windows host on which the remote agent operates, and copy the **ServiceInstaller.exe** from the **<EG_AGENT_INSTALL_DIR>\bin** directory on the agent host to any location on the Windows VM.
   - Next, login to the Windows VM.
   - Go to the command prompt of the VM, switch to the directory that contains the

ServiceInstaller.exe, and run the following command:

ServiceInstaller.exe uninstall eGRemoteExec

This command uninstalls the eGRemote Executionservice.

- Then, delete the **eGRExecSvc.exe** in the **C:\Windows** directory of the VM.
- Finally, delete the **C:\Windows\eg** folder on the VM.

6.2 How to Stop the "Agentless Monitoring" of all the VMs on an ESX server?

If you want to stop monitoring all the VMs on an ESX server, then follow the instructions given hereunder:

1. Follow the procedure discussed in Section **6.1** on each of the VMs on the target ESX server. This will disable the remote agent's capability to extract the *inside view* metrics from the VMs.
2. To ensure that the remote Windows agent does not even attempt to collect the *inside view* metrics from the VMs, you will have to disable all the *inside view* tests that are executed by the eG agent. For this, first, login to the eG manager as *admin* with password *admin*.
3. Follow the Agents -> Tests -> Enable/Disable menu sequence in the eG administrative interface. Figure 6.3 then appears. Select *VMware vSphere ESX* (or *VMware vSphere VDI*) from the **Component type** list of Figure 6.3 and pick **Performance** as the **Test type**. All tests currently enabled for the chosen component type will then appear in the **ENABLED TESTS** list.

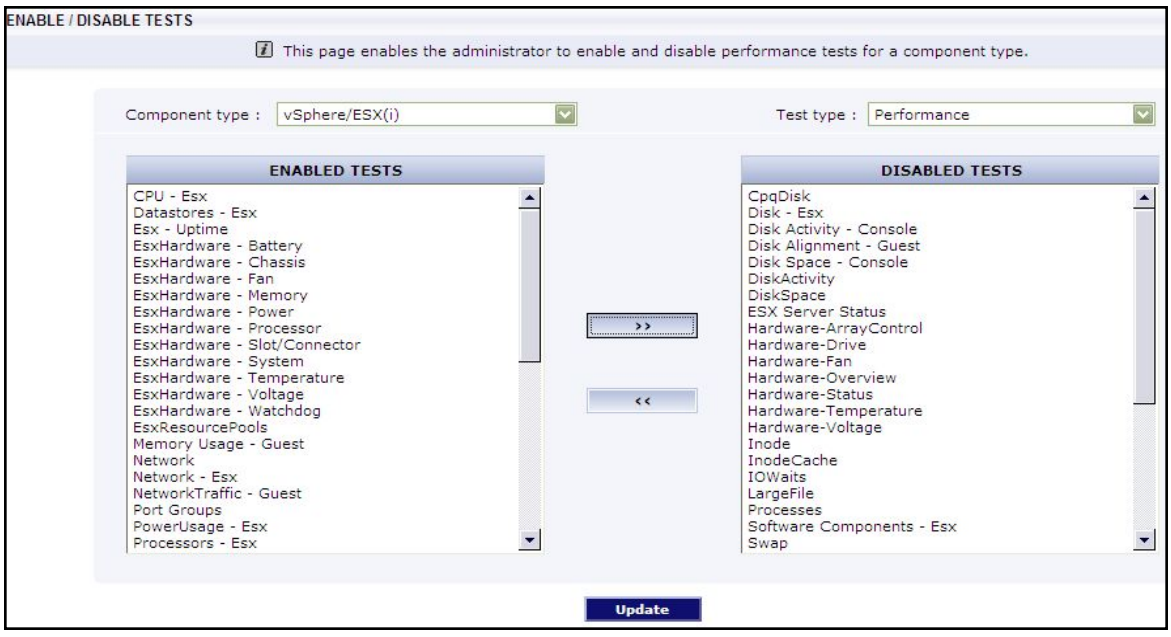


Figure 6.3: Disabling VM-related tests

4. Disable the VM-related tests by selecting them from the **ENABLED TESTS** list, and clicking the **>>** button. This will transfer the selection to the **DISABLED TESTS** list. The tests to be disabled are: Esx VM Details test, EsxResourcePools test, DiskActivity - Guest test, DiskSpace - Guest test, Uptime -

Guest test, System - Guest test, MemoryUsage - Guest test, TCP - Guest test, TcpTraffic - Guest test, NetworkTraffic - Guest test, WindowsMemory - Guest test, WindowsNetTraffic - Guest test, WindowsServices - Guest test, and HandlesUsage - Guest test.

5. Finally, click the **Update** button in Figure 6.3.
6. Log out of the eG administrative interface.

6.3 How to Stop the "Agentless Monitoring" of an ESX server and its VMs?

To achieve this, follow the steps given below:

1. First, unmanage the ESX server in question. For this, login to the eG administrative interface as *admin* with password *admin*.
2. Follow the menu sequence: Infrastructure -> Manage/Unmanage -> Server.
3. Select *VMware vSphere ESX* (or *VMware vSphere VDI*), depending upon how you have managed the target ESX server), as the **Component type** from Figure 6.4 that appears.

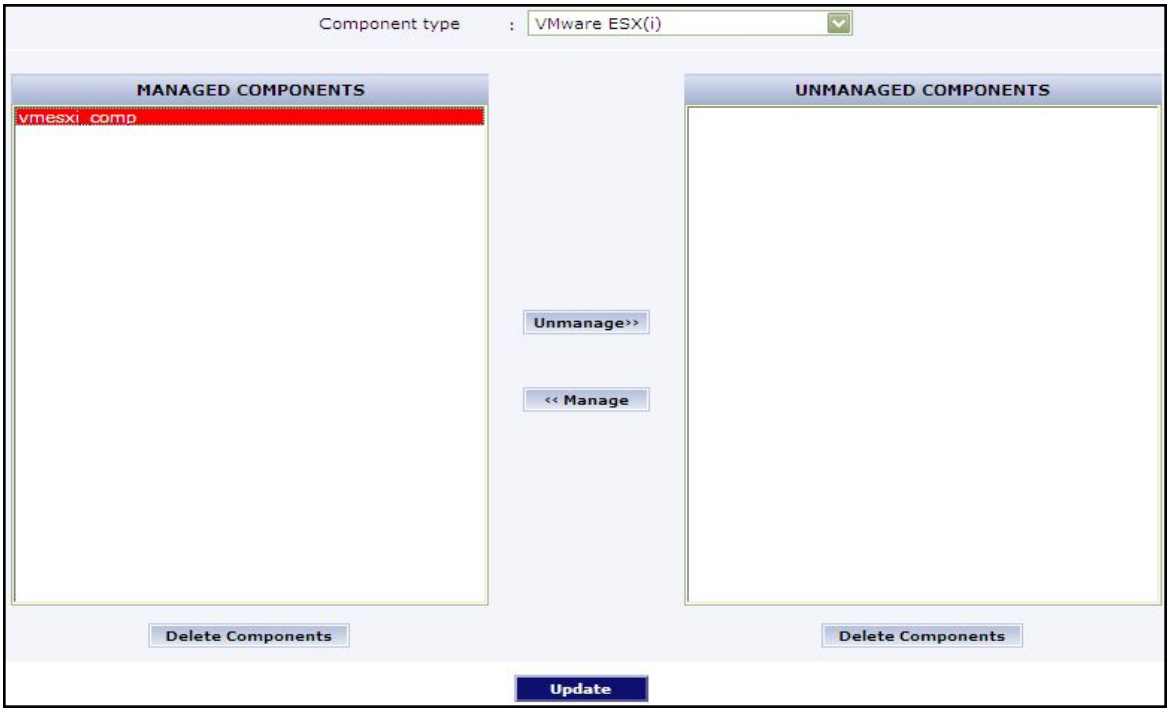


Figure 6.4: Selecting the ESX server to be unmanaged

4. Pick the ESX server to be unmanaged from the **MANAGED COMPONENTS** list of Figure 6.4.
5. Then, click the **Unmanage** button in Figure 6.5, so that the chosen component moves to the **UNMANAGED COMPONENTS** list (see Figure 6.5).

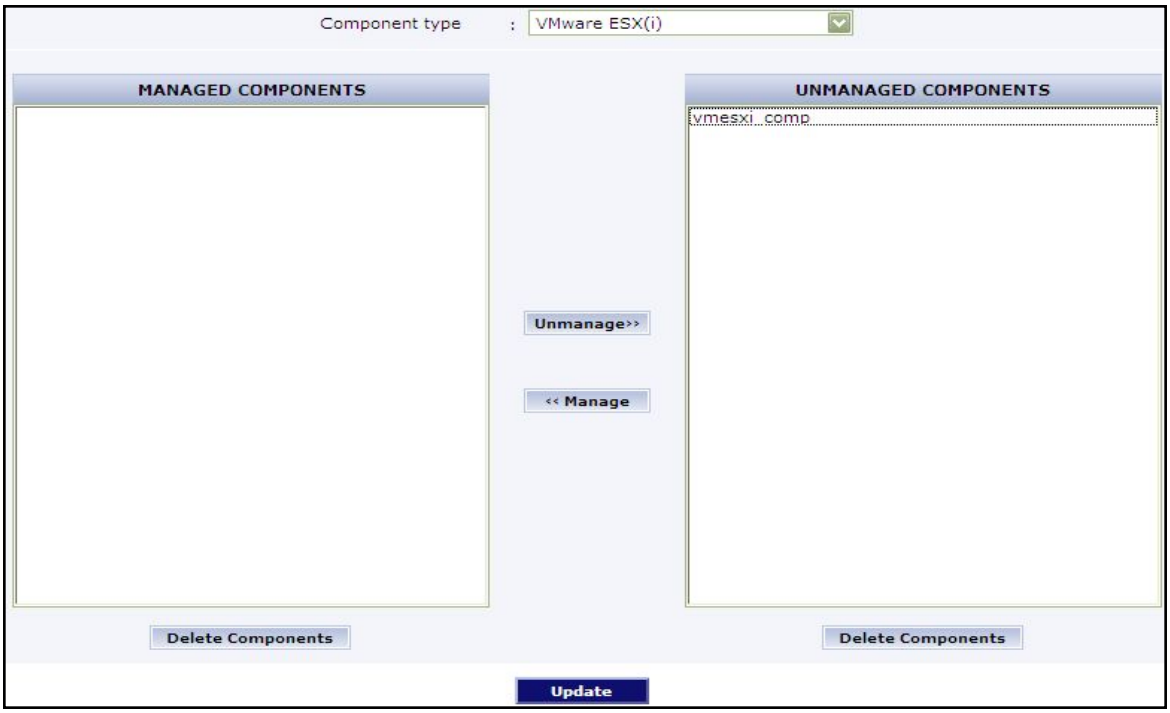


Figure 6.5: Unmanaging the ESX server

6. Finally, click the **Update** button in Figure 6.5.
7. Log out of the eG administrative interface.
8. Then, follow the steps discussed in Section **6.2** on all the VMs on the target ESX server to disable the *inside view* for the VMs.

7

The VM Dashboard

In large virtualized infrastructures today, a number of business-critical applications are being deployed on tens of virtual machines configured on a variety of virtualization platforms (e.g., VMware ESX, Citrix XenServer, Solaris LDoms, etc.), so as to optimize space and resource usage. In such environments, excessive resource usage by a single VM or a resource pool on a virtual host can cause a huge dent in the resources available to other VMs, thereby affecting the performance of the applications executing on those VMs! To ensure a high uptime for their key applications, administrators need to track in real-time the resource usage across VMs and physical servers regardless of the underlying virtualization platform, quickly detect abnormal usage patterns, accurately identify the VM(s) responsible for the same, and promptly initiate corrective action. Likewise, in environments where centralized management tools such as VMware vCenter are used, administrators also need to keep tabs on the availability and overall health of these tools, so as to ensure that performance degradations that the tool experiences does not impact the performance of the ESX servers it manages.

eG Enterprise provides a single, central **VM Dashboard** that provides an integrated interface from where administrators can compare resource usage across physical servers and VMs on each physical server, and provides them with real-time insights into the health of the physical servers, the status of the VMs, and how each VM currently uses the allocated and physical resources available to it.

Using this dashboard, administrators can:

- Understand, from a single look, the composition of the managed virtualized environment - in other words, using the dashboard, you can easily view and instantly figure out the complete hierarchical structure of your virtualized environment; for instance, for a VMware environment, the dashboard will help you identify the number of vCenter servers that are being managed, the number of datacenters configured on each vCenter, the clusters (if any) that exist within the datacenters, the ESX servers being managed by every datacenter, the resource pools on each ESX server, and the VMs that each resource pool comprises of.
- Detect at a glance, excessive resource usage by any VM, cluster, resource pool, or a physical server from across the environment, regardless of the virtualization technology in use; quickly diagnose the root-cause of the resource drain with the help of efficient drill down features;
- Accurately identify resource-intensive VMs/resource pools/clusters/physical servers;
- Promptly detect unavailable datastores;
- Instantly spot a powered-off VM anywhere in the environment;
- Know which VM is currently operating on which physical server, and thus keep tabs on VMotion/XenMotion (as the case may be) activities;
- View a consolidated list of issues currently encountered by physical servers and virtual machines across the environment and also per virtual component, so as to ease the troubleshooting efforts of a dedicated help-desk;
- Quickly identify and troubleshoot issues with vCenter servers (if any) in the environment;

To access the **VM Dashboard**, follow the menu sequence: Components -> Virtual -> Dashboard in the eG monitoring console. Figure 7.1 then appears:

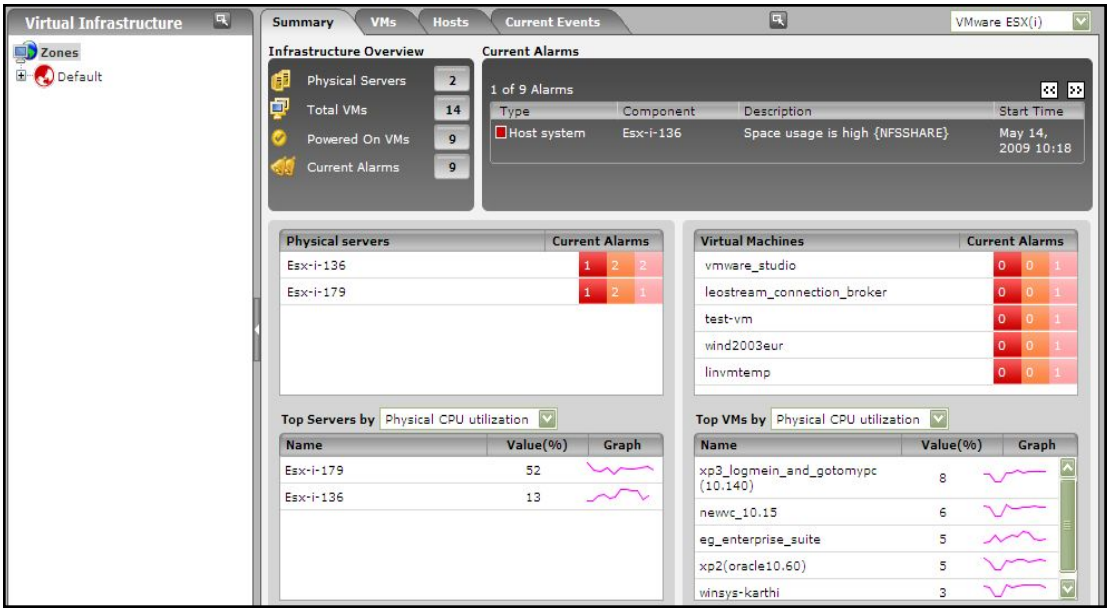


Figure 7.1: The VM Dashboard

The **VM Dashboard** of Figure 7.1 comprises of two panels. The left panel boasts of a tree-structure, comprising of a default global node named **Zones**. All the zones/farms in the target environment that have been configured with one/more virtual hosts (eg., VMware ESX servers, Citrix XenServers, etc.) or VMware vCenter servers, will be the sub-nodes of the **Zones** node. If you expand a particular zone node in the tree, you will find that the virtual component-types that have been added to that zone appear as its sub-nodes. If you expand any node under the zone node that corresponds to a particular virtual server-type (eg., *VMware vSphere ESX*, *VMware vSphere VDI*, *Citrix XenServer*, etc.), then such a tree will typically comprise of virtual hosts of that type that are included in the zone. Expanding the virtual host node will reveal the VMs executing on that host and resource pools (if any) configured on that host.

In addition to sub-nodes representing a virtual server-type, if a *VMware vCenter* component has been added to the zone, then a *VMware vCenter* sub-node will also appear under the corresponding zone node. If you expand the node representing the *VMware vCenter* component-type under a zone, then all the managed vCenter servers will appear as its sub-nodes. Expanding a particular vCenter server node will reveal the folders (if any) configured on that vCenter server; similarly, expanding a particular folder node will reveal the datacenters (if any) that are being managed by that vCenter server. To view the virtual hosts/clusters configured within a datacenter, you will have to expand the corresponding datacenter node. While the cluster tree will contain virtual hosts within the cluster as its sub-nodes, expanding a virtual host tree will reveal the VMs and resource pools that are executing on that virtual host. You can also view the virtual host tree by expanding any node that corresponds to a virtual host type under a particular zone node. The state of a node in the tree depends upon the current state of its sub-nodes. Independent virtual hosts/vCenter servers that are not part of any existing zone will be automatically grouped under the **Default** zone tree (see Figure 7.2).

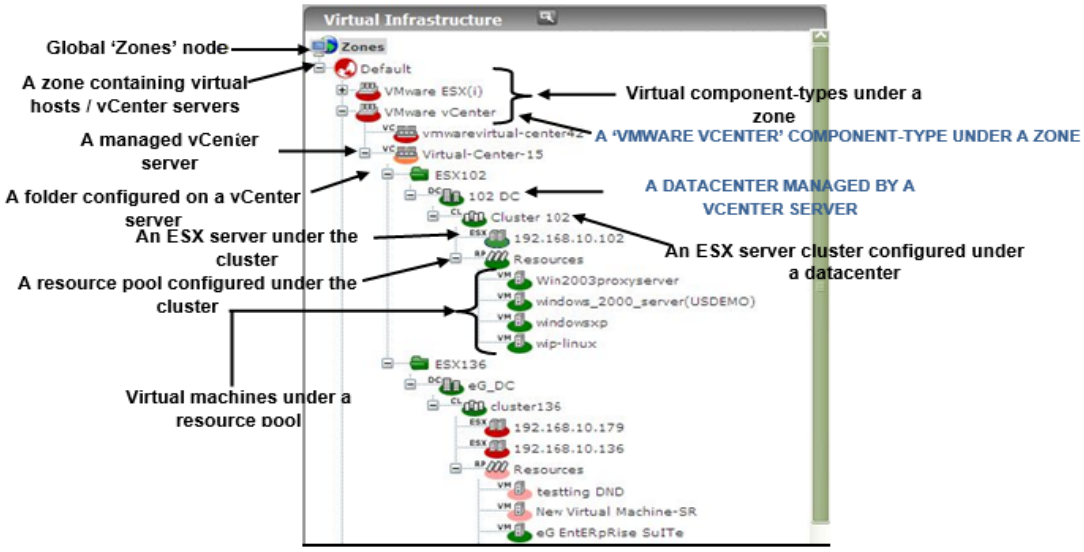


Figure 7.2: The Virtual Infrastructure tree

By default, the tree-structure lists all virtual hosts, clusters, and virtual machines. Accordingly, when you click on the button at the right, top corner of the left panel, both the **Hosts And Clusters** and **Show Virtual Machines** options will be chosen by default from the **Display Settings** section (see Figure 7.3) that appear. To view only the VMs in the tree, select the **Show Virtual Machines** check box, but deselect the **Hosts And Clusters** check box. To hide VMs from the tree-view, select the **Hosts And Clusters** check box, but leave the **Show Virtual Machines** check box deselected.



Figure 7.3: Display settings for the tree-view

The right panel (see Figure 7.1) is a context-sensitive panel, the contents of which will vary according to the node chosen from the left panel. By default, this panel provides 4 tab pages - a **Summary** tab page that provides a quick summary of performance and problem information pertaining to the node chosen from the tree, a **VMs** tab page that provides current status updates related to virtual machines, a **Hosts** tab page that displays virtual host-specific metrics gathered in real-time, and a **Current Events** tab page that lists the problems currently experienced by virtual hosts and guests. By default, all the tab pages provide information pertaining to managed *VMware vSphere ESX servers* in the environment. Accordingly, the **VMware vSphere ESX** option is chosen from the drop-down list in the top, right corner of the right panel (indicated by Figure 7.1). You can view details for a different virtual component type, by selecting a different option from this drop-down list.

Note:

As stated earlier, by default, the tab pages in the right panel provide details related to *VMware vSphere ESX* servers. This default setting can be overridden in the following manner:

- Click on the button at the right, top corner of the right panel. Figure 7.4 then appears.

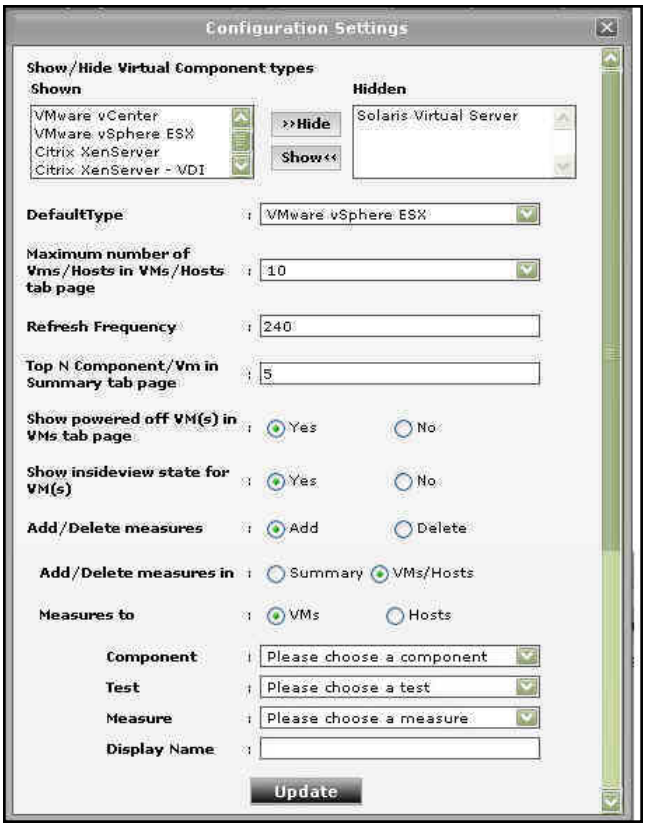


Figure 7.4: The Configuration Settings page

- By default, the *VMware vSphere ESX* option is chosen from the **DefaultType** list in Figure 7.4, indicating that the details in the right panel pertain to the managed *VMware vSphere ESX* components by default. To change the default setting, you will have to select a different option from this list.
- Then, click the **Update** button in Figure 7.4.

Similarly, every tab page, by default, displays the details of the top 10 VMs/virtual hosts only. This is indicated by the default value **10** chosen from the **Limit to** list at the right, top corner of each tab page. To view more (or less) number of records in the tab pages, you will have to select a different value from the **Limit to** list. Alternatively, you can even override the default value **10**, so that the tab pages display more or a less number of records by default. To achieve this, click the button in the right panel, and set the **Maximum number of VMs/Hosts in VMs/Hosts tab page** parameter in Figure 7.4 that appears next to any value other than 10.

The sections to come take you on a guided tour of each of the tab pages.

7.1 The Summary Tab Page

The **Summary** tab page serves as a single, central interface that combines 'problem and performance information' related to virtual infrastructures. Using this tab page, administrators can perform the following with ease:

- Oversee, by a mere glance, the composition of and the all-round performance of the virtual infrastructure as a whole or of the virtual infrastructure component chosen from the tree;
- View a consolidated list of current alarms pertaining to the node chosen from the tree, and instantly identify problem-prone virtual infrastructure elements;
- Receive real-time updates of the resource usage of physical servers and virtual machines, and instantly identify the hosts/guests experiencing a resource contention;
- Easily analyze and accurately detect disconcerting trends in the resource usage of the physical servers and virtual machines.

This section discusses elaborately discusses the contents and usage of this tab page.

1. As soon as you open the **Virtual Dashboard**, the **Summary** tab page in the right panel will, by default, provide a quick overview of the composition and performance of the monitored *VMware vSphere ESX* servers spread across all the managed zones in the environment (see Figure 7.5). To view the performance summary of a different virtualization platform, select a different option from the drop-down list in the right, top corner of the **Summary** tab page. By default, *VMware vSphere ESX* is chosen from this list.

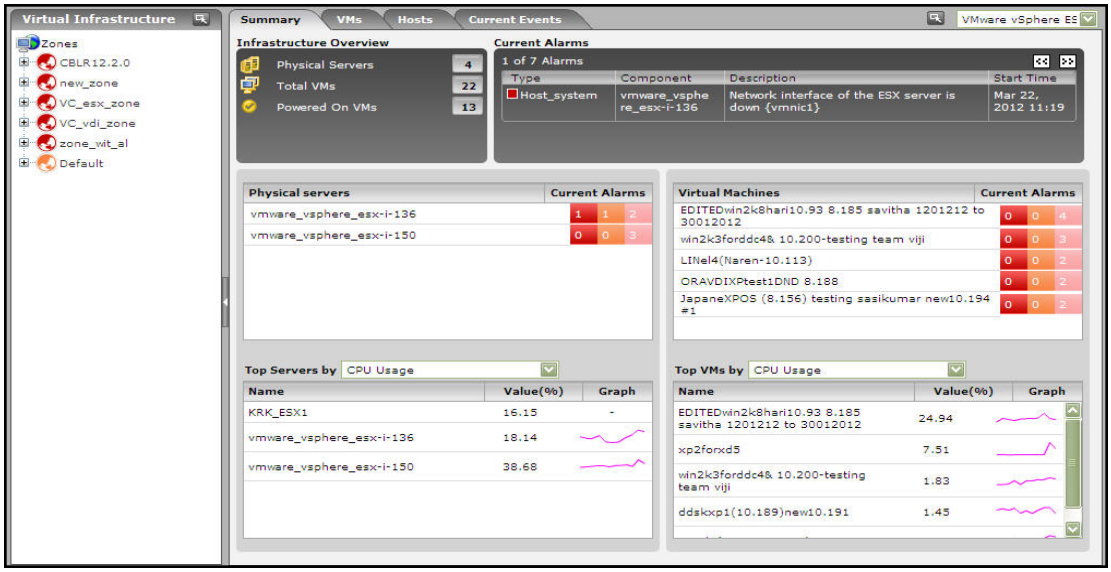


Figure 7.5: The Summary tab page if the global Zones node is clicked

2. For the default *VMware vSphere ESX* infrastructure, this tab page provides an **Infrastructure Overview** section that briefly discusses the key ingredients of the virtual infrastructure - i.e., the number of managed physical servers in the target virtual environment, the total number of VMs on the physical servers, the number of VMs that are currently powered on, and the number of alarms currently open on the infrastructure elements. For more details about the target environment, simply move your mouse pointer over every value displayed in the **Infrastructure Overview** section. For instance, to know the names of physical servers that are being managed in the environment, simply move the mouse pointer over the value corresponding to **Physical Servers** in the **Infrastructure Overview** section. The **Summary** tab page will then change to display a pop-up that lists the names of the physical *VMware vSphere ESX*

servers managed in the environment (see Figure 7.6). This way, you can view the names of virtual machines executing on the physical servers, the names of powered on VMs, and also the list of current alarms pertaining to the environment. Besides helping you identify VMs that are powered-off currently, the **Infrastructure Overview** also enables you determine the number and nature of the unresolved problems in the environment.

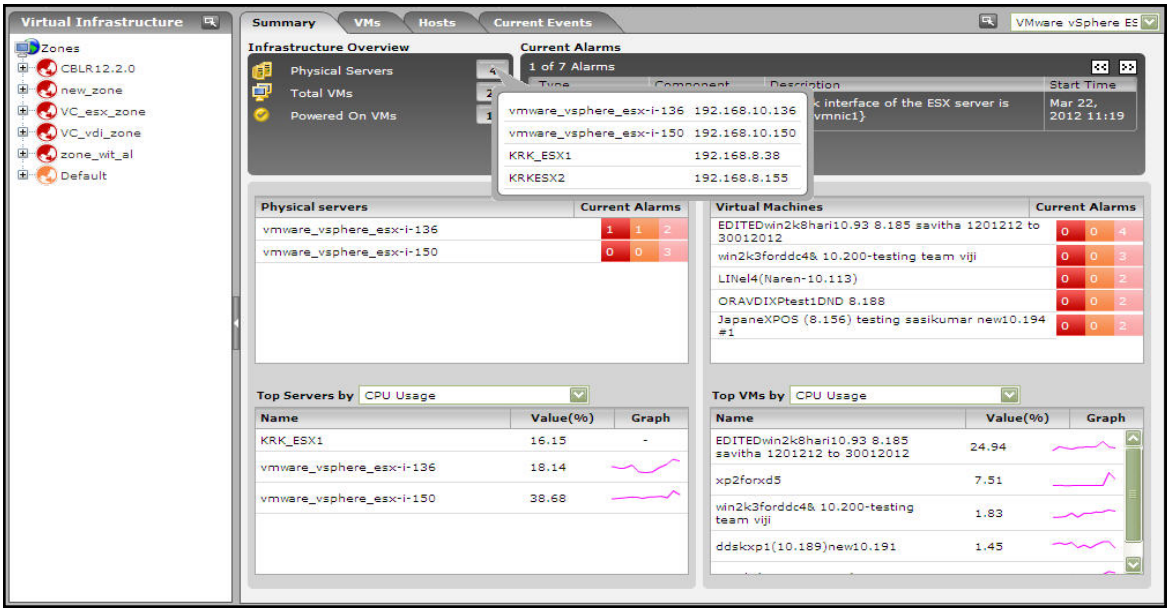


Figure 7.6: The names of physical vSphere/ESXi servers displayed in a pop-up

3. Now that you know the names of the physical servers, you might want to analyze the current resource usage of each of these servers to ascertain whether they are experiencing any resource shortages or not. For that, click on the **Physical Servers** label in the **Infrastructure Overview** section of Figure 7.6. This leads you straight to the **Hosts** tab page, which displays the physical servers managed across all zones configured in the environment, the current state of each server, and also the resource usage metrics pertaining to each server (see Figure 7.7).

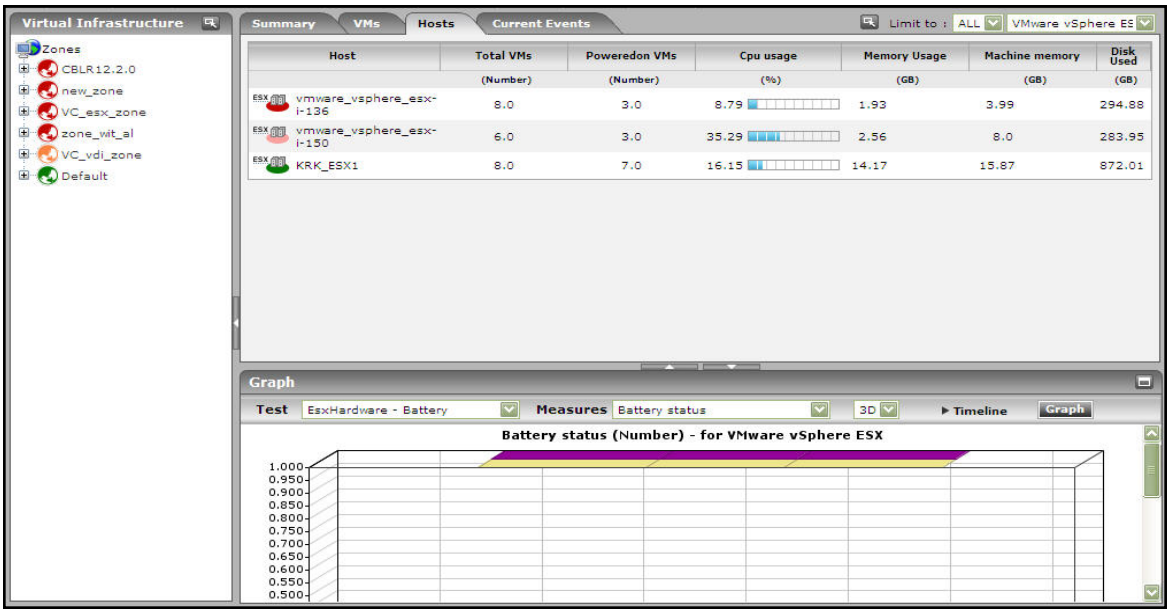


Figure 7.7: The Hosts tab page appears when the 'Physical servers' label in the Summary tab page is clicked

4. Similarly, you can click on the **Total VMs** label in the **Infrastructure Overview** section of Figure 7.6 to switch to the **VMs** tab page, focus on the performance of the individual VMs running on all the *VMware vSphere ESX* servers (by default) managed across the environment, and identify the VM that could be consuming resources excessively (see Figure 7.8).

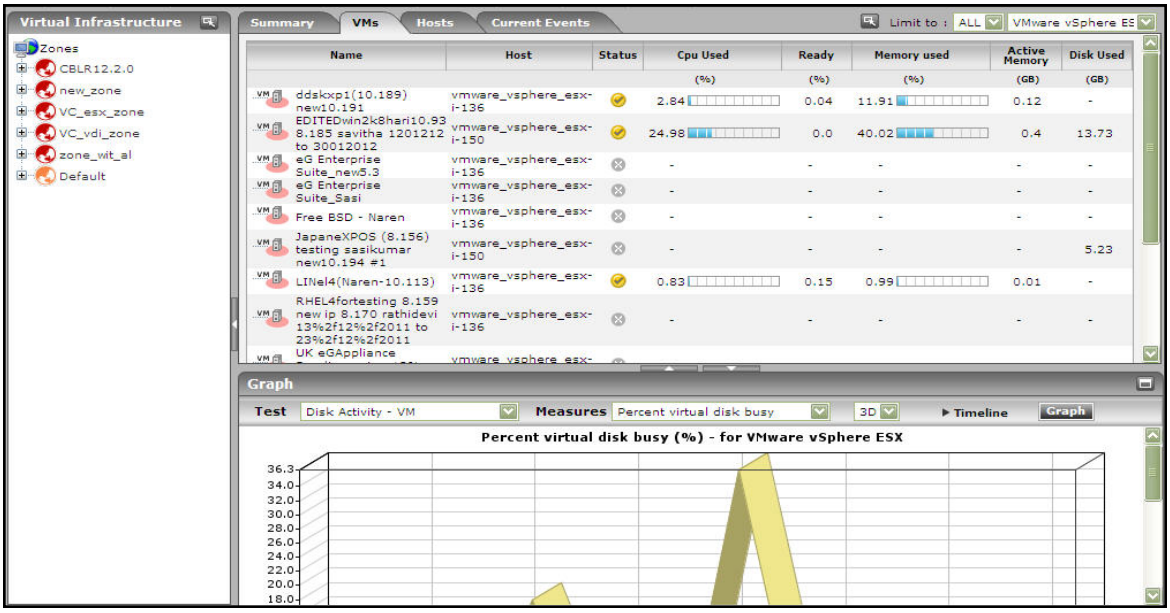


Figure 7.8: The VMs tab page that appears when the 'VMs' label in the Summary tab page is clicked

5. Similarly, clicking on the **Powered On VMs** label in the **Infrastructure Overview** section takes you to the **VM**s tab page and allows you to analyze the resource usage of powered-on VMs alone (see Figure 7.9).

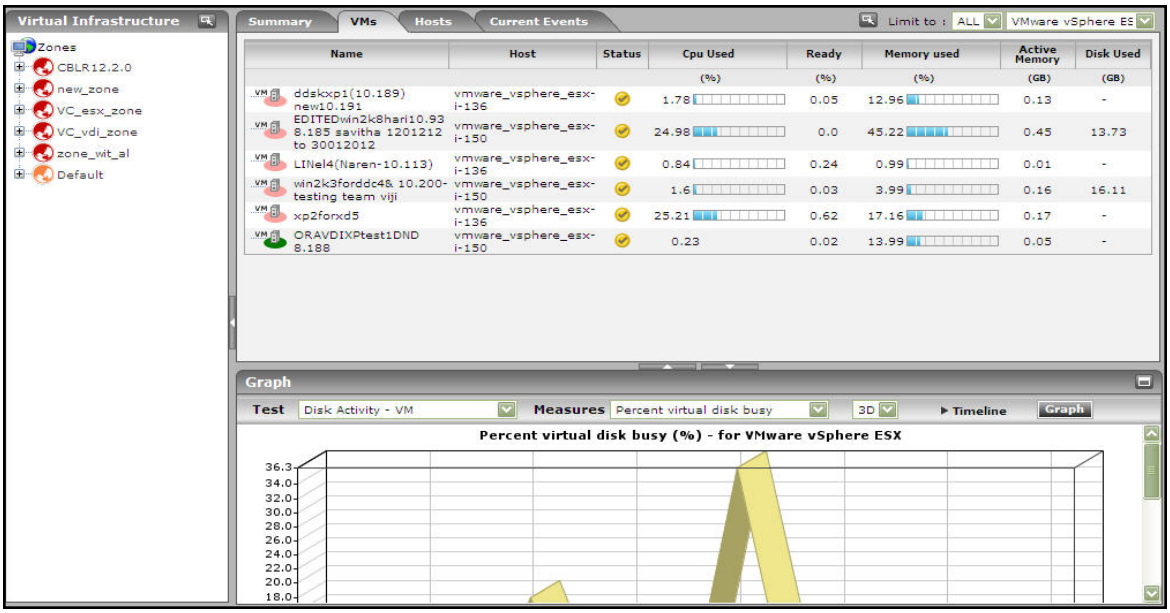


Figure 7.9: The VMs tab page listing only powered-on VMs

6. If you want to focus on each problem closely, then, you can use the **Current Alarms** section adjacent to the **Infrastructure Overview** section in the **Summary** tab page. Use the arrow buttons (<< and >>) above an alarm to navigate to the next alarm. Move your mouse pointer over an alarm to know which test has reported the problem.

7. The **Physical Servers** section below the **Infrastructure Overview** section enables you to determine how problem-prone the physical servers in your environment are, by revealing the number of critical, major, and minor issues that are currently unresolved for each physical server. By moving your mouse pointer over an alarm priority corresponding to a physical server, you can view the details of current alarms of that priority (see Figure 7.10).

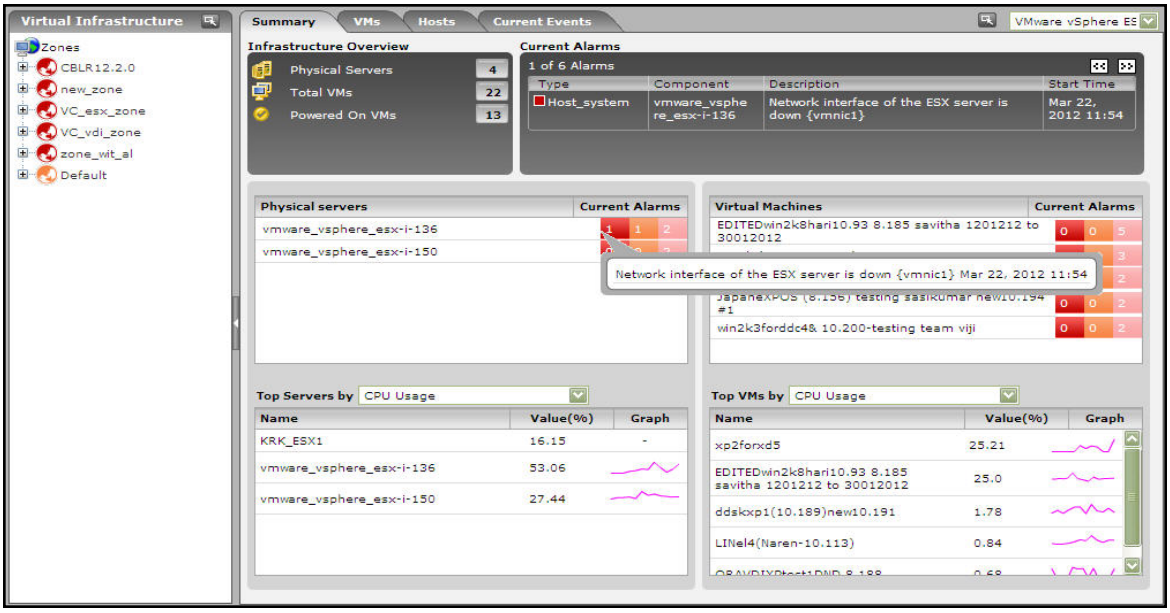


Figure 7.10: Moving the mouse pointer over an alarm priority that corresponds to a physical server

8. By clicking on a physical server in the **Physical Servers** section, you can zoom into the layer model, tests, and measurements of that server (see Figure 7.12). Besides indicating all the layers that have been affected by problems, Figure 7.12 will lead you to that layer-test-measure combination that is currently experiencing the most critical problem.

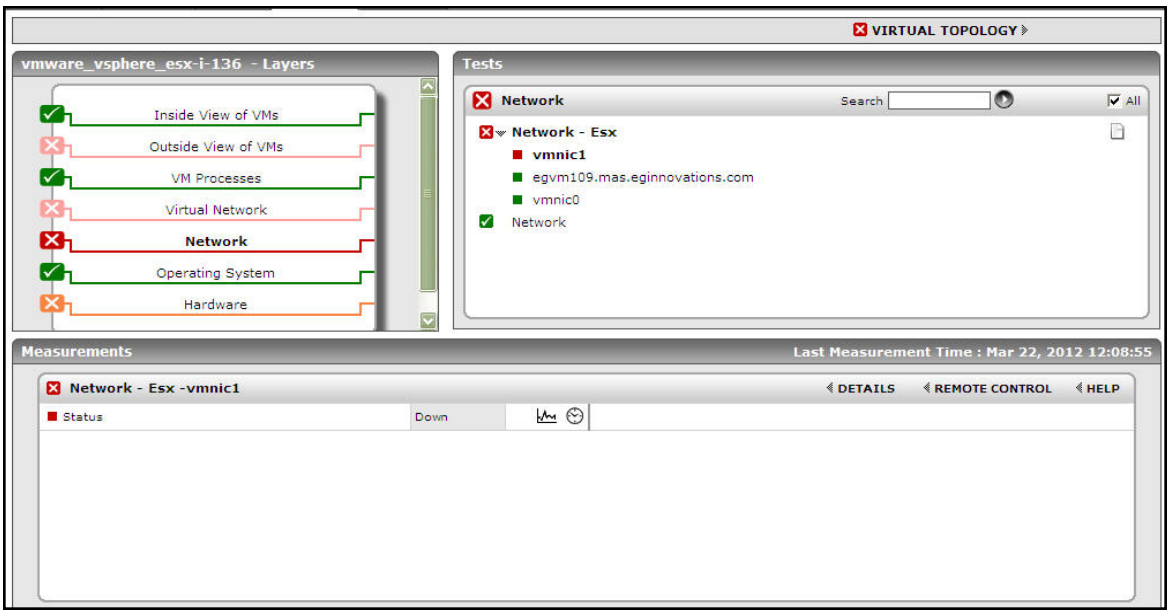


Figure 7.11: The layer model of a physical server indicating the layer where the root-cause of a problem lies

9. The **Virtual Machines** section enables you to determine how problem-prone the virtual machines in your environment are, by revealing the number of critical, major, and minor issues that currently remain unresolved for each virtual machine. By moving your mouse pointer over an alarm priority corresponding to a virtual machine, you can view the details of current alarms of that priority (see Figure 7.12).

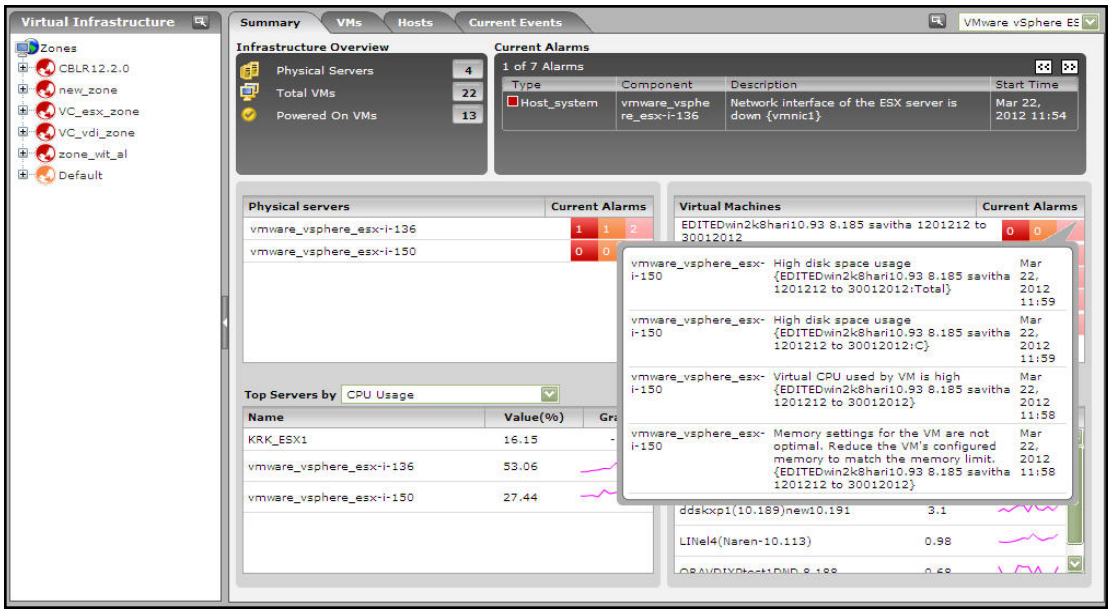


Figure 7.12: The alarms of a particular priority corresponding to a virtual machine

10. To know more about the problems affecting the internal performance of a VM, click on any virtual machine in the **Virtual Machines** section. Figure 7.13 will then appear indicating the same.

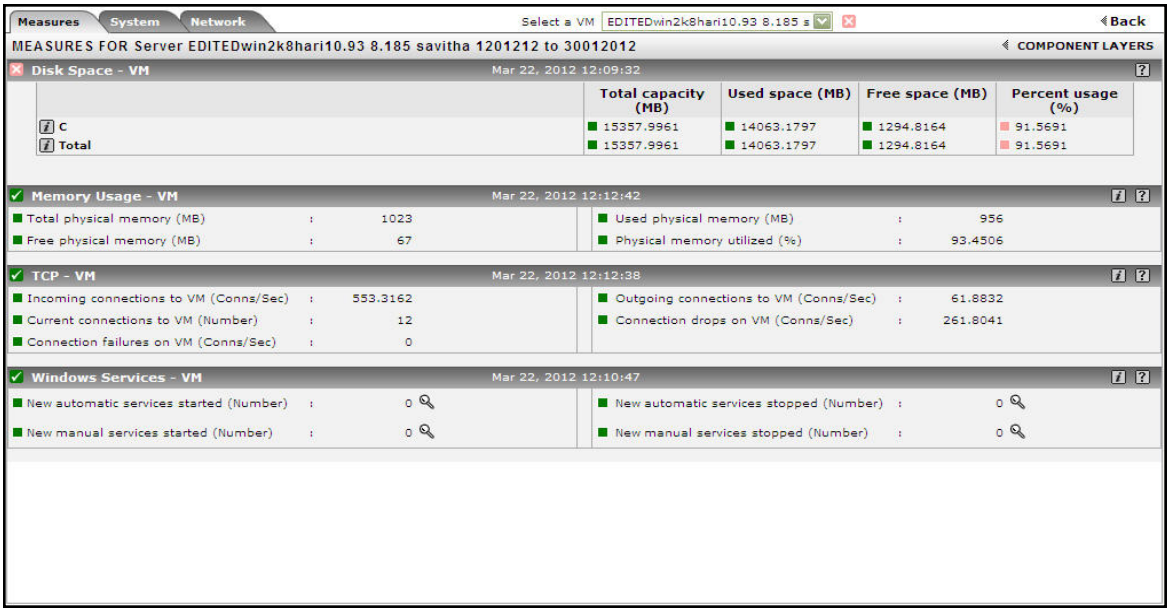


Figure 7.13: The 'inside view' of a VM clicked on in the Virtual Machines section of the Summary tab page

11. In large virtualized environments comprising of a multitude of virtual hosts that are configured with tens of VMs, it is often difficult for administrators to instantly and accurately locate resource-intensive physical servers and/or virtual machines. Similarly, identifying the physical servers with too many VMs is also a herculean task. In order to ease the pain of the administrators, the **Summary** tab page provides two sections - one each for physical servers and VMs - which can be configured to list the top physical servers

and VMs (respectively) in the environment, in terms of resource consumption. Also, by default, both sections will reveal the top consumers of physical CPU resources, starting with the leading consumer. Accordingly, the *Physical CPU utilization* measure is the default selection in both the **Top Servers by** and the **Top VMs by** lists. To view the top consumers of another resource, select a different measure from these lists.

12. In the **Top Servers by** list, in addition to the *Physical CPU utilization* measure, the following measures are available for selection by default: *Used physical memory*, *Used space*, *Registered guests*, and *VM power-on state*. If need be, you can override this default setting, so that new measures can be added to the list and one/more existing measures can be removed from the list. To do this, follow the steps given below:

    - Click on the button at the right, top corner of the **Summary** tab page.
    - Figure 7.14 then appears. To add a new measure to the **Top Servers by** list in the **Summary** tab page, first select the **Add** option from the **Add/Delete measures** section, and then pick the **Summary** option from the **Add/Delete measures in** section in Figure 7.14.

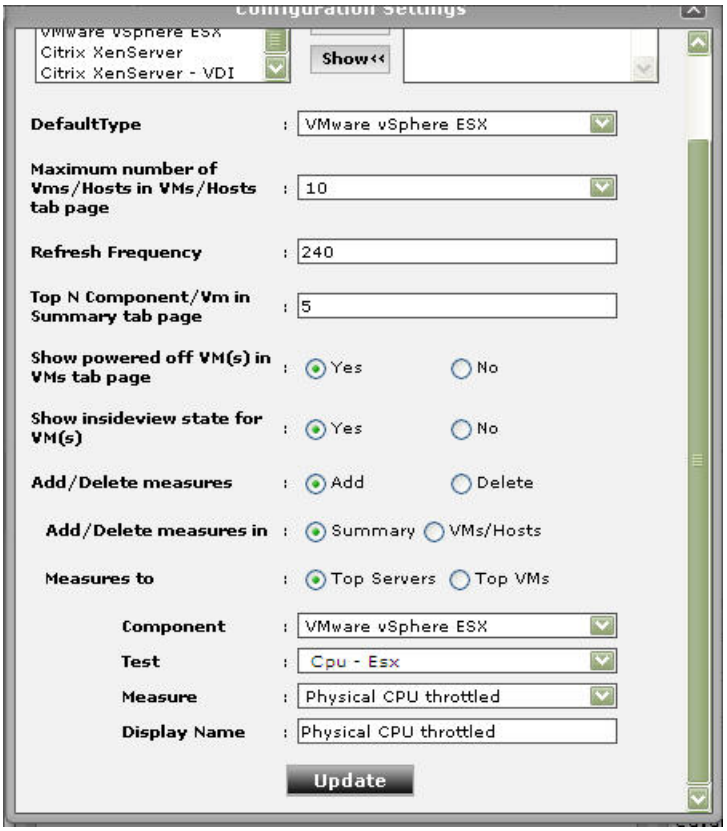


Figure 7.14: Adding a new measure to the Top Servers by list

- Next, set the **Measures to** flag to **Top Servers**.
- Then, pick the **Component** to which the new measure pertains.
- Select the **Test** that reports the measure of interest.
- Select the **Measure** to be added, and provide a **Display Name** for the measure.

- To add the measure, click the **Update** button. Doing so ensures that the **Display Name** specified in Figure 7.14 appears as an option in the **Top Servers by** list in the **Summary** tab page.
- Similarly, you can remove a measure from the **Top Servers by** list. For this purpose, set the **Add/Delete** flag to **Delete**, select the **Component** to which the measure to be deleted pertains, select the **Test** reporting the measure, select the **Measure** to be deleted, and finally, click the **Update** button.

13. In the **Top VMs by** list on the other hand, in addition to the *Physical CPU utilization* measure, the following measures are available for selection by default: *Memory usage*, *Disk capacity*, *and Percent disk usage*. If need be, you can override this default setting, so that new measures can be added to the list and one/more existing measures can be removed from the list. To do this, follow the steps given below:
    - Click on the button at the right, top corner of the **Summary** tab page.
    - Figure 7.15 then appears. To add a new measure to the **Top VMs by** list in the **Summary** tab page, first select the **Add** option from the **Add/Delete measures** section, and then pick the **Summary** option from the **Add/Delete measures in** section in Figure 7.15.

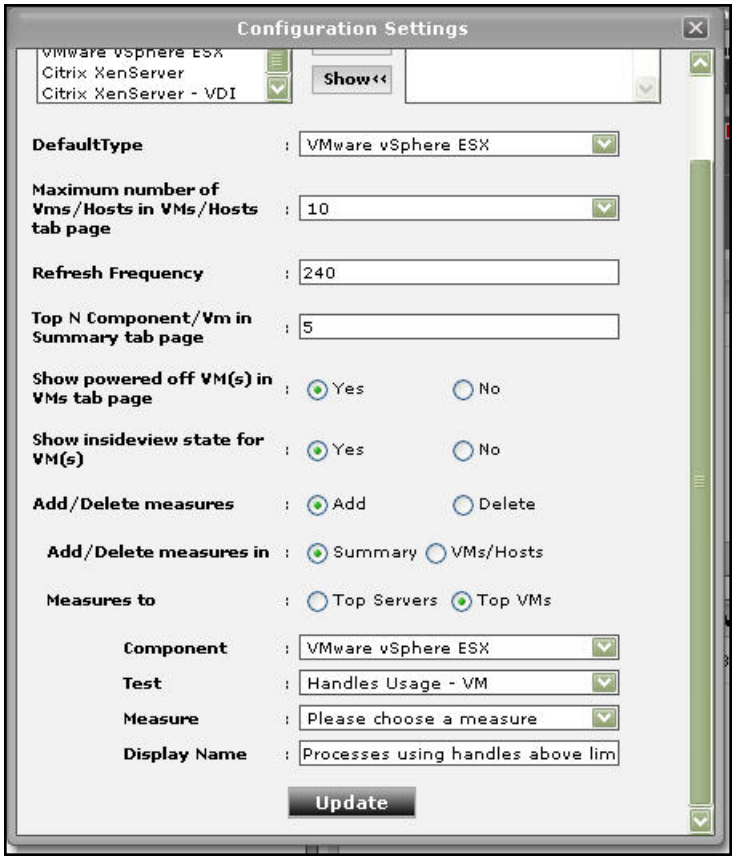


Figure 7.15: Adding a new measure to the Top VMs by list

- Next, set the **Measures to** flag to **Top VMs**.
- Then, pick the **Component** to which the new measure pertains.
- Select the **Test** that reports the measure of interest.

- Select the **Measure** to be added, and provide a **Display Name** for the measure.
- To add the measure, click the **Update** button. Doing so ensures that the **Display Name** specified in Figure 7.15 appears as an option in the **Top VMs by** list in the **Summary** tab page.
- Similarly, you can remove a measure from the **Top VMs by** list. For this purpose, set the **Add/Delete** flag to **Delete**, select the **Component** to which the measure to be deleted pertains, select the **Test** reporting the measure, select the **Measure** to be deleted, and finally, click the **Update** button.

14. Also, by default, both lists will display the top-5 resource consumers only. This default setting can be overridden by following the steps given below:
    - Click on the button at the right, top corner of the **Summary** tab page.
    - Figure 7.16 then appears. By default, the value **5** is displayed in the **Top N Component/VM in Summary tab page** text box, indicating that the **Summary** tab page displays the top-5 resource consumers, by default.

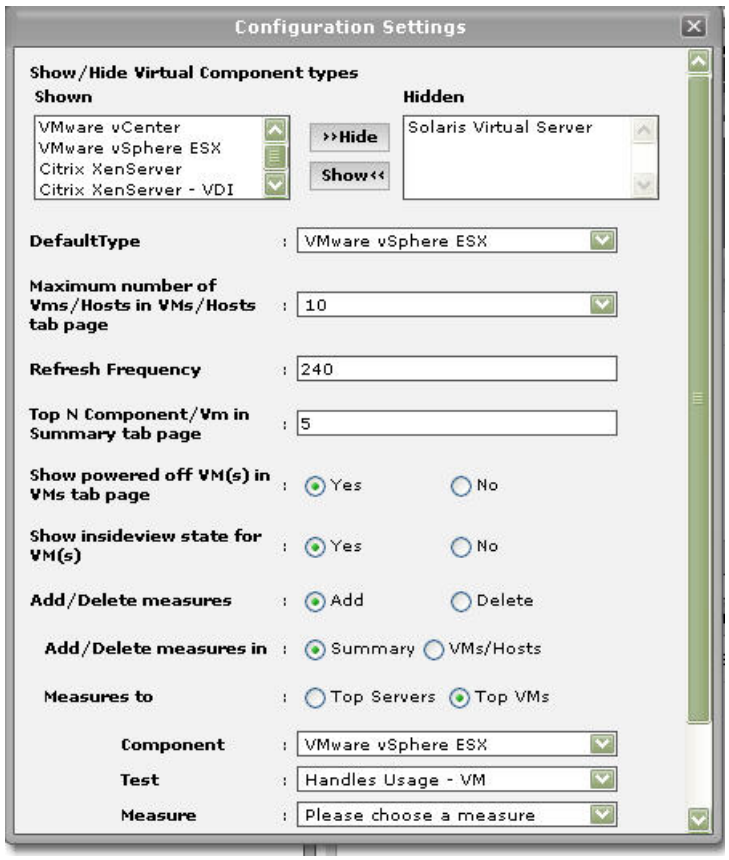


Figure 7.16: The Dashboard configuration page

- Override this default setting by specifying a different number in the **Top N Component/VM in Summary tab page** text box.
- Then, click the **Update** button in Figure 7.16.

15. Let us now focus on the **Top Servers by** section alone. Against every physical server displayed in the

Top Servers by section, the percentage of the chosen resource currently utilized by each physical server will be displayed, followed by a miniature graph tracking the usage of that resource over a period of time. If you click on a physical server in this section - say, the server that is the leading consumer of physical CPU resources - Figure 7.17 will appear revealing the layer model of that server. From the layer model, you can navigate to the test and the measure reporting the physical CPU usage of the server, perform further analysis, and accurately identify which processor supported by the server has contributed to the excessive resource usage.

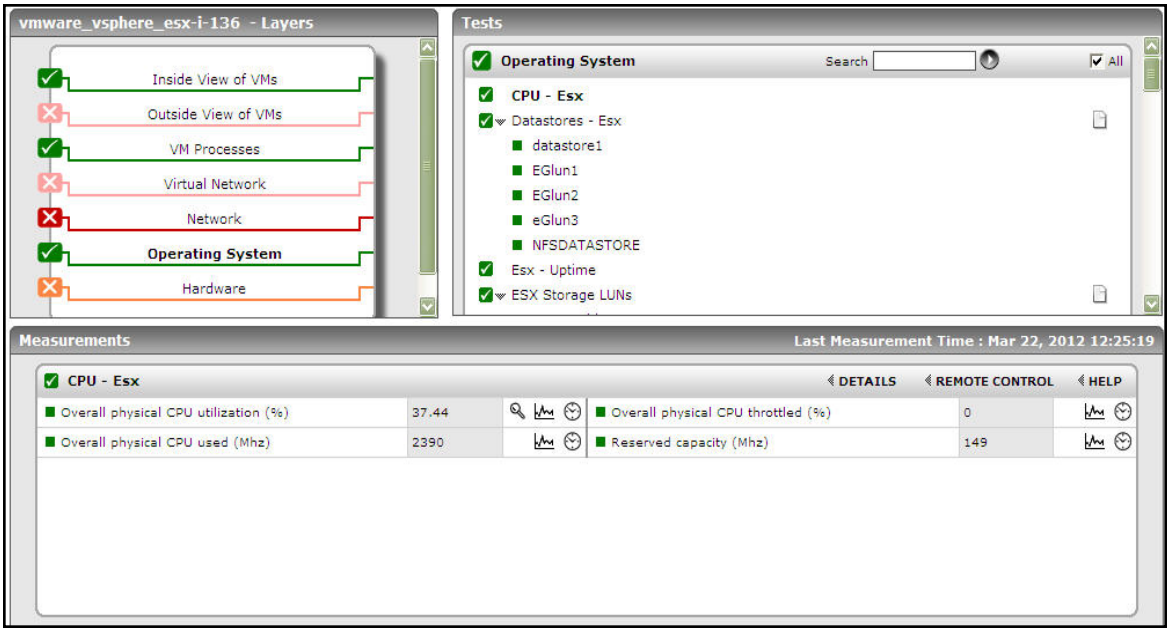


Figure 7.17: The layer model of a CPU-intensive physical server revealing the processor that is excessively consuming physical CPU

16. If you click on the miniature graph that corresponds to a physical server, the graph will expand as depicted by Figure 7.18. By default, the expanded graph tracks the variations in the measure selected from the **Top Servers by** list, during the last 1 hour. With the help of this graph, you can effortlessly observe how the physical server has been using the chosen resource over the last 1 hour by default. You can change this default period by choosing a different **Timeline** for the graph. This analysis will enable you to effectively study usage trends, and accurately detect the exact time at which the physical server began experiencing spikes in resource usage.

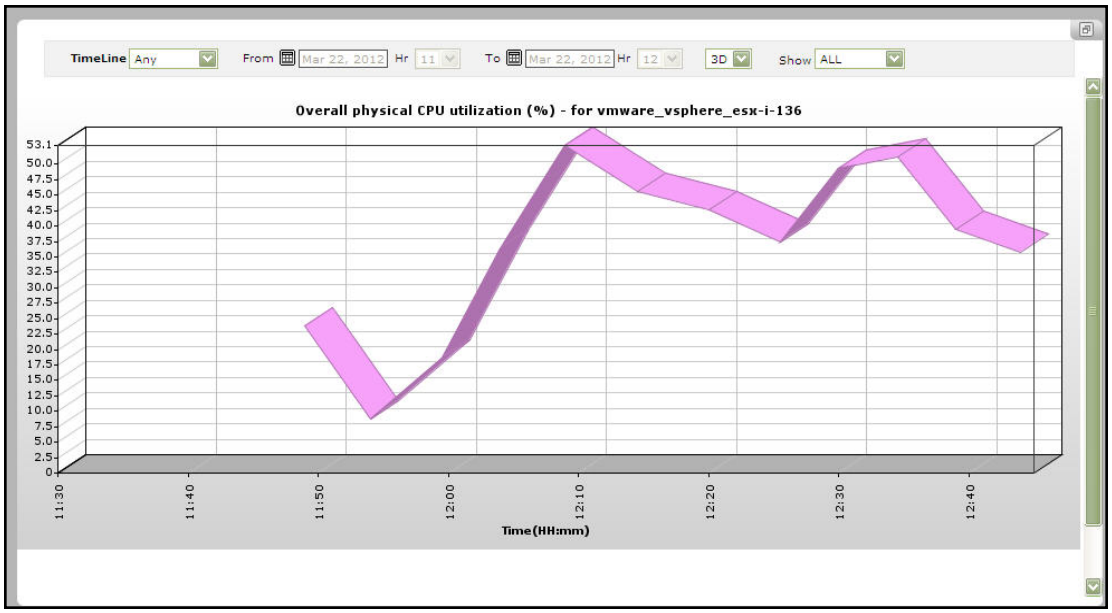


Figure 7.18: The expanded resource usage graph of a physical server

17. Let us now shift our focus to the **Top VMs by** section in the **Summary** tab page. Against every virtual machine displayed in the **Top VMs by** section, the percentage of the chosen resource currently utilized by each VM will be displayed, followed by a miniature graph tracking the usage of that resource over a period of time. If you click on a VM in this section - say, the VM that is the leading consumer of physical CPU resources - Figure 7.19 will appear revealing the layer model of the physical server on which the VM is executing. By default, the test that reports the physical CPU usage of the VM in question will be selected in the layer model, and all the measures reported by that test for the chosen VM will also be displayed. Using these metrics, you can effectively assess the overall resource usage of that VM.

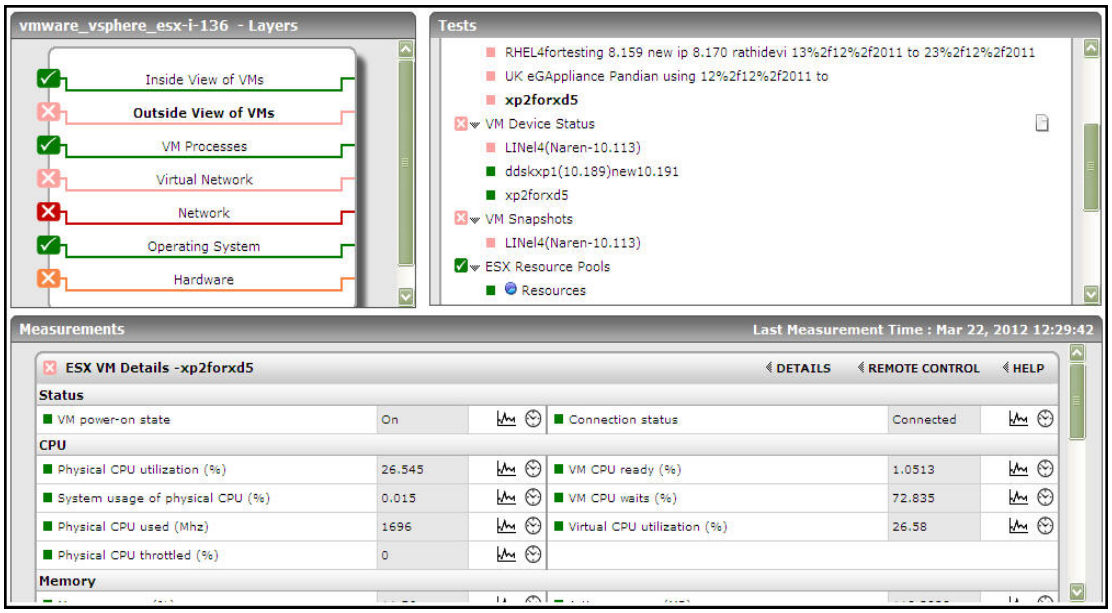


Figure 7.19: The layer model of a CPU-intensive VM revealing the resource usage of the VM

18. If you click on the miniature graph that corresponds to a VM in the **Top VMs by** section, the graph will expand as depicted by Figure 7.20. By default, the expanded graph tracks the variations in the measure selected from the **Top VMs by** list, during the last 1 hour. With the help of this graph, you can effortlessly observe how the VM has been using the chosen resource over the last 1 hour by default. You can change this default period by choosing a different **Timeline** for the graph. This analysis will enable you to effectively study usage trends, and accurately detect the exact time at which the VM began exhibiting unhealthy resource usage trends.

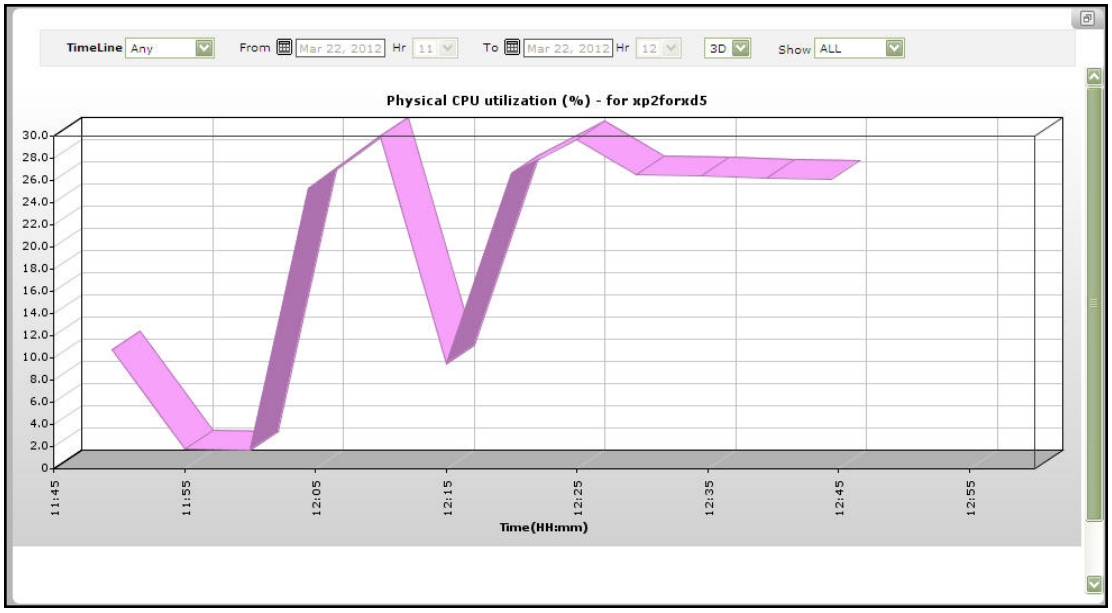


Figure 7.20: The expanded resource usage graph of a VM

19. If the node representing a particular zone is chosen from the tree-structure in the left panel, the contents of the **Summary** tab page will change accordingly. Besides revealing the number of physical servers, VMs, and powered-on VMs in the chosen zone, the **Infrastructure Overview** section also presents a macro view of the health of the zone by indicating the number of unresolved problems in the zone. The **Current Alarms** section will enable you to view these unresolved problems one after another. To know which physical servers in the zone are responsible for these problems, you can take the help of the **Physical Servers** section, which lists the names of the servers along with the number and severity (critical/major/minor) of problems (if any) each server is associated with. Similarly, the **Virtual Machines** section, in addition to displaying the names of VMs that are executing on the physical servers included in the zone, also reveals the problematic VMs by indicating the number and severity of problems (if any) that each VM is currently experiencing. Besides the above, the **Summary** tab page for a particular zone will also enable you accurately identify the resource-intensive physical servers and VMs in a zone. The **Top Servers by** section of this tab page displays the top-5 (by default) resource-hungry physical servers; this enables you to quickly identify the server in the zone that is most resource-intensive. The **Top VMs by** section displays the top-5 (by default) resource-hungry VMs, and thus enables you to identify the most resource-intensive VM in the zone.

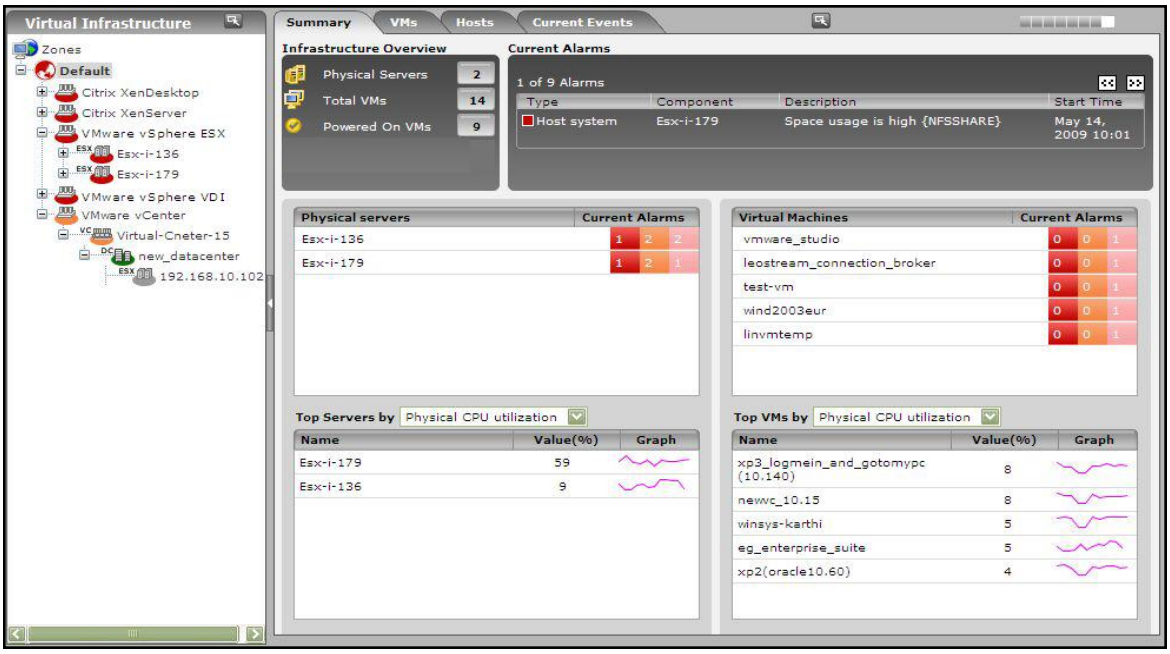


Figure 7.21: The Summary tab page if a particular zone is chosen from the tree-structure in the left panel

20. If a particular virtual server-type (eg., *VMware vSphere ESX*, *Citrix XenServer*, etc.) under a zone is chosen from the tree-structure, then the contents of the **Summary** tab page will change as depicted by Figure 7.22.

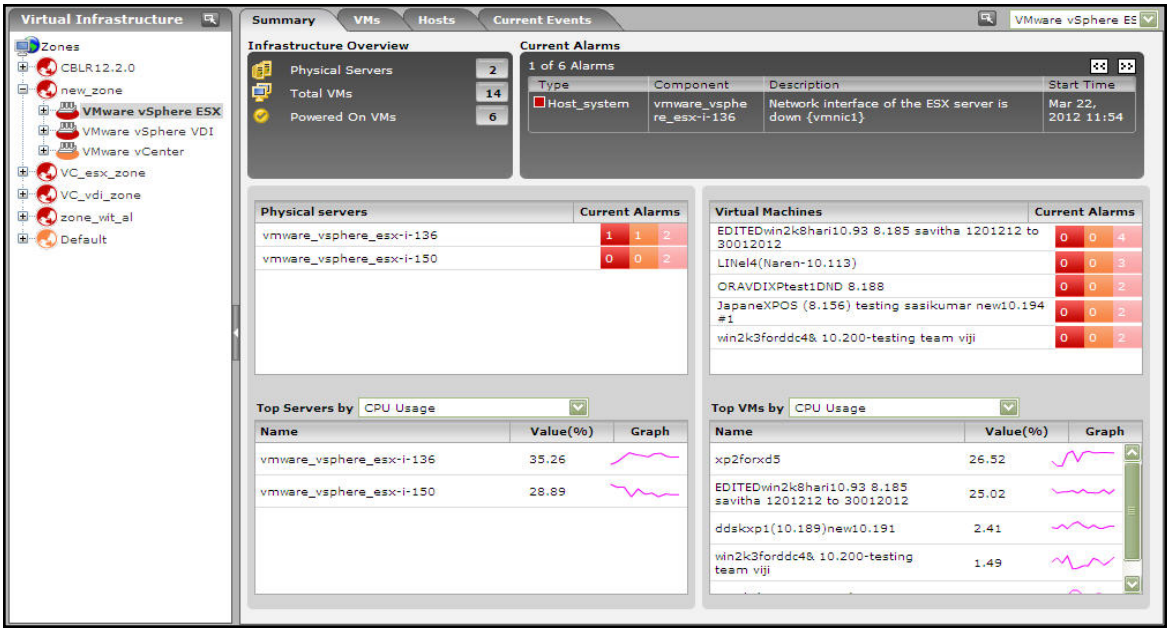


Figure 7.22: The Summary tab page of the node representing a virtual server-type

21. The **Infrastructure Overview** section in this case will provide a quick performance summary of only those servers in the zone that are of the type chosen from the tree-structure. In other words, if the *VMware vSphere ESX* node is chosen from the tree-structure, then the **Infrastructure Overview** section will

display the number of ESX servers in the zone, the number of VMs, and powered-on VMs executing on the ESX servers. You can move your mouse pointer over any number displayed in this section, to view the corresponding details.

22. To pay individual attention to each alarm, view them one after another in the **Current Alarms** section. You would now want to know which ESX servers in the zone are the most problematic, and also determine whether any of the VMs on ESX servers have contributed to these problems. For this purpose, you can use the **Physical Servers** and **Virtual Machines** sections, which list the problem virtual hosts and guests in the zone, and also indicate the number and type of problems currently encountered by each of the displayed hosts and guests. By moving your mouse pointer over an alarm priority corresponding to a physical server/VM, you can take a quick look at the alarms of that priority that are currently open on that physical server/VM.

23. Also, you can receive real-time updates on the resource utilization levels of these hosts and guests from the **Top Servers by** and **Top VMs by** sections; these sections, by default, display the top-5 physical servers and VMs in terms of *Physical CPU utilization*. To view the toppers in a different performance realm, you can select a different measure from the list box available in both the sections. You can even add a new measure to the list or remove one/more of the existing measures using the procedure discussed in page Section **7.1** and page Section **7.1** of this document. Similarly, to view more number of physical servers and VMs in this section, you can change the default value 5 to a different number using the procedure discussed in page Section **7.1** of this document. Using the information provided by these sections, you can determine the current resource usage of each displayed physical server and VM, and view graphs that can enable you to effectively assess the resource usage trends of these components over a broader period of time. Moreover, resource-intensive hosts and guests can be rapidly identified, sporadic/consistent surges in resource utilization by these hosts and guests can be promptly detected, and any potential resource contention can be diagnosed before its too late, and averted.

24. If a zone consists of a VMware vCenter server, then the node that corresponds to this zone in the tree-structure, when expanded, will also reveal a *VMware vCenter* sub-node. If this sub-node is clicked, the contents of the **Summary** tab page in the right panel will change as depicted by Figure 7.23.

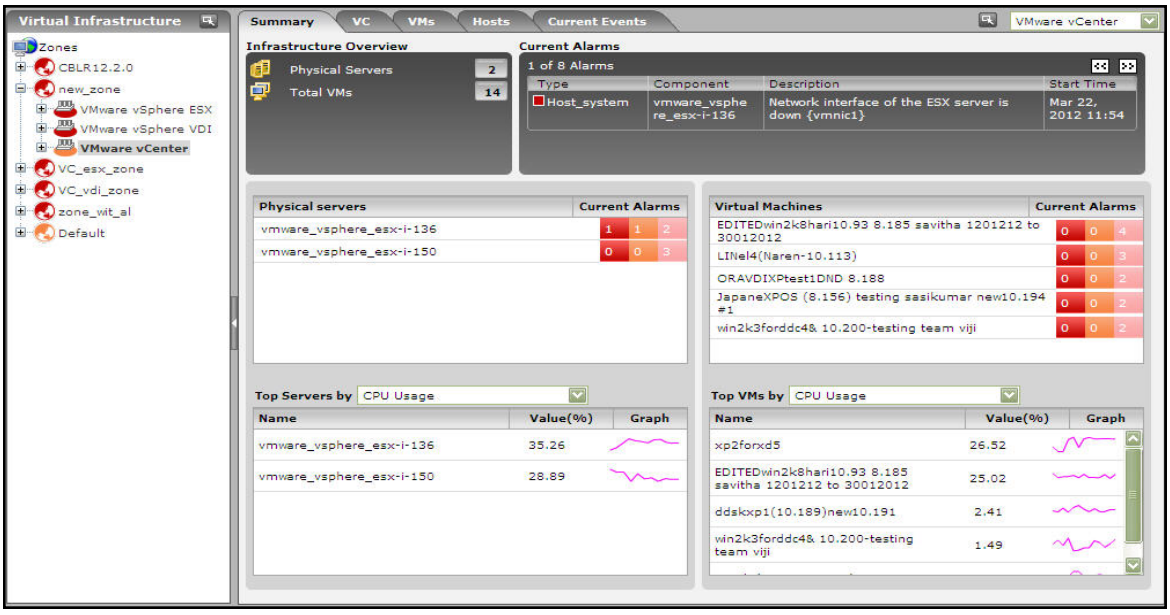


Figure 7.23: The Summary tab page if the 'VMware vCenter' node is clicked

25. The **Infrastructure Overview** section of Figure 7.23 reveals the number of physical servers and VMs managed by all vCenter servers in a zone. Move your mouse pointer over each of these numbers to know the names of the ESX servers and VMs, and also that of the vCenter server managing them. You can even focus on every performance issue experienced by these VMs and virtual servers individually by browsing the list of alarms in the **Current Alarms** section in the **Summary** tab page.

26. To know which physical sever and which VM has contributed the maximum to the problems list, use the **Physical Servers** and **Virtual Machines** sections; these sections display the problem-prone ESX servers and VMs (respectively) across vCenter servers in a zone, and indicate how many problems of what severity are currently affecting each of the ESX servers and VMs. Move your mouse pointer over a problem severity corresponding to an ESX server or VM to view the details of the related alarms.

27. Besides, you can quickly identify the most resource-hungry ESX servers and VMs across vCenter servers in a zone, using the **Top Servers by** and **Top VMs by** sections in the **Summary** tab page. By default, these sections display the top-5 physical servers and VMs in terms of *Physical CPU utilization*. To view the toppers in the usage of a different resource, you can select a different measure from the list box available in both the sections. You can even add a new measure to the list or remove one/more of the existing measures using the procedure discussed in **674** and page **675** of this document. Similarly, to view more number of physical servers and VMs in this section, you can change the default value 5 to a different number using the procedure discussed in page **676** of this document. Using the information provided by these sections, you can determine the current resource usage of each displayed physical server and VM, and thus identify the physical server or VM that is consuming resources excessively. Also, by clicking on the miniature graph alongside a physical server or VM, you can expand the graph and effectively analyze the ups and downs in resource usage of the corresponding physical server or VM over time. This way, you can accurately determine whether the increase in resource usage (if any) occurred suddenly, or whether an upward trend in resource usage began earlier on.

28. If you click on a particular vCenter server under the *VMware vCenter* node, then the resulting **Summary** tab page will provide an overview of the performance of that vCenter server alone (see Figure 7.24).

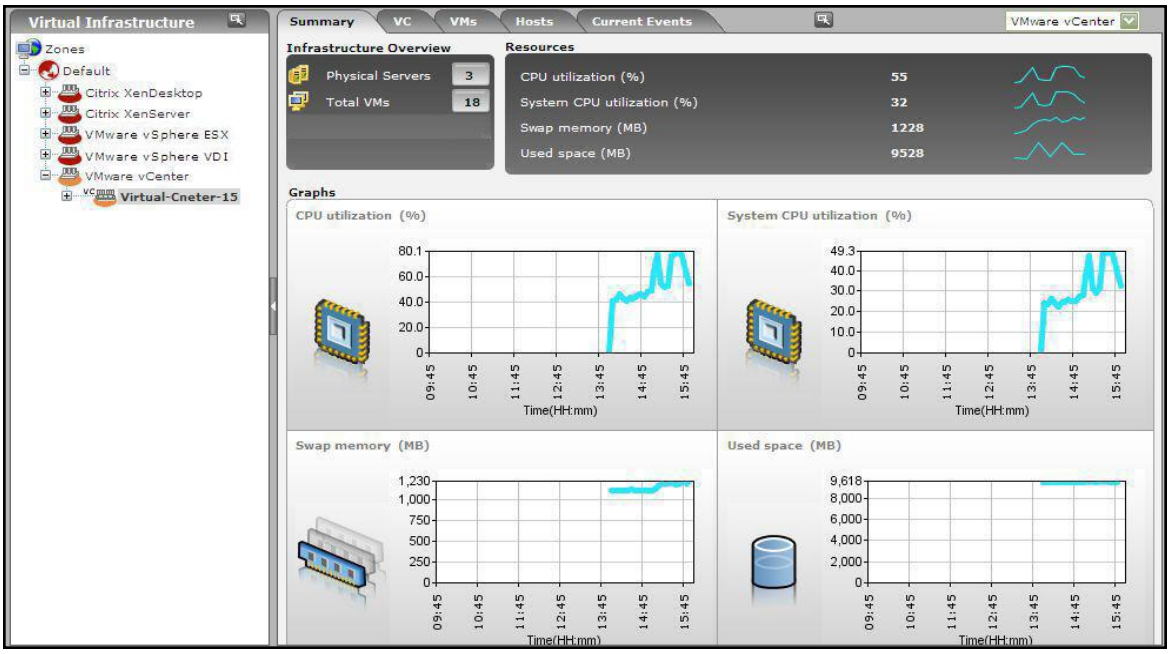


Figure 7.24: The Summary tab page if the node representing a particular vCenter server is clicked in the tree

29. Know how many ESX servers and VMs are managed by the chosen vCenter server using the **Infrastructure Overview** section of the **Summary** tab page. Move your mouse pointer over any number in this section to view the corresponding details.

30. Adjacent to the **Infrastructure Overview** section, you will find a **vCenter Health** section that reports in real-time, the availability and responsiveness of the vCenter server, the load on the server in the terms of current sessions to vCenter, and the current license usage of the server. You can proactively detect the non-availability or a slowdown of the vCenter server, a server overload, or excessive license usage by the server using the metrics reported by this section. Against every value displayed, a miniature graph is available, tracking the time-of-day variations in the values of the corresponding measures. Expand the graph by clicking on it. The expanded graph, by default, reveals how the corresponding measure has performed during the last 1 hour. You can plot the graph for a broader period by choosing a different **Timeline** for the graph. Using this graph, you can easily analyze the performance of the vCenter over time.

31. The **Graphs** section in the **Summary** tab page provides graphs that enable you to assess the CPU, disk, and memory usage of the vCenter server, over a default period of 6 hours. Resource usage trends can be accurately deduced from these graphs, and probable resource crunches can be proactively detected and averted. To zoom into a particular graph, click on it. The graph then expands (see Figure 7.25), so that you can study it clearly and make sound inferences. You can even change the **Timeline** of the graph, so that you can generate resource usage graphs for longer time periods, and perform more effective analysis.

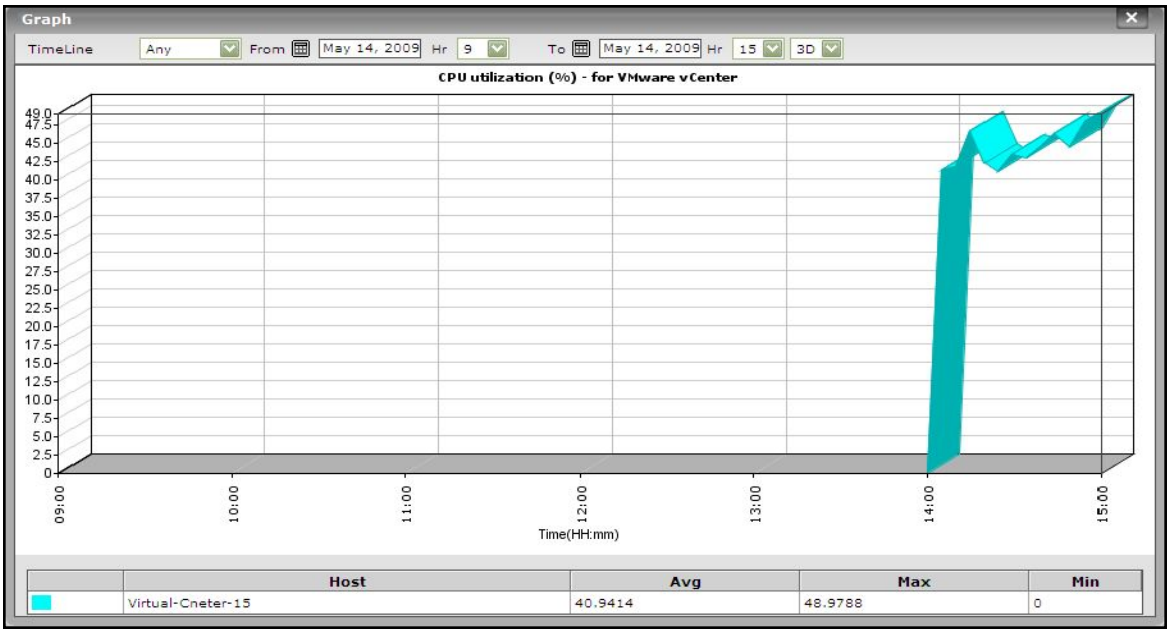


Figure 7.25: Expanding a graph in the Graphs section of the Summary tab page

32. If vCenter manages the target virtualized environment as **folders**, then expanding a vCenter server node in the tree will reveal the folders managed by that vCenter. If you click on the folder node in the tree, the **Summary** tab page will change to provide an overview of the composition and current state of that folder (see Figure 7.26).

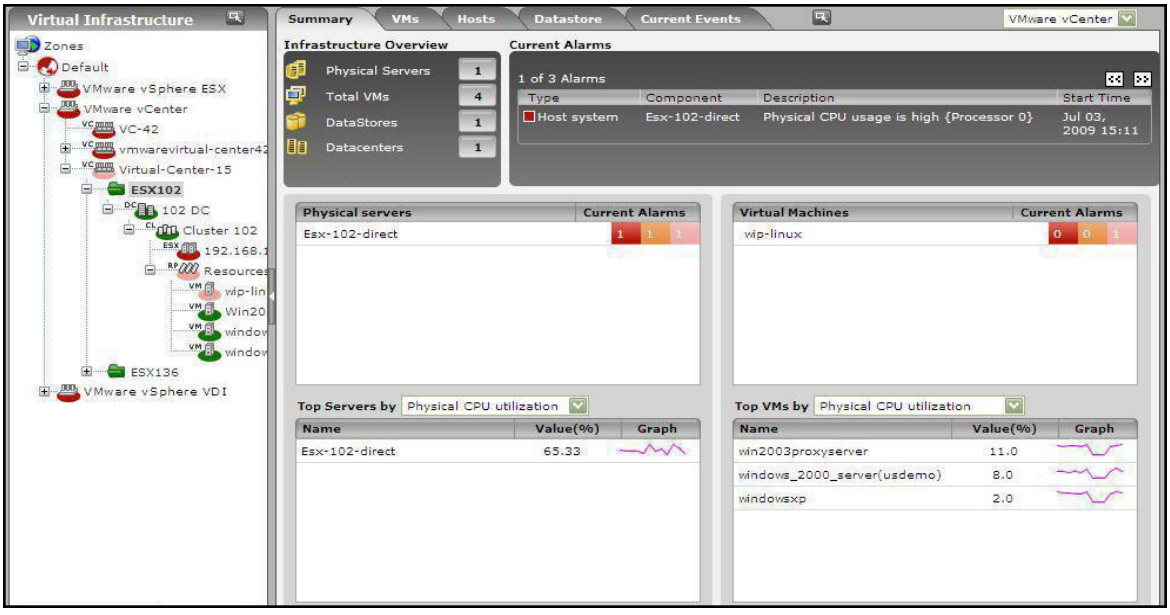


Figure 7.26: The Summary tab page if a folder in the tree is clicked

33. The **Infrastructure Overview** section of Figure 7.26 displays the number of physical servers, VMs, datastores, and datacenters managed by the chosen folder. To know the names of these elements, move

your mouse pointer over the corresponding element count. If you want to take a close look at each of the current alarms, use the **Current Alarms** section, and browse the alarms one after another.

34. In order to figure out which physical servers and VMs are responsible for these alarms, use the **Physical Servers** and **Virtual Machines** sections. These sections list the problem-prone physical servers and virtual machines (as the case may be) in the folder, along with the number and type of problems that each physical server/virtual machine is currently experiencing. Move your mouse pointer over an alarm priority corresponding to a physical server/VM to view the details of alarms of that priority that are currently open on that physical server/VM.

35. To track the resource usage of physical servers and VMs within a folder and to identify those physical servers and VMs that are consuming resources excessively, you can use the **Top Servers by** and **Top VMs by** sections. By default, both these sections list the top-5 physical servers and VMs (as the case may be) in terms of *Physical CPU utilization*. To identify the top-5 consumers of a different resource, you can select a different measure from the list box available in both the sections. You can even add more measures to these list boxes for selection using the procedure discussed in page **674** and page **675** of this document. Likewise, you can view more number of leading resource consumers in this section by changing the default value 5 to another number using the procedure discussed in page **676** of this document.

36. If a folder node exists, then expanding this node in the tree, will reveal the datacenters included in that folder. If no folders exist, then you would have to expand the vCenter server node in the tree to view the datacenter sub-nodes. If you click on the node representing a datacenter in the tree-structure, then the contents of the **Summary** tab page will change to provide an overview of the composition and health of that datacenter (see Figure 7.27).

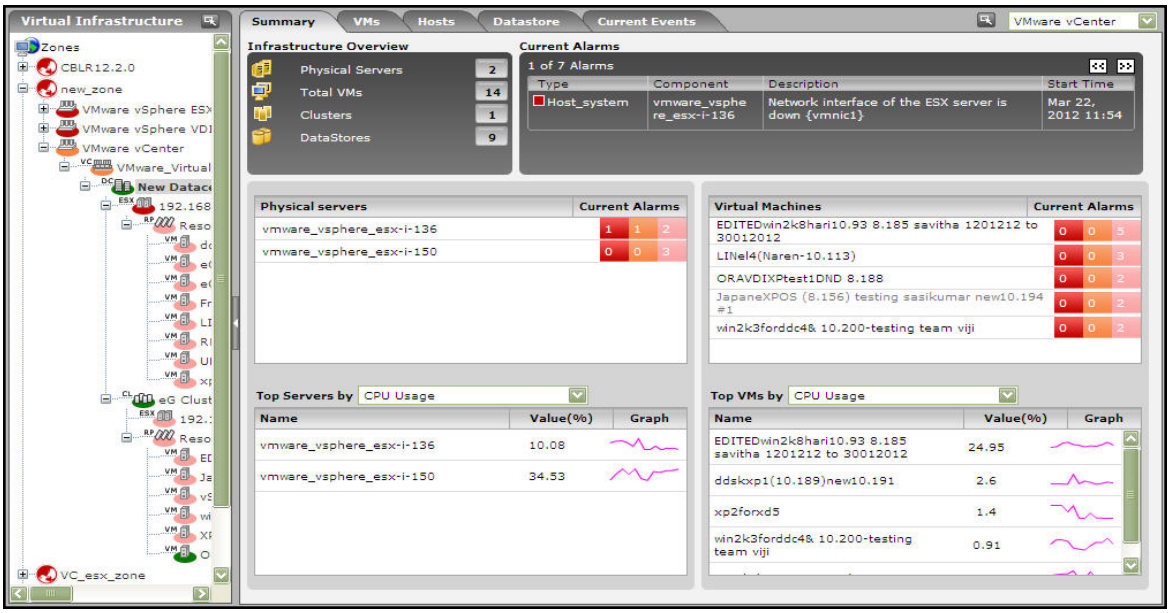


Figure 7.27: The Summary tab page of a datacenter chosen from the tree-structure

37. The **Infrastructure Overview** section of Figure 7.27 displays the number of physical servers, VMs, and datastores managed by the chosen datacenter. To know the names of these elements, move your mouse

pointer over the corresponding element count. If you want to take a close look at each of the current alarms related to the datacenter, use the **Current Alarms** section, and browse the alarms one after another.

38. In order to figure out which physical servers and VMs are responsible for these alarms, use the **Physical Servers** and **Virtual Machines** sections. These sections list the problem-prone physical servers and virtual machines (as the case may be) in the datacenter, along with the number and type of problems that each physical server/virtual machine is currently experiencing. Move your mouse pointer over an alarm priority corresponding to a physical server/VM to view the details of alarms of that priority that are currently open on that physical server/VM.

39. To track the resource usage of physical servers and VMs within a datacenter and to identify those physical servers and VMs that are consuming resources excessively, you can use the **Top Servers by** and **Top VMs by** sections. By default, both these sections list the top-5 physical servers and VMs (as the case may be) in terms of *Physical CPU utilization*. To identify the top-5 consumers of a different resource, you can select a different measure from the list box available in both the sections. You can even add more measures to these list boxes for selection using the procedure discussed in page **674** and page **675** of this document. Likewise, you can view more number of leading resource consumers in this section by changing the default value 5 to another number using the procedure discussed in page **676** of this document.

40. To receive an overview of the performance of a cluster within a datacenter, you will have to click on the cluster sub-node under the datacenter node in the tree-structure. The **Summary** tab page will then change as depicted by Figure 7.28.

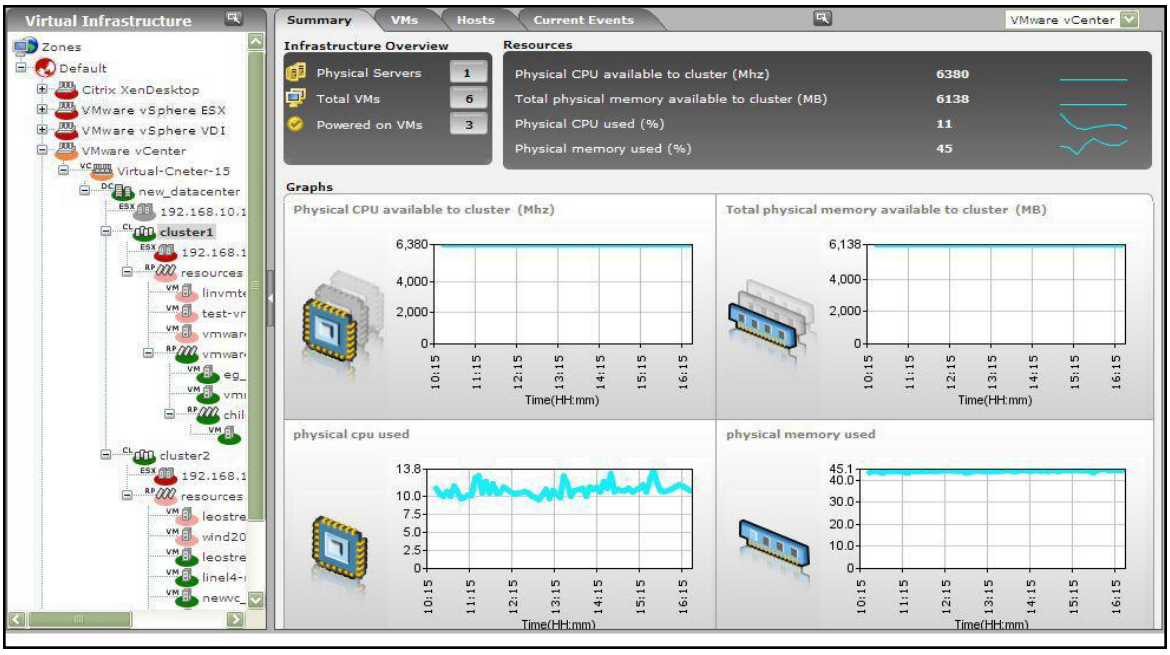


Figure 7.28: The Summary tab page of the cluster sub-node

41. For a particular cluster, the **Summary** tab page will provide a quick summary of the performance of that cluster, so as to enable you to gauge how healthy the cluster is. The **Infrastructure Overview** section of the **Summary** tab page displays the number of physical servers, VMs, powered-on VMs, and resource

pools that the chosen cluster consists of. In addition, the **Current Alarms** information in this section displays the number of issues related to the cluster that are still unresolved; this serves as an effective indicator of the overall health of the cluster. Move your mouse pointer over any number in this section to view the corresponding details.

To be alerted in real-time to abnormalities in resource usage by the cluster, you can use the **Resources** section. This section reports how well the cluster is currently using the physical memory and CPU resources available to it; alongside every usage value displayed in this section, you will find a miniature graph. Click on the graph to expand it, and view the time-of-day variations in resource usage during the last 1 hour (by default). You can even change the **Timeline** of the graph for analyzing usage patterns over a longer period of time. This graph helps you quickly detect disturbing trends in the usage of physical resources, and initiate corrective actions at the earliest.

The **Graphs** section in the **Summary** tab page displays a series of graphs depicts how the cluster has been using the physical memory and CPU resources available to it over a default period of 6 hours. You can magnify any of these graphs by clicking on it. You can even change the **Timeline** of the expanded graph, to facilitate more effective analysis of the resource usage. Using these graphs, you can determine how resource-intensive the cluster has been.

42. To figure out, from a mere glance, the current state of an ESX host included in a cluster, click on the sub-node representing an ESX host under the cluster node. The **Summary** tab page will then change as depicted by Figure 7.29.



Figure 7.29: The Summary tab page for an ESX host

43. For an ESX host, the **Infrastructure Overview** section of Figure 7.29 reveals the number of VMs configured on the ESX host, the number of powered-on VMs, and the number of datastores used by the ESX host. By moving your mouse pointer over any of these numbers, you can view the corresponding details - i.e., the names of VMs/datastores, as the case may be. This way, you can rapidly identify the VMs that are currently powered-off.

The **Resources** section of Figure 7.29 reports the percentage/amount of physical CPU, memory, and disk resources that the ESX host chosen is currently using. Sudden spikes in resource consumption by the ESX host can be promptly detected by closely observing the change in the resource usage levels reported by this section. Every value displayed in this section is accompanied by a miniature graph, which when clicked, expands to reveal how well the corresponding resource usage metric has performed during the last 1 hour (by default). To analyze the behavior of the said measure over a longer period of time, you can change the **Timeline** of the expanded graph. In the event of excessive resource usage by the ESX host, you can use this graph to figure out when exactly the upward trend in resource consumption began, and then, proceed to investigate the reasons for the same.

In addition to the **Resources** section, a **Graphs** section is also available in the **Summary** tab page. This section provides a series of graphs, which track the physical CPU, memory, and disk resources used by the ESX host during the last 6 hours (by default). Click on a graph in this section to expand it. Using the expanded graph, you can even change the **Timeline** of the graph, so that you can observe usage patterns over longer periods of time.

44. If resource pools are configured on a cluster, then each resource pool will appear as a sub-node of that cluster node. When a resource pool sub-node is clicked, the **Summary** tab will change as depicted by Figure 7.30.



Figure 7.30: The Summary tab page of a resource pool

45. The **Infrastructure Overview** section of Figure 7.30 reveals the number of VMs and powered-on VMs Move your mouse pointer over any number in this section to view the corresponding details. This information enables you to swiftly identify powered-off VMs and also assess the overall health of the resource pool.

The **Resources** section reports the availability and usage of critical resources allocated to the resource pool, in real-time. Any sudden increase in usage of a resource can be promptly detected using the values reported by this section. Adjacent to every value, a miniature graph is provided. To track the usage of a resource over time, click on the miniature graph. The graph expands to reveal how well the resource pool used the

corresponding resource during the last 1 hour (by default). You can even change the **Timeline** of the graph to understand resource usage trends over longer time periods.

The **Graphs** section of the tab page provides time-of-day graphs that reveal the variations in the physical CPU and memory usage by the resource pool during the last 6 hours (by default). Analysis of the resource consumption of the pool over time reveals whether the pool has been using resources optimally or inefficiently. Clicking on a graph expands it. You can change the **Timeline** to analyze resource usage over broader time periods.

7.3 The VC Tab Page

Promptly detect issues with vCenter and swiftly diagnose the root-cause of these issues, using the vc tab page offered by the VM dashboard. This tab page lists one/all vCenter servers in the environment (depending upon the node chosen from the tree), and reports the number of ESX servers managed, the resource usage, and the overall health of each vCenter server.

This section discusses how the contents of this tab page change with the node chosen from the tree.

1. If the global **Zones** node is chosen from the tree, then the **VC** tab page will display the names of all vCenter servers included in all configured zones, and will report a default set of metrics per vCenter server, revealing the availability and resource usage of each server. Resource-intensive/unavailable vCenter servers in the target environment can thus be instantly identified.

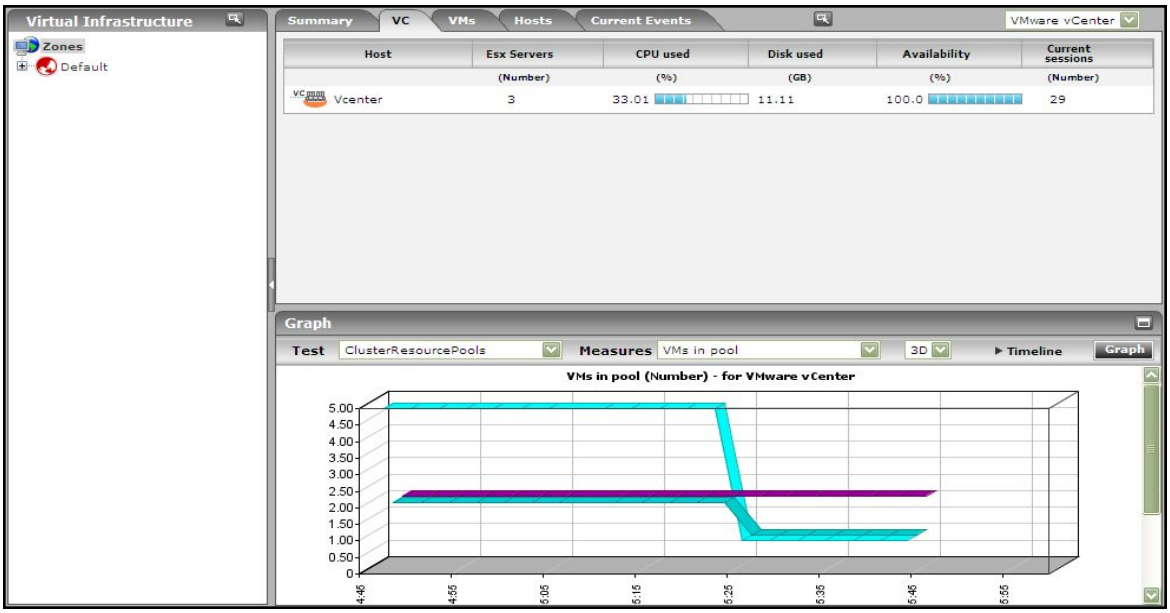


Figure 7.50: The VC tab page if the global Zones node is clicked

2. The default measure list that accompanies every vCenter listed in the **VC** tab page can be modified. For this purpose, follow the steps given below:

- Click on the button at the right, top corner of the tab page. Figure 7.51 will appear.

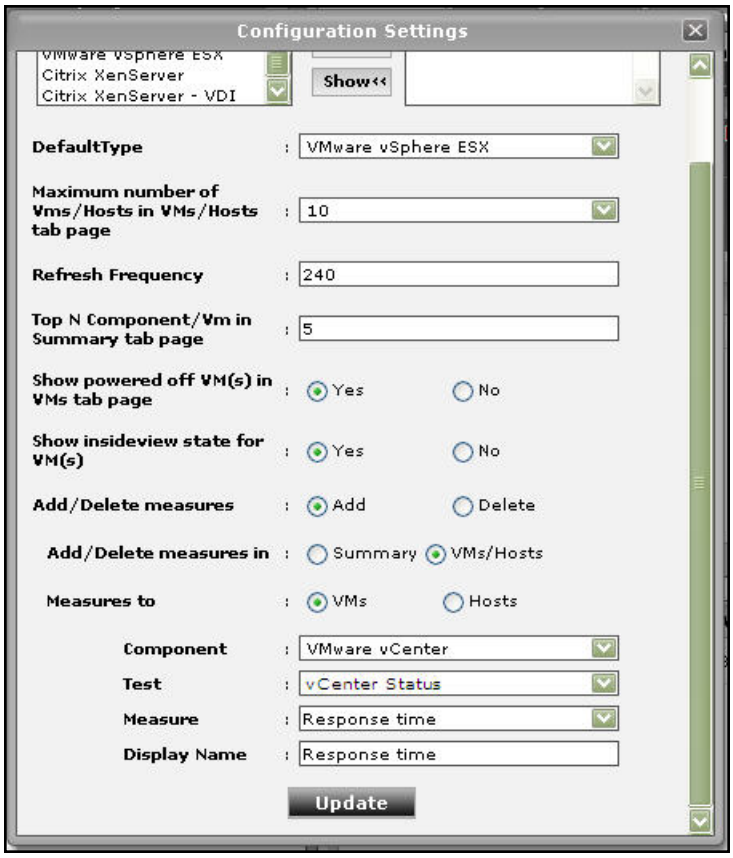


Figure 7.51: Adding a new measure to the VC tab page

- Select **Add** from **Add/Delete measures** section to add a new measure to the **VC** tab page.
- Select **VMs/Hosts** from the **Add/Delete measures in** section in Figure 7.51.
- Set the **Measures to** flag to **VMs**.
- Then, set *VMware vCenter* as the **Component** type. Select the **Test** that reports the measure to be added, and then pick the **Measure**. Provide a **Display Name** for the new measure, and finally, click the **Update** button.
- To remove a measure that pre-exists in the **VC** tab page, select **Delete** from **Add/Delete**, select **VM/Hosts** from the **Add/Delete measures in** section, set the **Measures to** flag to **VM**, select the **Component** type, pick the **Test**, choose the **Measure**, and click the **Update** button.

3. In the event of a slowdown in the performance of a vCenter server, you can click on the name of that server in the **VC** tab page; this will lead you straight to the layer model of the vCenter server, which reveals the exact layer where the problem has originated. Click on the problem layer to view the problem test, and click on the problem test to view the measures (see Figure 7.52).

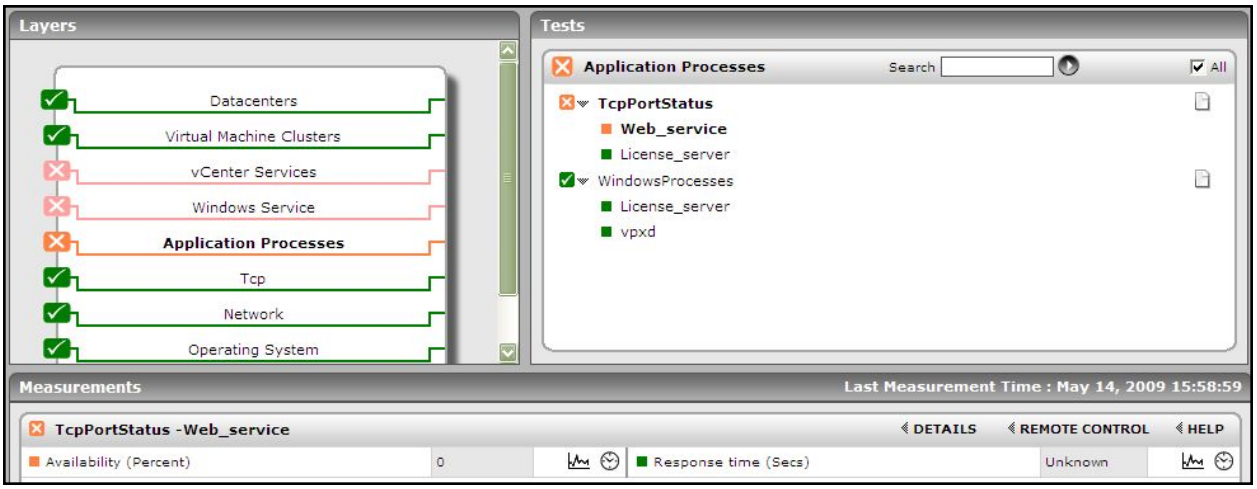


Figure 7.52: The layer model, tests, and measures pertaining to the vCenter server

4. The **VC** tab page also embeds a time-of-day graph, which, by default, reveals the variations in the number of VMs in each resource pool configured within each cluster on vCenter, during the last 1 hour. Accordingly, **ClusterResourcePools** is chosen by default from the **Test** list, the default **Measure** is **VMs in pool**, and the default **Timeline** is 1 hour. You can select a different **Test** and **Measure** combination for the graph, and also define a different **Timeline** for the graph, if required. Using this graph, you can efficiently analyze performance trends and proactively detect performance issues.
5. To view the zones configured with one/more virtual component-types, expand the global **Zones** node in the tree. The individual zones will appear as sub-nodes of the global **Zones** node. Selecting a particular zone will reveal the performance details related to each vCenter server within that zone in the **VC** tab page.
6. Alternatively, you can select the *VMware vCenter* sub-node under a zone node to see how all the vCenter servers in a zone are performing (see Figure 7.53).

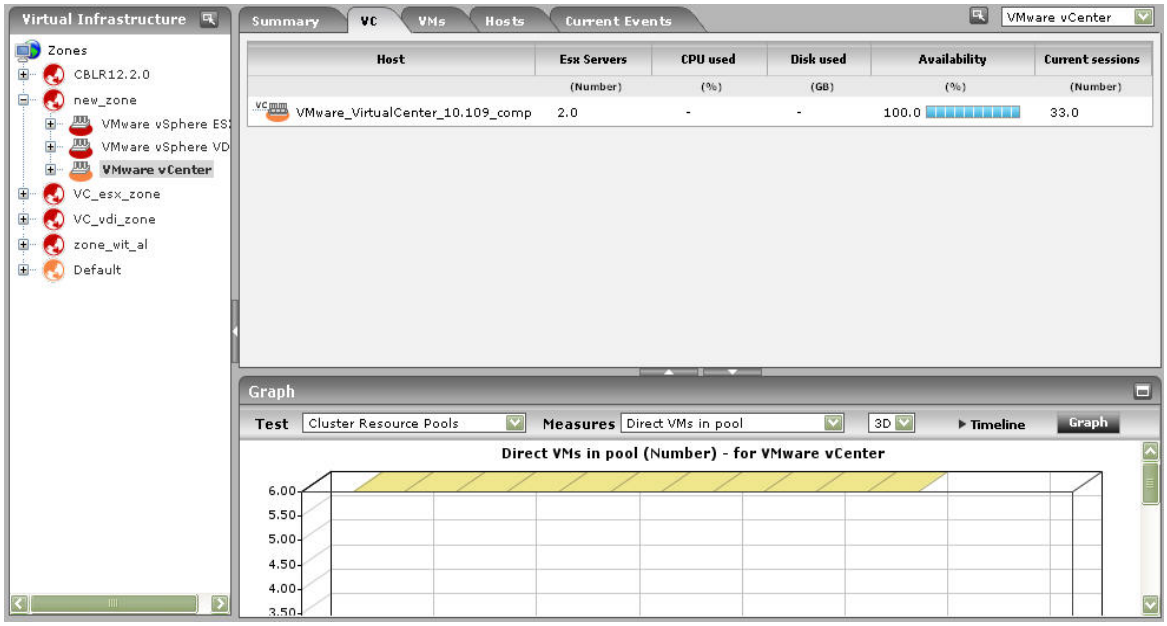


Figure 7.53: The VC tab page if the 'VMware vCenter' node is chosen from the tree

7. If you want to check whether a particular vCenter server within a zone is available or not, and if available, how well its using the resources available to it, select the node representing a vCenter server under the *VMware vCenter* node in the tree structure (see Figure 7.54).

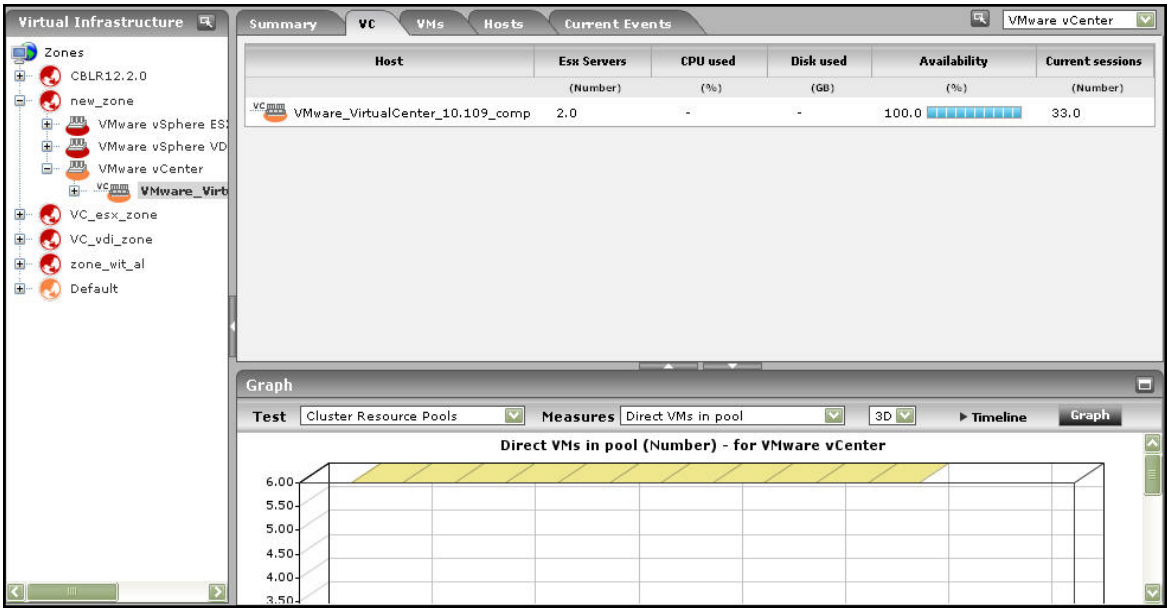


Figure 7.54: The VC tab page for a particular vCenter server chosen from the tree

7.4 The Hosts Tab Page

The **Hosts** tab, as mentioned already, provides insights into the current state and the extent to which resources are currently used by the managed virtual hosts in the environment. This tab page sheds light on resource-intensive hosts, and embeds efficient drill downs to discover the underlying cause for the high resource consumption of a host.

As stated earlier, this tab page, by default, provides the details of those managed virtual hosts that are of the type *VMware vSphere ESX*. Similarly, the details of the top-10 virtual hosts alone will be displayed in this tab page, by default. These default settings can however be overridden using the procedure already discussed.

Here again, the hosts displayed depend upon the node chosen from the tree-structure in the left panel. This section explains how the contents of this tab page change with context.

1. If the global **Zones** node is selected in the left panel, then the **Hosts** tab page in the right panel will list the top 10 (by default) virtual hosts that the eG agent auto-discovers from across all the managed virtual hosts of the chosen type (see Figure 7.55). This list is typically sorted by the current state of the hosts. If state is the same across hosts, then the hosts are arranged in the order of their host names. Against every virtual host, the state of the host, the total number of VMs configured on the host, the number of VMs powered-on, and a default set of metrics indicating the extent to which resources are currently utilized by the host, will be displayed. This default measure list can be modified by adding new measures to be displayed in the **Hosts** tab page or by removing one/more existing measures from the tab page. To achieve this, follow the

steps given below:

2. Besides revealing the VM load on each virtual host, this tab page also enables administrators to instantly figure out the following:

   - Is any host experiencing performance issues currently?
   - Is any VM on this host currently powered off?
   - Are there any resource-hungry virtual hosts in the environment? If so, which ones are they?

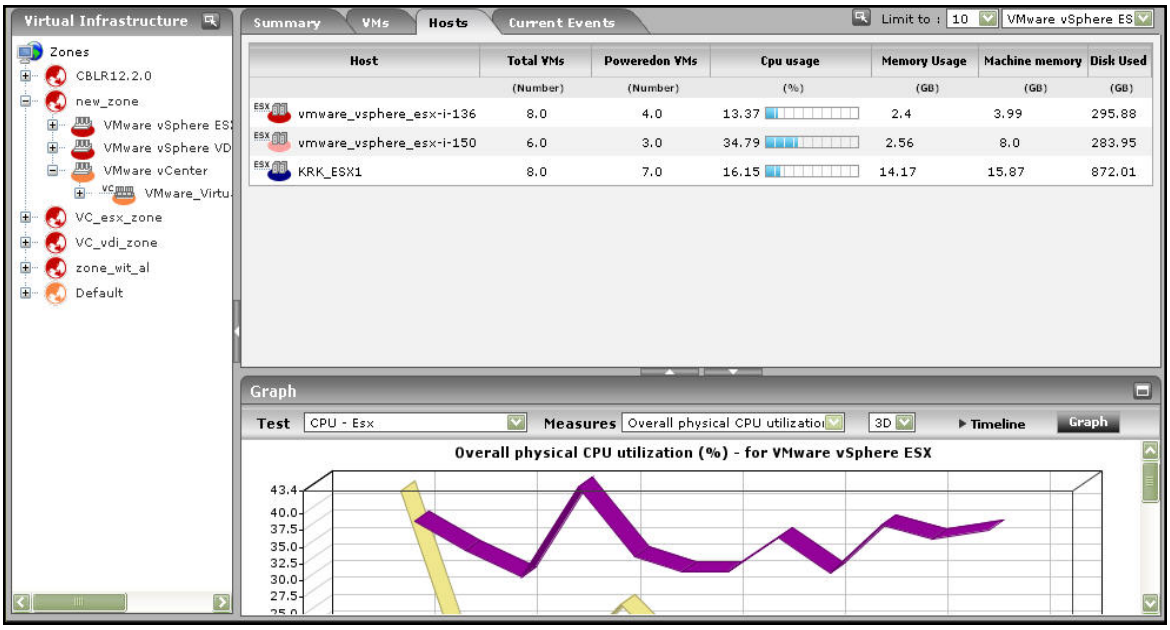


Figure 7.55: Zooming into the performance of a VM from the VM dashboard

3. Below the host list, you will find a **Graph** section (see Figure 7.55). This section will be available in every tab page in the right panel of the VM dashboard. The **Test** list in this section will be populated with all the tests related to the virtual hosts of the chosen type. **Note that VM-related tests will not be available for selection in this list**. These tests will be sorted in the order of the test names, and the top test in the sorted test list will be selected by default in the **Test** list box. The **Measures** list naturally, will be populated with those metrics that the default **Test** reports. Since this list too is sorted in the order of the measure names, the top measure in the sorted list will be chosen by default in the **Measures** list. Accordingly, for the global **Zones** node in the left panel, the graph that appears in this section traces the variations in the default measure across the top 10 (by default) virtual hosts (of the chosen type) in the environment during the default timeline of 1 hour. Using this graph, administrators can compare the performance of a particular measure across virtual hosts, and accurately identify those virtual hosts that are poor performers. If need be, you can plot a comparison graph for a different **Test-Measure** pair, for a different timeline. To change the timeline, click on **Timeline**; the window depicted by Figure 7.56 will then pop out, allowing you to change the date and time.



Figure 7.56: The Timeline window

4. The default graph will be a **3D** graph. If required, you can select the **2D** option from the drop-down list in the **Graph** section to generate a 2D graph.

5. A legend is provided at the end of every graph clearly indicating which virtual host is represented using which color in the graph. This is accompanied by the **Avg**, **Max**, and **Min** values that the chosen **Measure** has recorded for every virtual host during the chosen **Timeline**. To view the legend, you can scroll down the **Graph** section. If need be, you can hide/unhide the **Graph** section or the **Tree** in the left panel using the buttons provided in the **VM Dashboard**. These buttons have already been described in the previous sections.

   Note:

   By default, while plotting a graph for a descriptor-based test across virtual hosts, eG Enterprise aggregates the measure values across all descriptors for a host, and plots only a single value for each virtual host. Accordingly, the **AggregateGraphs** flag in the **Configuration Settings** window that appears when the  button is clicked is set to **Yes** by default. Sometimes, administrators might want the graph to plots values per descriptor. In such a case, set the **AggregateGraphs** flag to **No**.

6. Now, let us see what happens if a particular zone is chosen from the tree-structure in the left panel. When this is done, the **Hosts** list in the right panel will change to display the state and resource usage metrics related to the top 10 virtual hosts that are operating within the zone that is clicked on. This information helps administrators analyze how the performance of one/more virtual hosts in a zone impact the performance of the zone as a whole.

7. If you then drill down a particular zone in the tree, you will be able to view the virtual component-types that form part of the zone, and their current state. If you click on a particular component-type in the tree, the **Hosts** tab page in the right panel will allow you to view the state and usage metrics pertaining to the top-10 (by default) virtual hosts of that type that are included in the corresponding zone (see Figure 7.57).

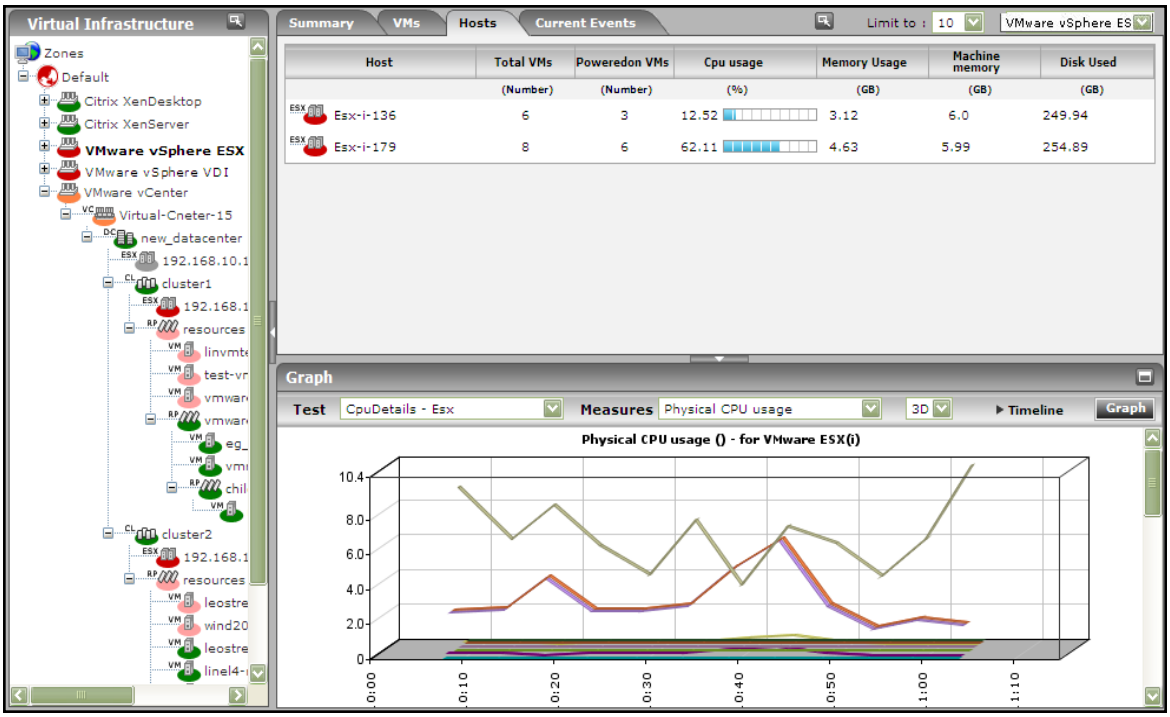


Figure 7.57: he Hosts tab page for a virtual component type in the tree

8. If the vCenter servers in your environment are being monitored as part of a zone, then expanding that zone's node in the tree would reveal the *VMware vCenter* component-type. When this component-type is clicked on, the **Hosts** tab page will change to display the state and resource usage metrics related to the top-10 (by default) virtual hosts managed by all vCenter servers included in that zone.

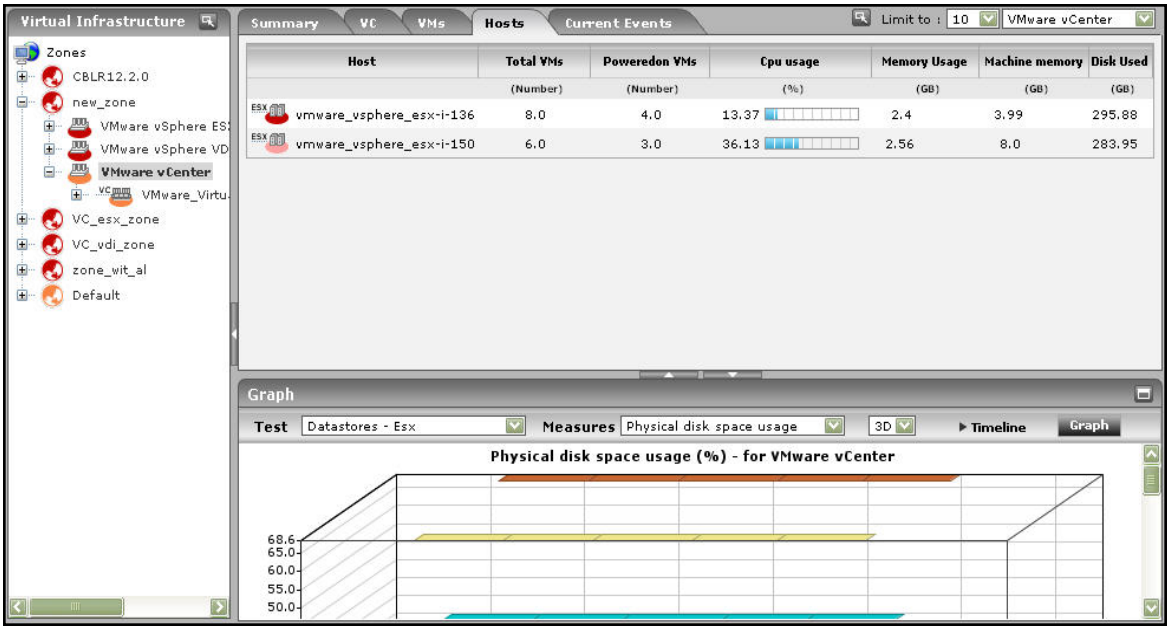


Figure 7.58: The Hosts tab page if the 'VMware vCenter' node is chosen from the tree

9. To know which vCenter servers have been added to a zone, just expand the *VMware vCenter* sub-node

under the zone node in the tree. This will reveal the name and the current state of the vCenter servers in that zone. Clicking on a particular vCenter server in the tree will provide the complete details of the top-10 hosts (by default) managed by that vCenter component.

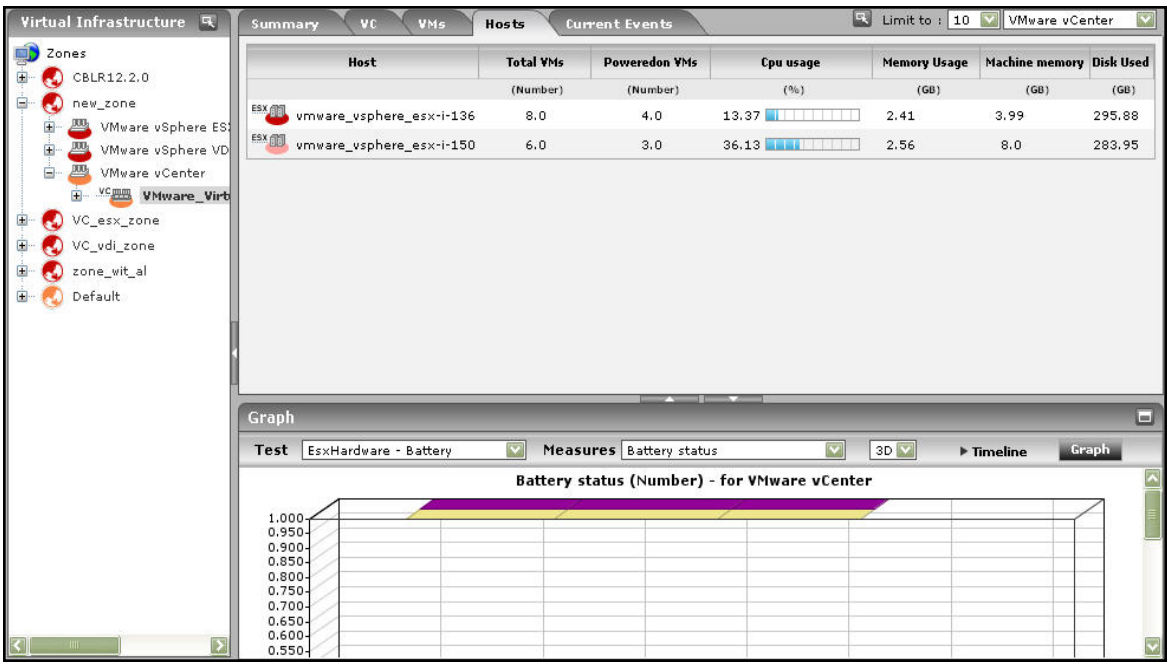


Figure 7.59: The Hosts tab page if a particular vCenter server is chosen from the tree

10. Typically, when you expand a vCenter server node in the tree, the physical ESX servers that are being managed by that vCenter server will appear.

    Note:

    While the vCenter server tree will list even those ESX servers that are not monitored by eG Enterprise, the tree will not indicate the current state of such servers; also, clicking on any such server will not display corresponding performance information in the tab pages in the right panel.

    However, if folders have been configured on the vCenter server, then these folders will appear as sub-nodes of the vCenter server node. If you click on a folder, then the **Hosts** tab page will indicate how well the top-10 hosts (by default) within the folder are performing.

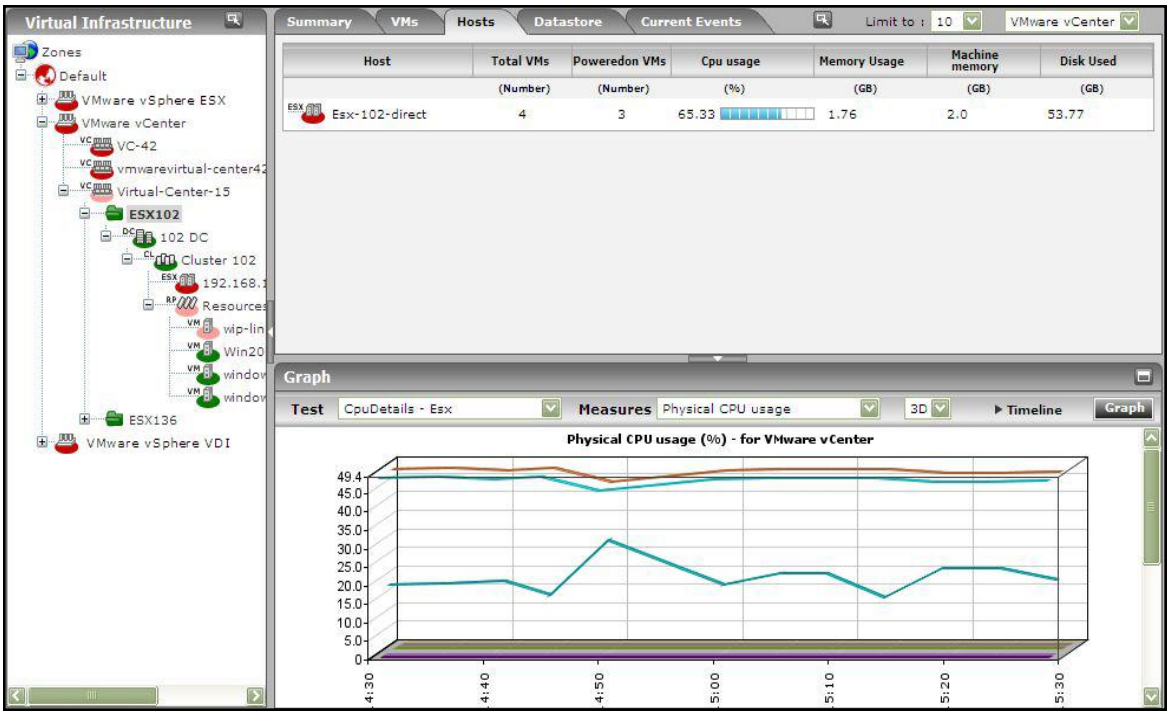


Figure 7.60: The Hosts tab page for a folder selected from the tree

11. If folders exist within a vCenter server, then expanding the folder node reveals sub-nodes representing the datacenters that have been configured on the vCenter server. If no folders exist, then expanding the vCenter server node will reveal the datacenter sub-nodes. If you click on a datacenter, then the **Hosts** tab page will indicate how well the top-10 hosts (by default) within that datacenter are performing (see Figure 7.61).

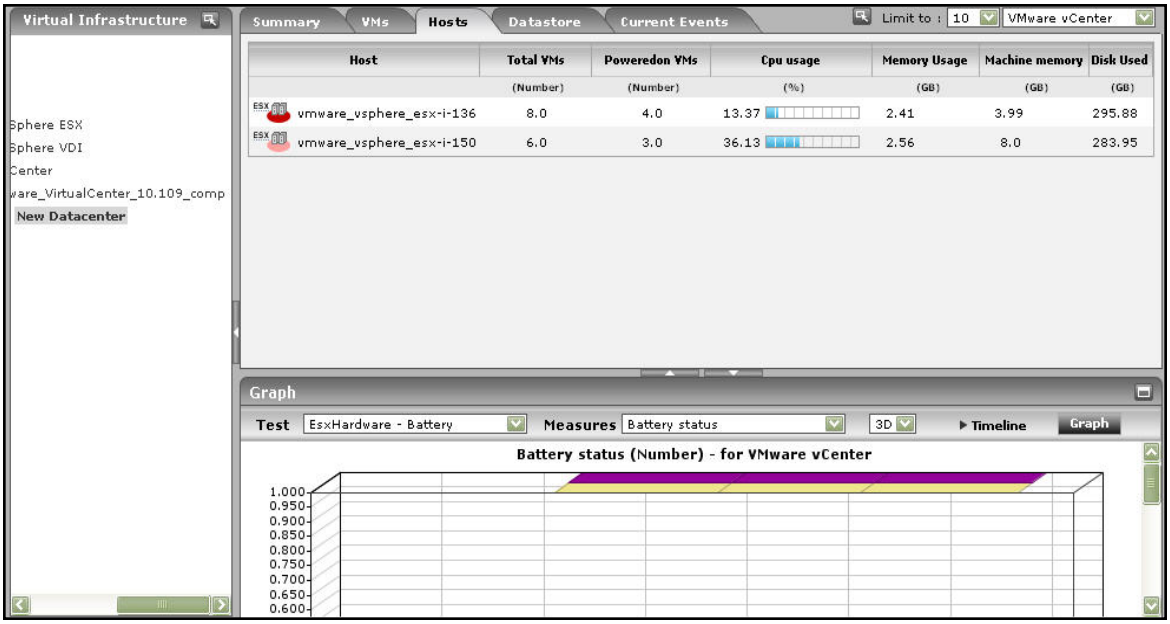


Figure 7.61: The Hosts tab page for a datacenter chosen from the tree

12. Many vCenter installations manage clusters of ESX servers. If such clusters have been configured on any monitored vCenter server, then, in the **Virtual Infrastructure** tree, these clusters will appear as sub-nodes of the datacenter node. If you click on a cluster sub-node in the tree, the **Hosts** tab page will reveal the state and performance information pertaining to the top-10 hosts (by default) that are part of the cluster clicked on.

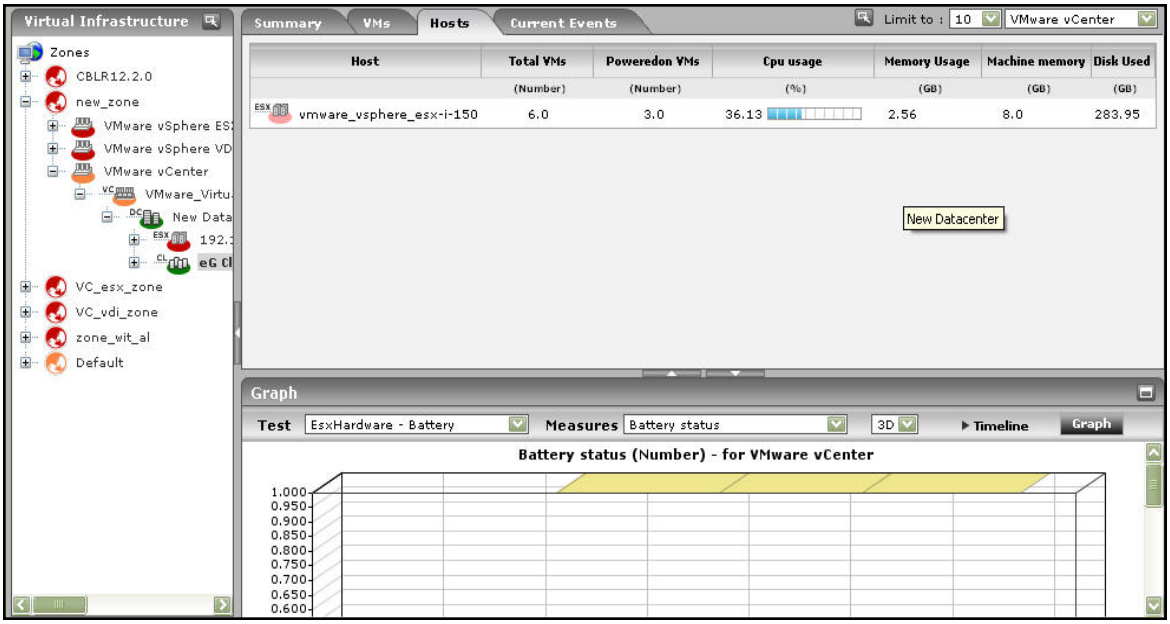


Figure 7.62: The Hosts tab page if the cluster sub-node is clicked on the tree

13. To view the individual virtual hosts that are part of a zone, do any of the following

    - Expand the nodes representing the virtual component-types in the zone;
    - If the zone consists of components of type *VMware vCenter*, expand the node representing the monitored vCenter server in your environment;
    - If datacenters are configured on a monitored vCenter server, expand a datacenter sub-node under the vCenter server node;
    - If clusters are configured within a datacenter, expand the cluster sub-node;

    If you then click on a virtual host in the tree, the **Hosts** tab page will change to display the state and measures extracted from the chosen virtual host alone (see Figure 7.63).

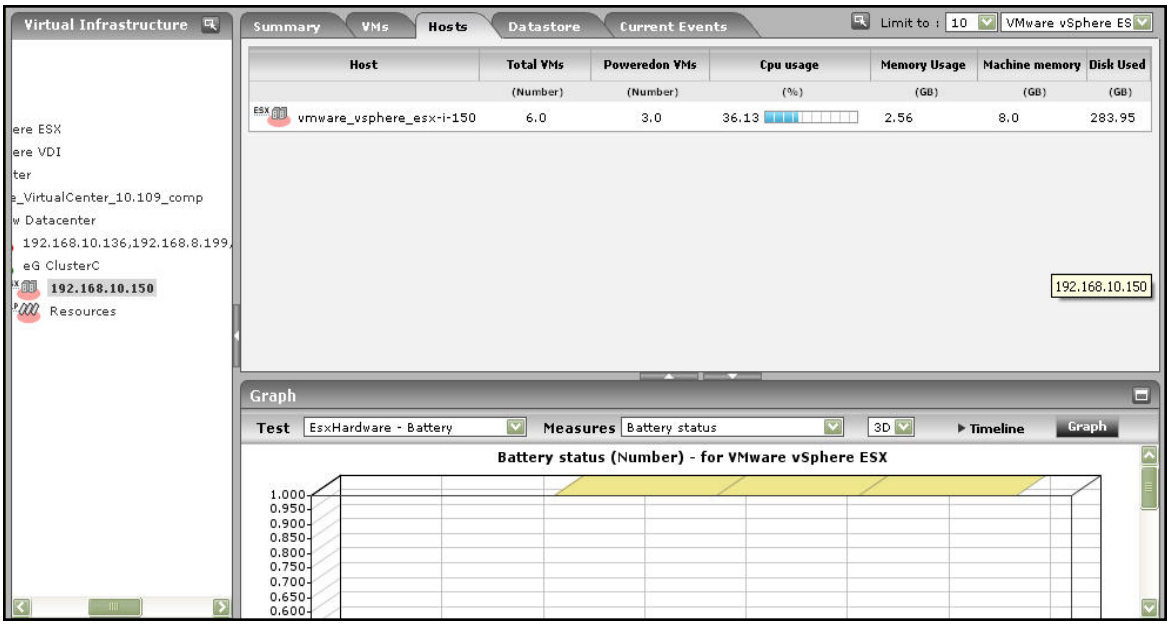


Figure 7.63: The VMs tab page for a particular virtual host selected from the tree

14. If a virtual host in the **Hosts** tab page is found to be in a critical state, then to zoom into the problems affecting the health of that virtual host, simply click on it. Figure 7.64 then appears revealing the layer model of the virtual host, and clearly indicating the problem layer. While clicking on the problem layer will reveal the problem test, a click away from the problem test is the problem measure, which sheds light on the root-cause of the problem with the virtual host (see Figure 7.64). To return to the VM dashboard, just click on the **Back to Virtual Dashboard** link at the right, top corner of the layer model page.

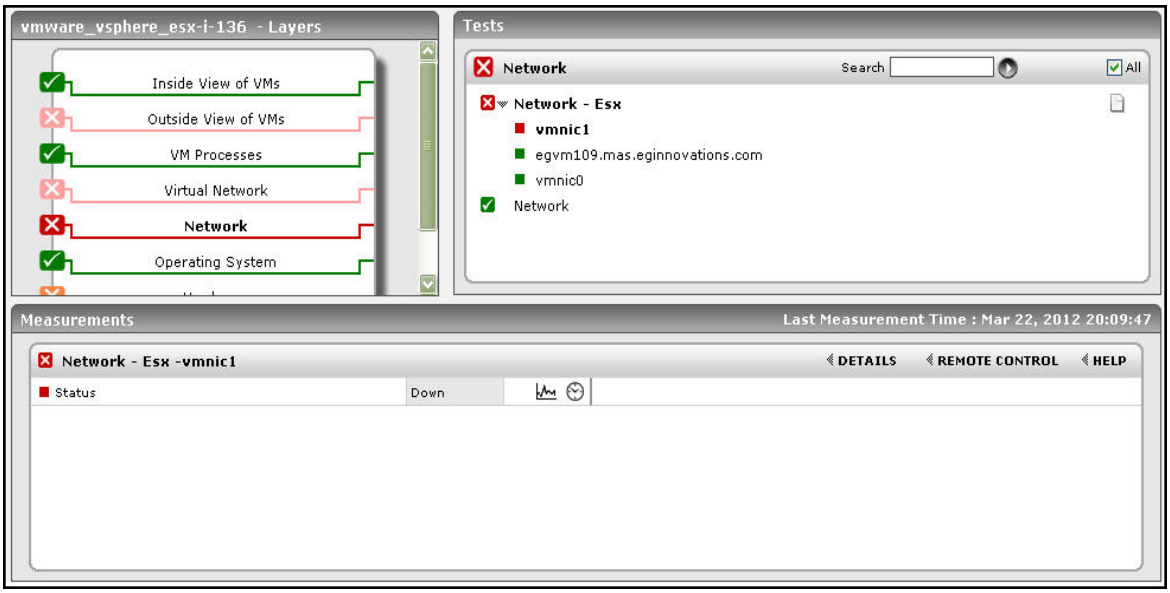


Figure 7.64: The layer model of the virtual host

7.5 The Datastores Tab Page

The **Datastores** tab page lists the datastores used by a particular ESX host or one/more ESX hosts within a datacenter, and reports their availability and usage, so that unavailable datastores and those that are currently running out of space can be accurately identified, and the hosts affected by this anomaly can be easily isolated.

This section discusses how the contents of this tab page change according to the node chosen from the tree-structure.

1. If the node representing a particular folder is chosen from the tree, then the **Datastores** tab page will display the top-10 (by default) datastores used by the ESX hosts managed by that folder. Against every datastore displayed here, the availability and space usage metrics pertaining to that datastore will be provided, along with the number of VMs and ESX servers that have been using the datastore. Using this information, you can swiftly isolate unavailable or over-utilized datastores, and the number of ESX servers and VMs that have been impacted by the anomaly (see Figure 7.65).

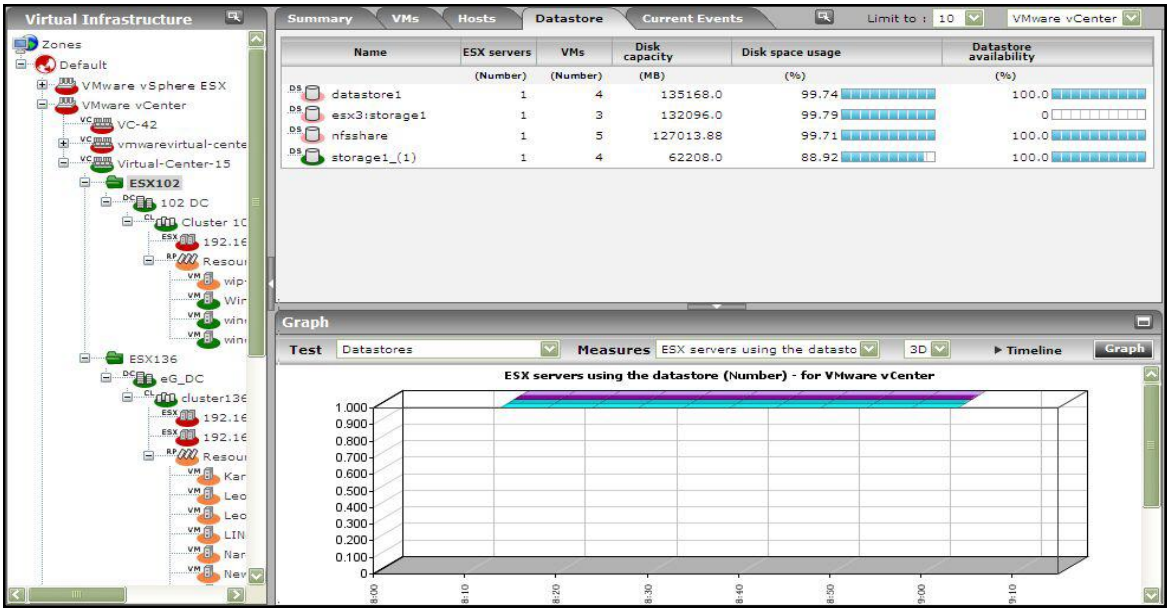


Figure 7.65: The Datastore tab page of a folder node

2. If the node representing a particular datacenter is chosen from the tree, then the **Datastores** tab page will display the top-10 (by default) datastores used by the ESX hosts managed by that datacenter. Against every datastore displayed here, the availability and space usage metrics pertaining to that datastore will be provided, along with the number of VMs and ESX servers that have been using the datastore. Using this information, you can swiftly isolate unavailable or over-utilized datastores, and the number of ESX servers and VMs that have been impacted by the anomaly.

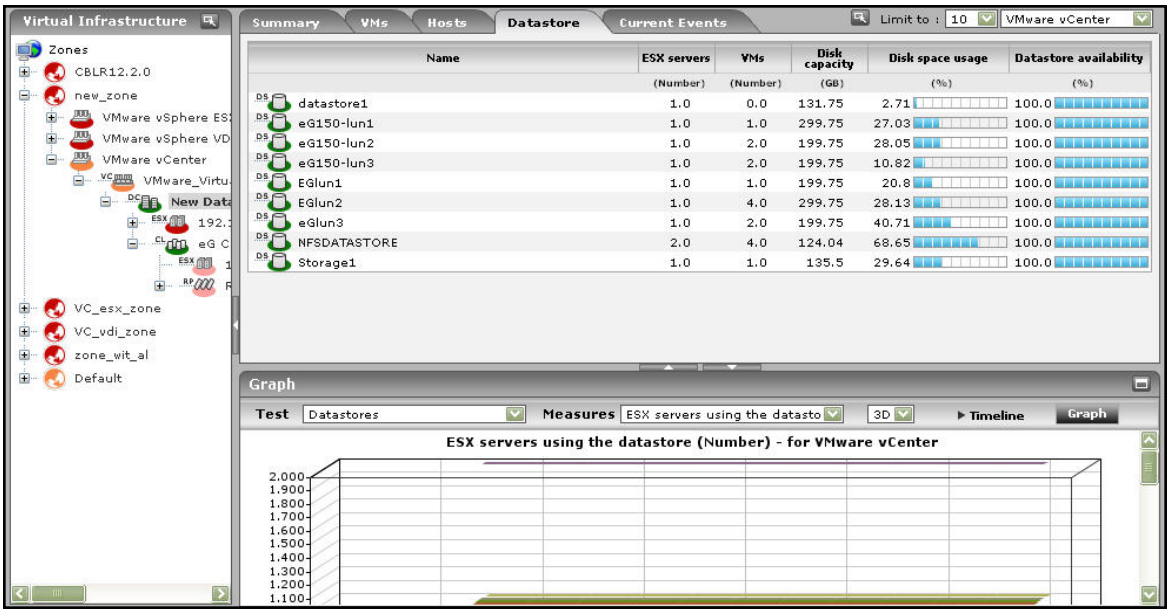


Figure 7.66: The Datastores tab if a particular datacenter is chosen from the tree

3. In addition to listing datastores, the **Datastores** tab page provides a time-of-day graph, which, by default, reveals the number of ESX servers that have been using each datastore during the last 1 hour. Accordingly, the default selection in the **Measure** list is *ESX servers using the datastore*, and the default **Timeline** is 1 hour. Typically, the **Measure** list contains all the measures reported by the default **Test**, which is the *Datastores* test. The options available in the **Measure** list box are sorted in the ascending order of the measure names, and, by default, the first measure in the sorted list will be displayed as the default **Measure**. The default graph clearly indicates how workload on a datastore has varied during the last 1 hour. You can plot the graph for a different measure or a timeline by choosing a different option from the **Measure** list, and by altering the timeline for the graph by clicking the right-arrow button that prefixes **Timeline**.

4. If you click on a datastore in the **Datastores** tab of Figure 7.66, Figure 7.67 will appear leading you straight to the layer model of the *VMware vCenter* server, which manages that datastore. In the event of the non-availability or excessive usage of a datastore, you can use this model to instantly identify the layer that has been affected by the problem with the datastore. Click on the problem layer to view the problem test, and then, click on the problem test to view the problem measure(s). This way, you can easily understand the nature of the problem with the datastore, and how it has impacted the state of the vCenter server.

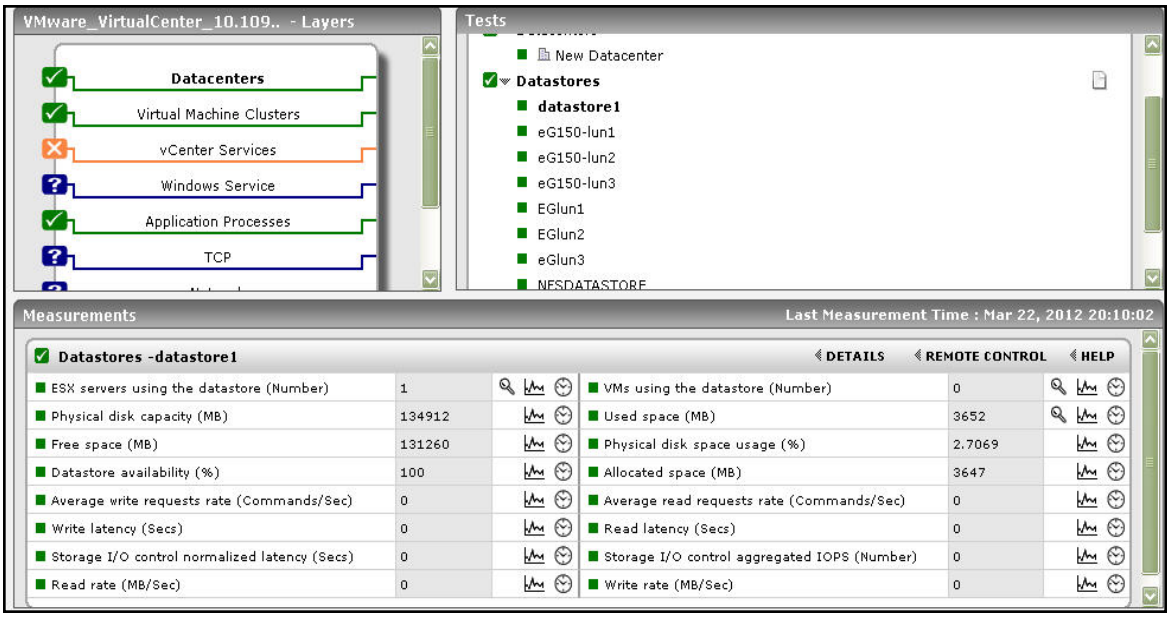


Figure 7.67: The layer model of the vCenter server that appears when a datastore is clicked on in the Datastores tab page

5. Expanding the datacenter node in the tree will enable you to view the ESX hosts that are managed by that datacenter. If a sub-node representing an ESX host is clicked in the tree, then the **Datastores** tab page will only display those datastores that are currently in use by the ESX host and the VMs on it.

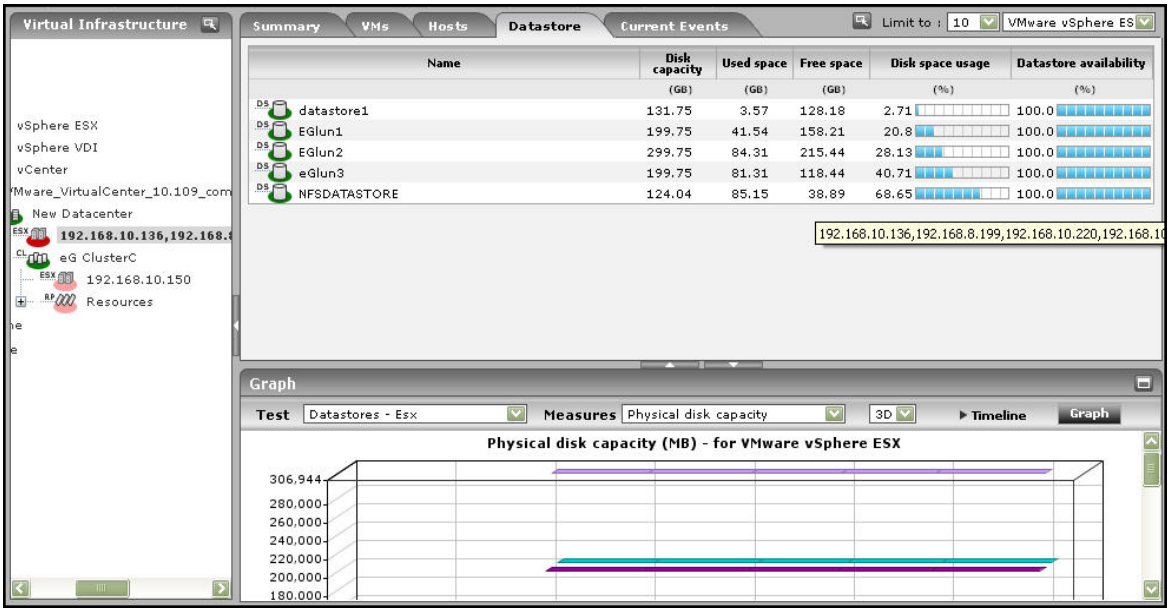


Figure 7.68: The Datastores tab if a particular ESX host is selected from the tree

6. Here again, a time-of-day graph is available, but this graph, by default, reveals the physical disk capacity of each datastore during the last 1 hour. Accordingly, the default selection in the **Measure** list is *Physical disk capacity*, and the default **Timeline** is 1 hour. Typically, the **Measure** list contains all the measures reported by the default **Test**, which is the *Datastores-Esx* test. The options available in the **Measure** list

box are sorted in the ascending order of the measure names, and, by default, the first measure in the sorted list will be displayed as the default **Measure**. You can plot the graph for a different measure or a timeline by choosing a different option from the **Measure** list, and by altering the timeline for the graph by clicking the right-arrow button that prefixes **Timeline**.

7.6 The Current Events Tab Page

The **Current Events** tab page lists the problems that are currently affecting the performance of managed virtual hosts and virtual machines executing on them. This tab page enables administrators to focus on issues related to their virtualized environment alone, without being distracted by the "non-virtual" issues.

As stated earlier, this tab page, by default, lists the alarms pertaining to those virtual hosts that are of type *VMware vSphere ESX*.

Like the other tab pages, the **Current Events** tab page too changes with respect to the node chosen from the tree-structure. To see how, read on.

1. If the global **Zones** node is selected in the left panel, then the **Current Events** tab page in the right panel will list the problems that are adversely impacting the performance of the virtual hosts of the chosen type across the environment (see Figure 7.70). This list is typically sorted by event priority. If event priority is the same across events, then the events are arranged in the order of the names of the problem hosts. The details provided here include the name of the problem component, a brief description of the problem, and the time at which the problem was reported. This information enables administrators to understand how problem-prone their virtualized environment is, and also provides them with pointers to the root-cause of the problem.

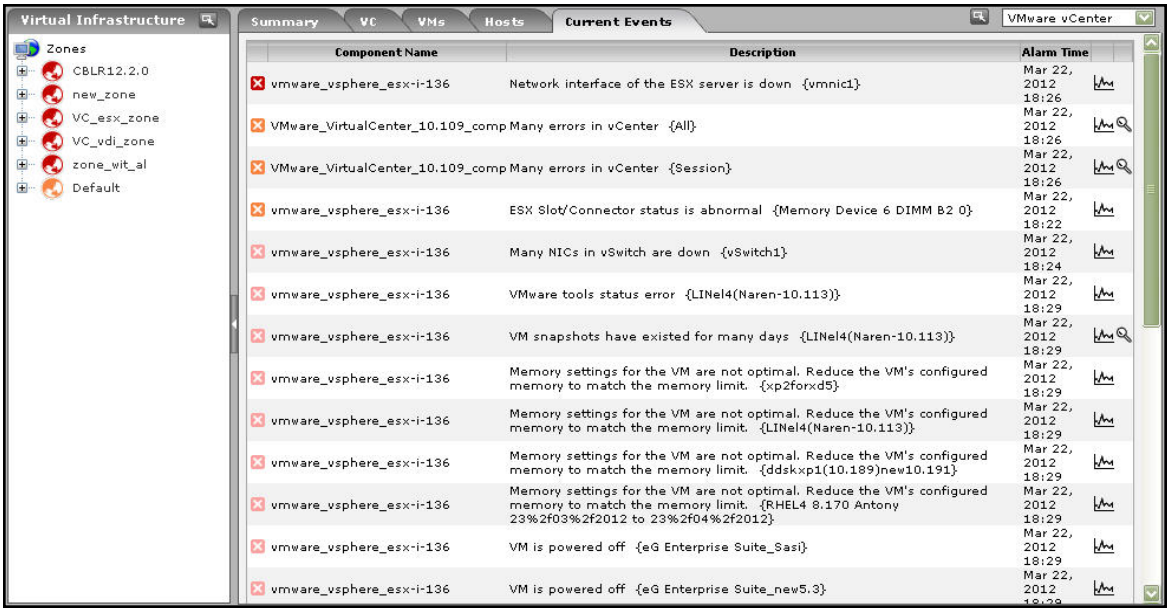


Figure 7.69: The Current Events tab for the global Zones node

2. Now, let us see what happens if a particular zone is chosen from the tree-structure in the left panel. When this is done, the **Current Events** list in the right panel will change to display the problems related to those

virtual hosts that are included in the zone that is clicked on. This information helps administrators identify those problems that are affecting the performance of a particular zone.

3. If you then drill down a particular zone in the tree, you will be able to view the virtual component-types that form part of the zone, and their current state. If you click on a particular component-type in the tree, the **Current Events** tab page in the right panel will allow you to view the problems related to virtual hosts of that type that are included in the corresponding zone (see Figure 7.70).

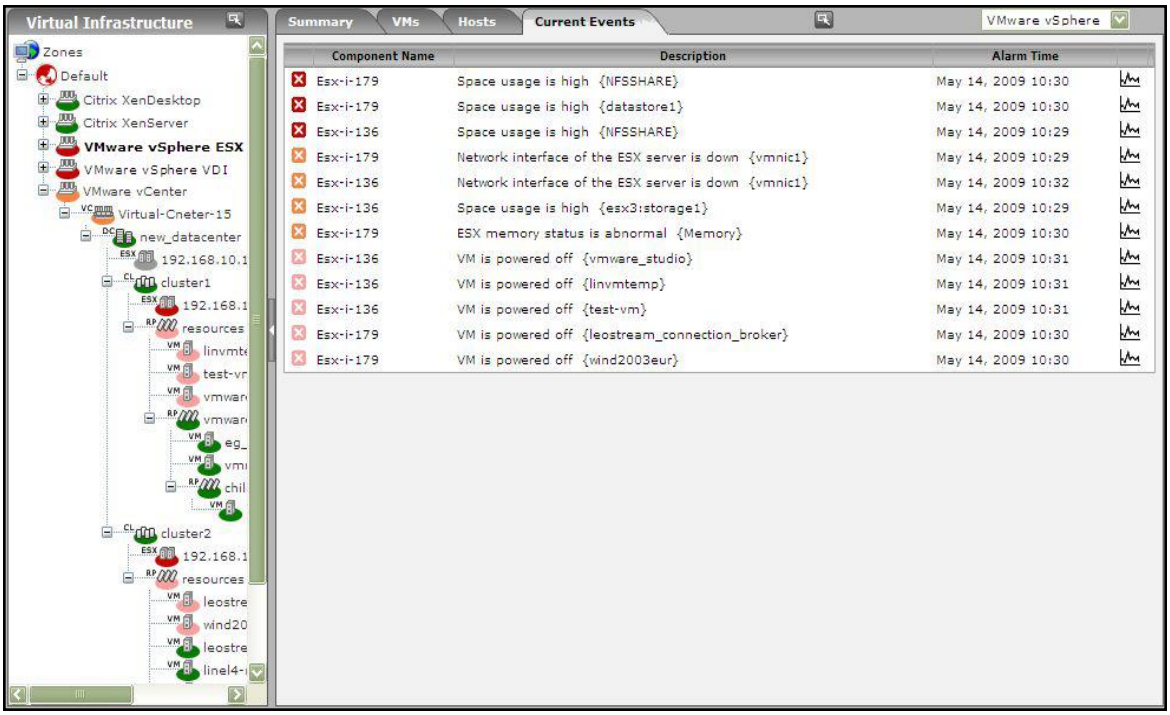


Figure 7.70: The Events tab page for a virtual component type in the tree

4. If the vCenter servers in your environment are being monitored as part of a zone, then expanding that zone's node in the tree would reveal the *VMware vCenter* component-type. When this component-type is clicked on, the **Current Events** tab page will change to display the problem events pertaining all the vCenter servers included in that zone, and those that correspond to the ESX servers managed and VMs by every vCenter server in the zone.

5. To know which vCenter servers have been added to a zone, just expand the *VMware vCenter* sub-node under the zone node in the tree. This will reveal the name and the current state of the vCenter servers in that zone. Clicking on a particular vCenter server in the tree will display problem events pertaining to that vCenter component and those that relate to ESX servers and VMs managed by that vCenter, in the **Current Events** tab page.

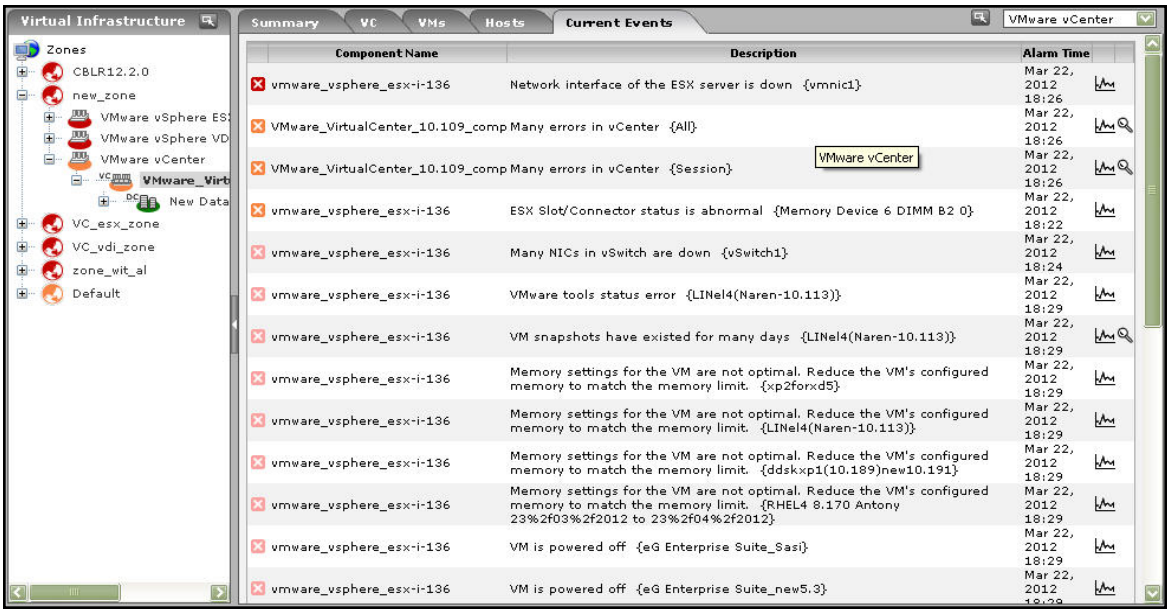


Figure 7.71: The Events tab page for a vCenter server chosen from the tree

6. Typically, when you expand a vCenter server node in the tree, the physical ESX servers that are being managed by that vCenter server will appear.

   Note:

   While the vCenter server tree will list even those ESX servers that are not monitored by eG Enterprise, the tree will not indicate the current state of such servers; also, clicking on any such server will not display corresponding performance information in the tab pages in the right panel.

   However, in some environments, folders may be configured on vCenter server, where every folder could contain one/more datacenters, ESX servers, clusters, datastores, and VMs. In such cases, expanding the vCenter server node will reveal the folders configured on that vCenter server as sub-nodes. Clicking on a folder sub-node will display the complete list of problems currently affecting the performance of datacenters, clusters, ESX servers, and VMs that are included in that folder in the **Current Events** tab page (see Figure 7.72).

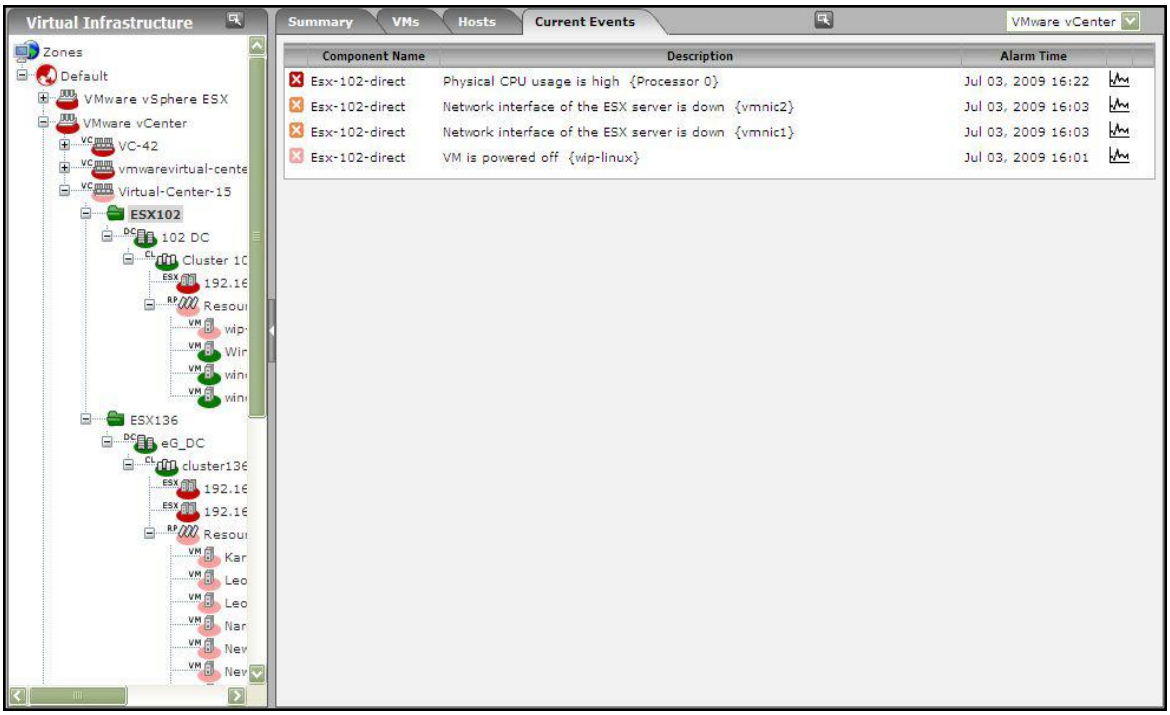


Figure 7.72: The Current Events tab page of a folder

7. Expanding the folder node will reveal the datacenters within that folder. If no folders exist, then expanding a vCenter server node in the tree, will display the datacenters that have been configured within a vCenter. To know what problems are affecting the performance of a particular datacenter currently, click on the node representing a datacenter in the tree. Figure 7.73 then appears.

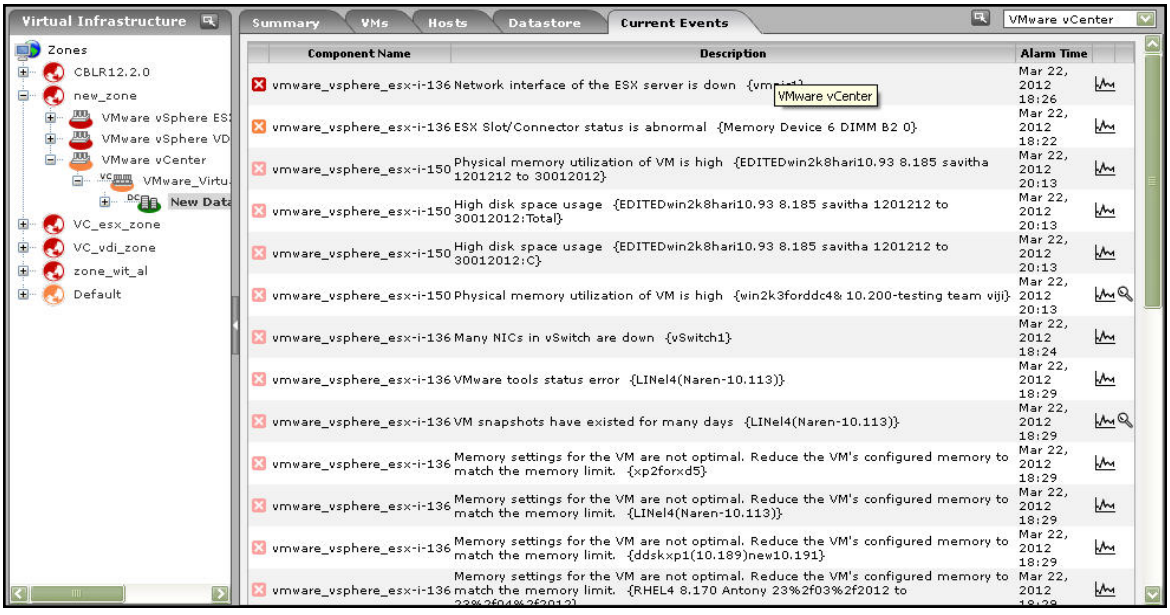


Figure 7.73: The Events tab page for a datacenter chosen from the tree

8. Every datacenter, in turn, may manage clusters of ESX servers. If such clusters have been configured on

any monitored vCenter server, then, in the **Virtual Infrastructure** tree, these clusters will appear as sub-nodes of that datacenter node. If you click on a cluster sub-node in the tree, the **Current Events** tab page will reveal the problem events pertaining to all managed ESX servers included in the cluster.

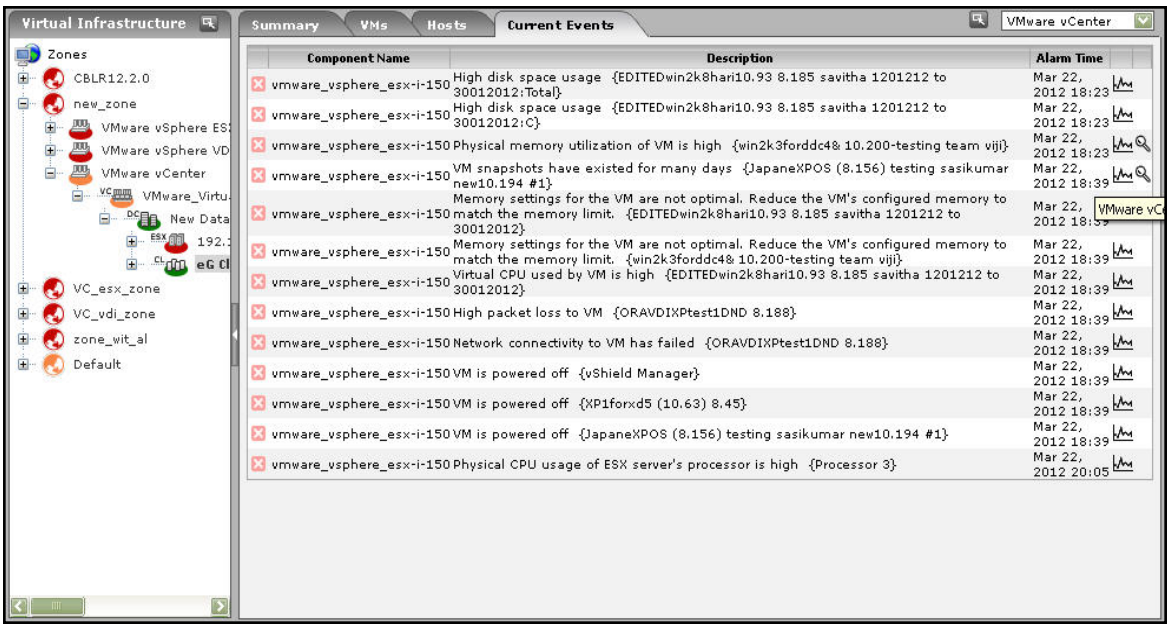


Figure 7.74: The Current Events tab page for a cluster

9. To view the individual virtual hosts that are part of a zone, do any of the following
   - Expand the nodes representing the virtual component-types in the zone;
   - If the zone consists of components of type *VMware vCenter*, expand the node representing the monitored vCenter server in your environment;
   - If datacenters are configured on a monitored vCenter server, expand a datacenter sub-node under the vCenter server node;
   - If clusters are configured within a datacenter, expand the cluster sub-node;
10. If you then click on a virtual host in the tree, the **Current Events** tab page will change to display problems affecting that virtual host (see Figure 7.75).

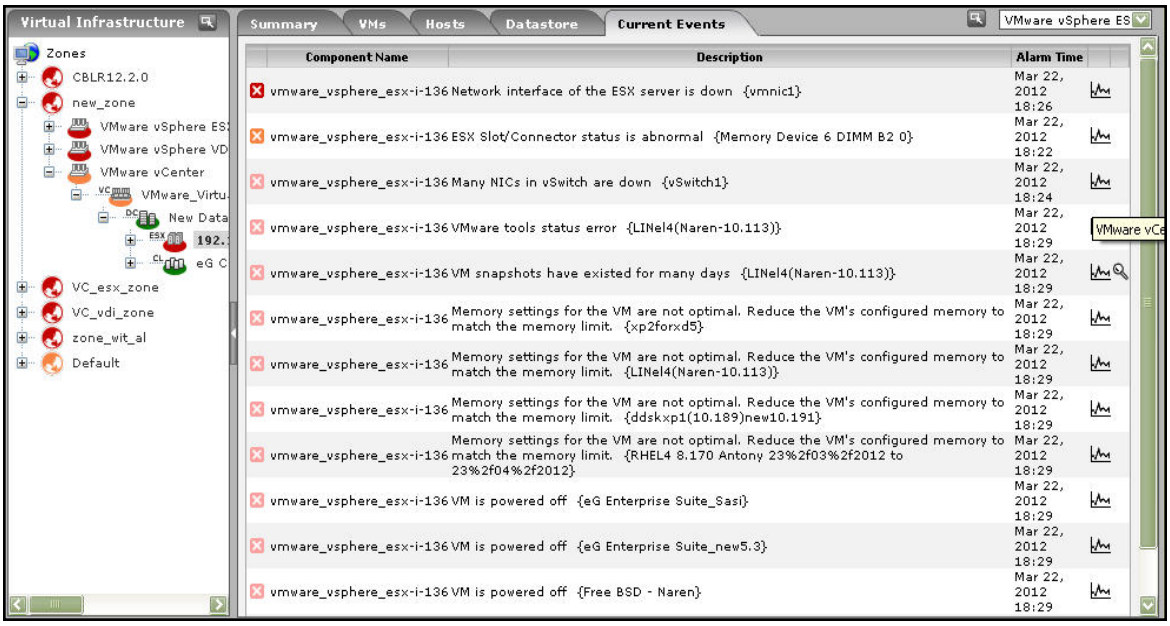


Figure 7.75: The Events tab page for a particular virtual host selected from the tree

11. Similarly, if you expand the virtual host node in the tree-structure, you can view the name and state of those VMs that are executing on that virtual host. If you now click on a VM in the tree, the problem affecting that VM alone will be displayed in this tab page.
12. If resource pools are configured on a virtual host, then the resource pools also will appear as the sub-nodes of the virtual host-node. If you click on a resource pool in the tree, the corresponding **Current Events** tab page will list the current problems affecting the performance of one/more virtual machines included in the resource pool.
13. To perform additional diagnostics on one of the problems listed in this tab page, you can click on the corresponding **Graph** icon. A graph of the problem measure for the last 1 hour (by default) will then appear revealing when exactly the problem occurred.
14. Clicking on an event listed in this tab page will lead you to the layer model page of the problem component, using which you can quickly determine the problem layer, test, and measure.

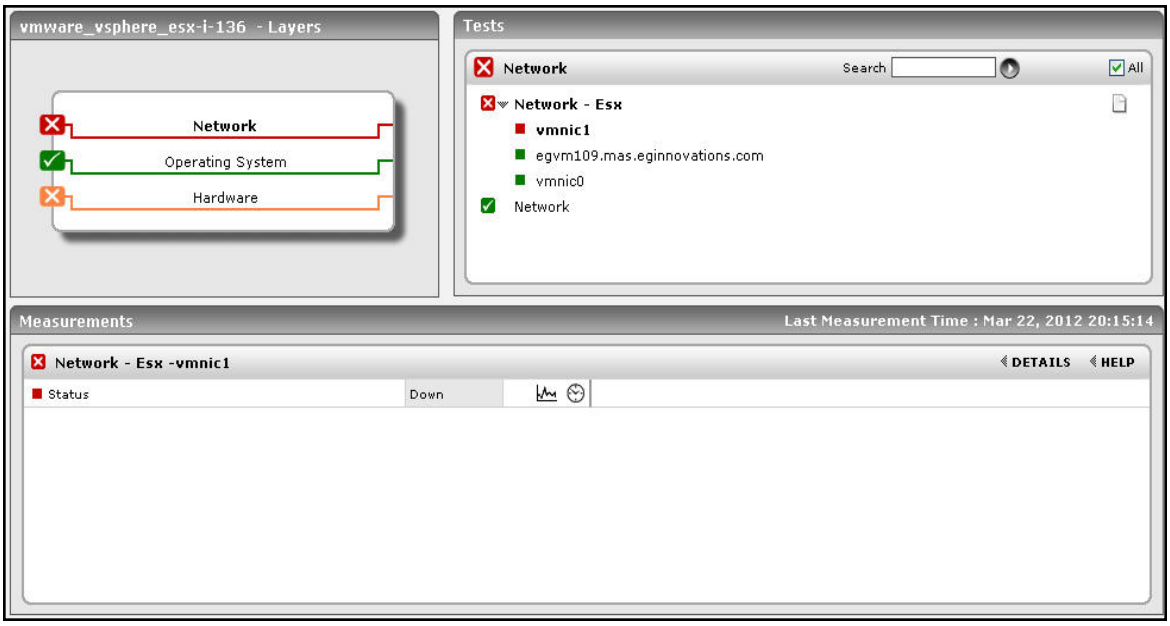


Figure 7.76: The problem layer, test, and measure of the problem component

7.7 The Resource Pools Tab Page

If resource pools have been configured on a virtual host, then the eG agent auto-discovers these pools and displays them as sub-nodes of a virtual host-node. The **Resource Pools** tab page appears only when a resource pool in the tree is clicked on (see Figure 7.77). This tab page reveals the current configuration of the chosen resource pool, which includes the number of virtual machines on the resource pool, the number of running virtual machines, and the number of child resource pools. Besides the configuration, administrators can also use this tab page to determine the current state of each of the virtual machines and child resource pools under the chosen resource pool, and simultaneously analyze the resource usage by the pool.

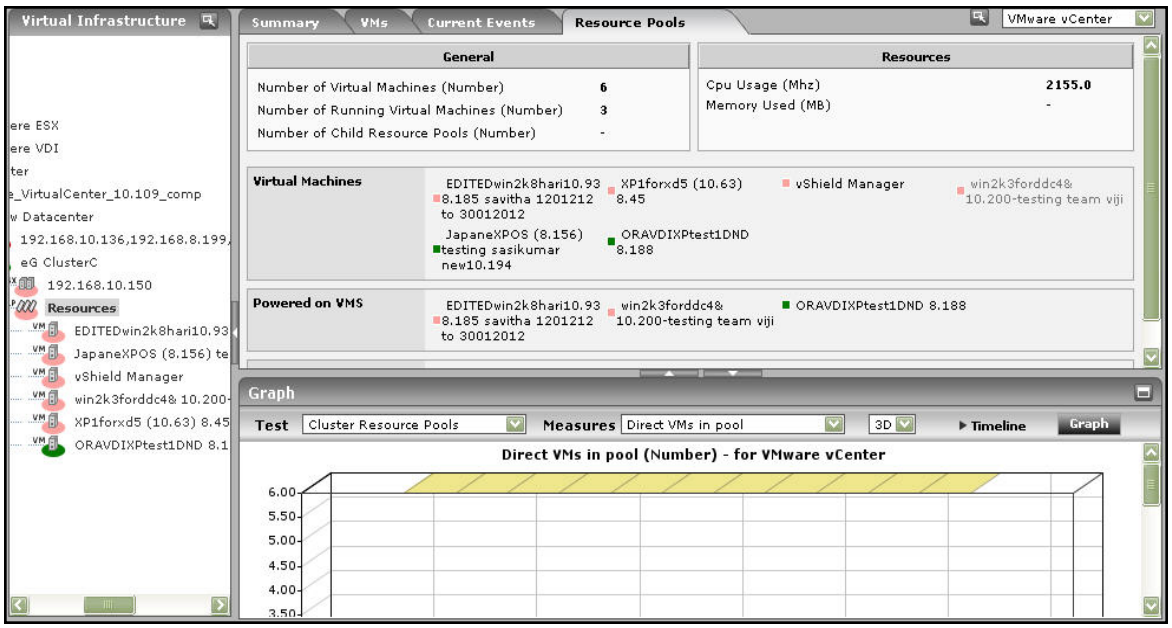


Figure 7.77: The Resource Pools tab page

Clicking on a VM listed against the **Virtual Machines** section of a resource pool leads administrator directly to the layer model page of the virtual host on which that resource pool is configured, and automatically displays metrics that provide an "outside view" of that VM's performance (see Figure 7.78). These metrics help administrators in understanding the impact the VM has on the physical resources of the virtual host.

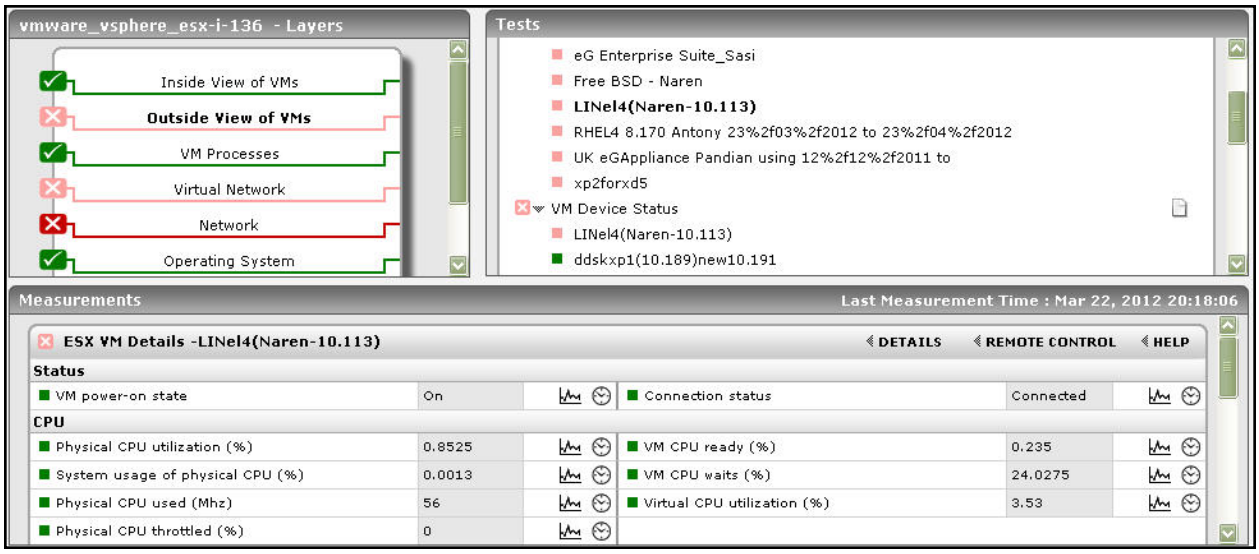


Figure 7.78: The outside view of a VM under the resource pool

Similarly, clicking on a particular child resource pool displayed against the **Child Resource Pools** section of Figure 7.77, takes the administrator to the layer model page of the corresponding virtual host, thereby granting him a sneak peek at the resource usage metrics of that child resource pool, and enabling him to analyze how resource-intensive the child is (see Figure 7.79).

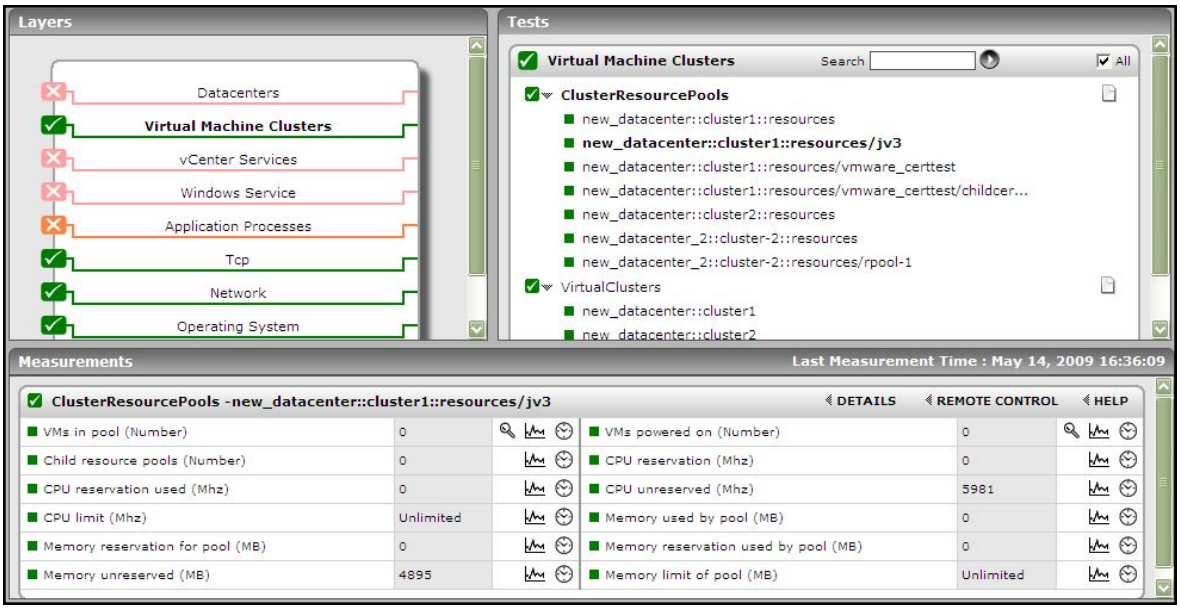


Figure 7.79: The metrics related to the child resource pool

8

Frequently Asked Questions (FAQ)

1. **The virtualization management space is becoming somewhat crowded lately, with a number of companies either migrating their management solution over to support virtualization or new companies starting up to try and answer the management issue. How does the eG VMware Monitor differ from other products in the virtualization space?**

   Many solutions in the virtualization space focus on monitoring the virtual environment as a "silo". These solutions extract metrics using instrumentation/APIs available in VMware ESX and vCenter, and even though these metrics may be incorporated into a common console where metrics for physical servers, networks, and applications in the infrastructure are also available, there is little or no correlation between the virtual infrastructure's performance and the performance of the applications that rely on the virtual infrastructure. The eG Enterprise suite also relies on the in-built VMware instrumentation for insights into the VMware infrastructure. However, it differs from other VM monitoring solutions in several key ways:

   - Firstly, the ESX infrastructure has only an "outside" view of each of the VMs supported by it. The VM Kernel does not know the details of what is going on within a VM - e.g., if the CPU usage of a VM is high, is that because the workload is high, or because there is a single run-away process that is taking up all the CPU resources? To effectively monitor a virtual infrastructure, it is essential for a monitoring system to have an "inside" view of each VM - to know what is happening within the VM. eG Enterprise includes a patented In-N-Out monitoring technology wherein a single agent is able to monitor the ESX server, the service console, and provide both an inside and an outside view of each of the VMs.
   - Each of the metrics collected by the eG agents can be automatically baselined - past history is used to determine time-varying norms for the metrics, and alerts triggered when a bound is violated. By avoiding the need to set manual thresholds for thousands of metrics, auto-baselining as supported by eG Enterprise can save hundreds of man hours of consulting time. This capability is also critical for an IT infrastructure to be managed proactively - so alerts are generated before users complain.
   - Most monitoring solutions monitor virtual environments as silos. In reality, the virtual environment is an integral part of an end-to-end business service. For example, a banking application may include a web server front-end, a middleware application server, and a database backend. The database server may be running on a VM that is hosted on an ESX server. If another VM on the same ESX server has an I/O intensive application (e.g., a media server) running on it, the I/O intensive application may impact the performance of the database server. Inter-dependencies in a virtual environment (which applications are running on VMs that are co-hosted on the same ESX server) play an important role in the performance of the end-to-end business service. eG Enterprise analyzes the performance of the virtual infrastructure in the context of the business services that it supports. Inter-application dependencies as well as inter-VM dependencies are taken into account when eG Enterprise diagnoses where the root-cause of a problem with a business service lies.

2. **Usually you hear things like "just let VMware's DRS, HA and VMotion handle the**

resources and move the virtual machines around for you to keep them up and operational." Why then do you find it important to track VMotion server moves?

VMware DRS and VMotion are very good technologies that enable an adaptive, dynamic environment. However, the dynamic nature of the virtual environment makes proper monitoring and management a challenge. For example, a database server may be on ESX server A at one instant of time and on ESX server B at another instant of time. Hence, dependency mappings used for root-cause diagnosis (as discussed in the first question) are no longer static. The monitoring system needs to be able to dynamically determine which VM is running on which ESX server.

VMotion/DRS are based on pre-specified criteria - if these criteria are set too aggressively, this may result in unnecessary movement of virtual machines across ESX servers. Such situations need to be monitored and remedied.

One of the key questions that comes up with VMotion/DRS is - why did a VM move? Since it tracks what is happening at every layer of the virtual infrastructure, eG Enterprise has the information that administrators can use to answer this question - I.e., what happened on the ESX server prior to the VM movement - what triggered this?

3. **What are some of the challenges with monitoring VMware infrastructures?**

VMware technologies and virtualization in general aim to make better use of existing hardware resources, by running multiple VMs on the same hardware. Any time there is resource sharing, there are also issues of resource contention. In a VMware environment, the performance of a VM may be impacted by another VM running on the same physical server. These dependencies have to be taken into account when monitoring and managing this infrastructure. The use of VMotion and DRS makes the problem even harder - since VMs could be moving from one ESX server to another.

One of the benefits of virtualization is the speed with which new VMs can be provisioned to address the business imperatives of flexibility and rapid response to changing service requirements. This of course also carries the widely discussed risk of server sprawl, with its potential for loss of control and oversight. In these times of strict corporate governance and regulatory compliance, the monitoring solution needs to give management absolute confidence that they are in control and are monitoring the overall infrastructure in an appropriate manner.

4. **Can you tell us more about the information you are gathering from the virtual machines themselves? Are agents involved or not?**

Knowing what is happening within the VM is key to effective root-cause diagnosis and performance management in a virtual infrastructure. Recognizing this, eG Enterprise has incorporated a patented In-N-Out monitoring technology that includes in-depth insights into what is happening with the VM guest, simply by installing a single agent on an ESX server. Without requiring an agent on every guest VM,

eG Enterprise is able to determine what processes are running inside a VM, which are the top resource consuming processes, which user is logged on to the guest VM, etc.

5. **Is your solution specific to VMware only or does it support other applications?**

eG Enterprise offers monitoring of the virtual as well as the physical environment. Support for over 150+ applications (Citrix, terminal services, SAP, Active Directory, SQL, Oracle, Sun ONE, and many others) and 10+ operating systems (Windows, Linux, Solaris, HPUX, AIX, etc.) is available in the eG Enterprise suite - see http://www.eginnovations.com/web/platforms.htm for details.

6. **Are you supporting vSphere/ESXi?**

   Yes; eG Enterprise provides monitoring support to vSphere/ESXi. Whereas ESX 3 and 3.5 have the service console where an agent can be installed, vSphere/ESXi does not have the service console. Hence, eG Enterprise provides a completely agentless monitoring solution for vSphere/ESXi. This solution has almost all the capabilities of the local agent in ESX 3 / 3.5.

7. **How accurate would your "inside view" be, considering the fact that the software clock that the VMs support is different from that of the physical ESX server?**

   VMs are known to suffer from clock skews and hence, metrics collected from within a VM have to be analyzed with care. Rather than use metrics collected from within a VM in absolute terms, eG Enterprise uses the inside view to compare the resource usage levels across applications running in the VMs. Other key indicators of VM and application bottlenecks such as memory leaks, handle leaks, disk partitions filling up, etc., can also be obtained using the inside view of a VM.

8. **Do you plan to support other virtualization technologies?**

   Yes - our goal is to be the leading provider of virtualization management software - and towards this end, we are already supporting Citrix Xen, Microsoft Hyper-V, Solaris LDom, Solaris Containers, and AIX LPAR technology as well.

9. **How is the product priced?**

   eG Enterprise is priced per VMware ESX server being monitored - i.e., the licensing is not per CPU, per socket, or per VM. This provides administrators insights into the performance of the VM kernel, the service console (if applicable), and the individual VMs and the applicationsthat run on these VMs.

10. **Does the product have a web-based interface?**

    Yes - eG Enterprise follows a 100% web-based architecture - the agents communicate with the management console using the web protocols (HTTP/HTTPS) and the only way to access the management console to view alerts, pop reports, etc. is using the web browser.