Monitoring Microsoft Hyper-V

eG Innovations Product Documentation

www.eginnovations.com

eG
Total Performance Visibility

Table of Contents

Table of Figures

Chapter 1: Introduction

Microsoft Hyper-V Server provides a simplified, reliable, and optimized virtualization solution, enabling improved server utilization and reduced costs.

Hyper-V is a hypervisor-based virtualization technology for x64 versions of Windows Server 2008. The hypervisor is the processor-specific virtualization platform that allows multiple isolated operating systems to share a single hardware platform.

Hyper-V supports isolation in terms of a partition. A partition is a logical unit of isolation, supported by the hypervisor, in which operating systems execute. The Microsoft hypervisor must have at least one parent, or root partition, running Windows Server 2008 64-bit Edition. The virtualization stack runs in the parent partition and has direct access to the hardware devices. The root partition then creates the child partitions which host the guest operating systems. A root partition creates child partitions using the hypercall application programming interface (API).

Partitions do not have access to the physical processor, nor do they handle the processor interrupts. Instead, they have a virtual view of the processor and run in a virtual memory address region that is private to each guest partition. The hypervisor handles the interrupts to the processor, and redirects them to the respective partition. Hyper-V can also hardware accelerate the address translation between various guest virtual address spaces by using an Input Output Memory Management Unit (IOMMU) which operates independent of the memory management hardware used by the CPU. An IOMMU is used to remap physical memory addresses to the addresses that are used by the child partitions.

Child partitions also do not have direct access to other hardware resources and are presented a virtual view of the resources, as virtual devices (VDevs). Requests to the virtual devices are redirected either via the VMBus or the hypervisor to the devices in the parent partition, which handles the requests. The VMBus is a logical inter-partition communication channel. The parent partition hosts Virtualization Service Providers (VSPs) which communicate over the VMBus to handle device access requests from child partitions. Child partitions host Virtualization Service Consumers (VSCs) which redirect device requests to VSPs in the parent partition via the VMBus. This entire process is transparent to the guest operating system.

Virtual Devices can also take advantage of a Windows Server Virtualization feature, named Enlightened I/O, for storage, networking, graphics, and input subsystems. Enlightened I/O is a specialized virtualization-aware implementation of high level communication protocols (such as SCSI) that utilize the VMBus directly, bypassing any device emulation layer. This makes the communication more efficient but requires an enlightened guest that is hypervisor and VMBus aware. Hyper-V enlightened I/O and a hypervisor aware kernel is provided via installation of Hyper-V integration services. Integration components, which include virtual server client (VSC) drivers, are also available for other client operating systems. Hyper- V requires a processor that includes hardware assisted virtualization, such as is provided with Intel VT or AMD Virtualization (AMD-V) technology.

Figure 1.1 provides a high-level overview of the architecture of a Hyper-V environment.

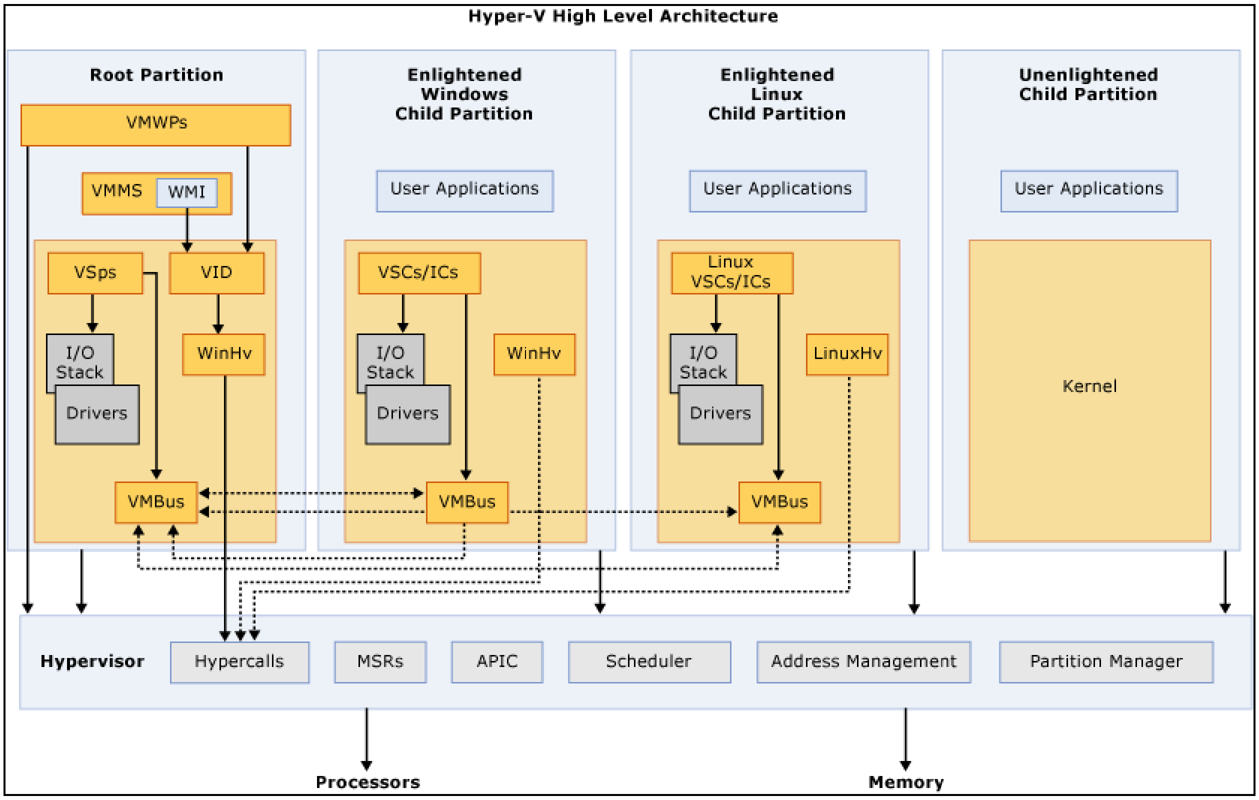


Figure 1.1: A high-level overview of the Hyper-V architecture

Acronyms and terms used in the diagram above are described below:

1. APIC – Advanced Programmable Interrupt Controller – A device which allows priority levels to be assigned to its interrupt outputs.
2. Child Partition – Partition that hosts a guest operating system - All access to physical memory and devices by a child partition is provided via the Virtual Machine Bus (VMBus) or the hypervisor.

3. Hypercall – Interface for communication with the hypervisor - The hypercall interface accommodates access to the optimizations provided by the hypervisor.
4. Hypervisor – A layer of software that sits between the hardware and one or more operating systems. Its primary job is to provide isolated execution environments called partitions. The hypervisor controls and arbitrates access to the underlying hardware.
5. IC – Integration component – Component that allows child partitions to communication with other partitions and the hypervisor.
6. I/O stack – Input/output stack
7. MSR – Memory Service Routine
8. Root Partition – Manages machine-level functions such as device drivers, power management, and device hot addition/removal. The root (or parent) partition is the only partition that has direct access to physical memory and devices.
9. VID – Virtualization Infrastructure Driver – Provides partition management services, virtual processor management services, and memory management services for partitions.
10. VMBus – Channel-based communication mechanism used for inter-partition communication and device enumeration on systems with multiple active virtualized partitions. The VMBus is installed with Hyper-V Integration Services.
11. VMMS – Virtual Machine Management Service – Responsible for managing the state of all virtual machines in child partitions.
12. VMWP – Virtual Machine Worker Process – A user mode component of the virtualization stack. The worker process provides virtual machine management services from the Windows Server 2008 instance in the parent partition to the guest operating systems in the child partitions. The Virtual Machine Management Service spawns a separate worker process for each running virtual machine.
13. VSC – Virtualization Service Client – A synthetic device instance that resides in a child partition. VSCs utilize hardware resources that are provided by Virtualization Service Providers (VSPs) in the parent partition. They communicate with the corresponding VSPs in the parent partition over the VMBus to satisfy a child partitions device I/O requests.
14. VSP – Virtualization Service Provider – Resides in the root partition and provide synthetic device support to child partitions over the Virtual Machine Bus (VMBus).
15. WinHv – Windows Hypervisor Interface Library - WinHv is essentially a bridge between a partitioned operating system's drivers and the hypervisor which allows drivers to call the hypervisor using standard Windows calling conventions

16. WMI – The Virtual Machine Management Service exposes a set of Windows Management Instrumentation (WMI)-based APIs for managing and controlling virtual machines.

Since many “real” infrastructures these days are rapidly becoming “virtual”, a simple-to-use, cost-effective, resource-thin solution like Hyper-V is gaining immense popularity. Moreover, VMs configured on Hyper-V can run complex server applications and/or simple desktop applications. This implies that critical business services can be delivered using virtual applications deployed on Hyper-V, and virtual desktops can be published on Hyper-V so that users receive instant and easy access to their desktop applications from thin clients. Owing to these dependencies, a resource crunch experienced by the virtual environment can grossly affect the quality of the business service / user experience with Hyper-V. To avoid this, 24 x 7 monitoring of the resource usage by the Hyper-V host and VMs is imperative.

Chapter 2: How eG Enterprise Monitors Microsoft Hyper-V Servers?

eG Enterprise offers two specialized monitoring models – one each for each of the distinct deployment architectures of Hyper – V. While the generic *Hyper-V* model is to be used for monitoring Hyper-V servers with VMs hosting server applications, the *Hyper-V VDI* model is ideal for virtual desktop environments.

Regardless of the model being used, eG Enterprise adopts a patented *In-N-Out* approach to monitoring it. This approach enables administrators to monitor the Hyper-V server inside out and determine the following:

- The overall health of the Hyper-V host
- The physical resource usage by the Hyper-V host and host processes
- Whether critical Hyper-V services are available or not;
- The availability of the Hyper-V server
- The current status of the VMs configured on the host and the the fraction of physical resources used up by each VM, as seen from outside the VMs; this represents the “outside” view
- The fraction of allocated resources used up by each VM; this represents the “inside” view

The broad steps for monitoring the Microsoft Hyper-V/ Hyper-V VDI are as follows:

1. Deploy an eG agent on the target host. Refer to Section **2.1**
2. Fulfill the pre-requisites for monitoring the Hyper-V. Refer to Section **2.2**.
3. Manage the target Hyper-V using eG administrative interface.

2.1 Agent Deployment Model

Using eG Enterprise, administrators can manage a Hyper-V server in an agent-based manner.

The agent-based approach requires that an eG agent be installed on the root partition of the Hyper-V server. The root partition runs a Windows 2008 64-bit operating system. Therefore, to monitor the Hyper-V server, you need to install the Windows 2008 64-bit agent on the root partition. The steps for the installation are clearly laid out in the eG Installation Guide document. The agent then uses Perfmon to extract metrics from the Hyper-V host, auto-discovers the IP addresses of the guests on the host, communicates with every guest via WMI, and then collects the “inside view” metrics using Perfmon (see Figure 2.1).

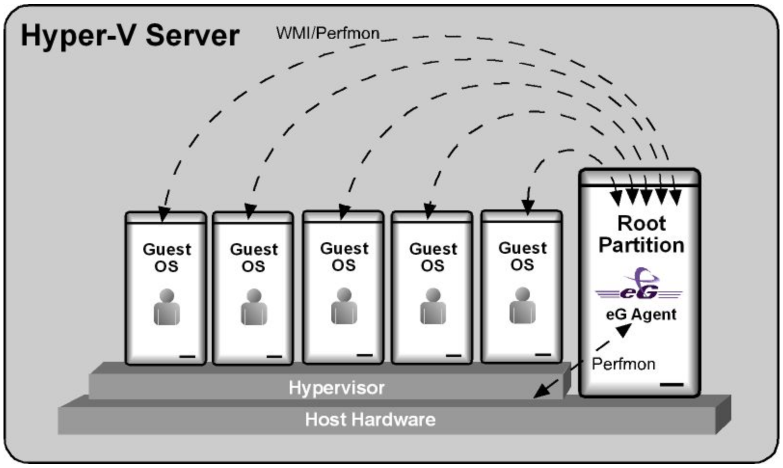


Figure 2.1: Agent-based monitoring of Hyper-V

Note:

The eG agent can collect “inside view” metrics from Windows VMs alone; for Linux VMs, only the powered-on status and “outside view” metrics will be available.

The eG agent then communicates remotely with every Windows VM on the target Hyper-V host (using WMI) to obtain the inside view of the VMs. To establish this remote connection with Windows VMs, 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 to collect “inside view” metrics from the VMs **without domain administrator rights**. Refer to 2.2.2 for more details on the **eG VM Agent**.

For a detailed list of pre-requisites for monitoring Hyper-V, refer to 2.2.

2.2 Pre-requisites for Monitoring Microsoft Hyper-V

There are several pre-requisites for an eG agent to be able to monitor a Hyper-V server and the guest VMs hosted on it.

1. The eG agent on the root partition should be able to communicate with the eG manager port (default is 7077).

2. The **Integration Services** component should be installed on every VM to be monitored, so that the IP address of the VMs is discovered; also, the IP address should be resolvable in DNS.

   Note:

   - The eG agent can automatically discover the IP address of the Windows VMs only; for Linux VMs, only the name of the VM will be discovered and not its IP address.
   - If multiple IP addresses are configured on a single Windows VM, the eG agent will discover only one of the IP addresses and not all of them.
   - The **Integration Services** component provides a **Key/Value Pair Exchange** script, which the eG agent uses for discovering the IP addresses of the VMs on the Hyper-V host. Since this script is supported only on specific Windows versions, the eG agent can discover the IP address of the VMs executing on those versions only. The supported Windows versions are as follows:
     - Windows Server 2008 64-bit
     - Windows Server 2008 x86
     - Windows Server 2003 x64 with SP2
     - Windows 2000 Server with SP4
     - Windows 2000 Advanced Server SP4
     - Windows Vista x64 with SP1
     - Windows Vista x86 with SP1
     - Windows XP x86 with SP2/SP3
     - Windows XP x64 with SP2

3. To obtain the “inside view” of Windows VMs **without using the ‘eG VM Agent’**, the following will have to be performed:

   - The **ADMIN$** share should be enabled for all Windows virtual guests being monitored and the administrative account must have permissions to this share drive. Refer to for a step-by-step procedure to achieve this.
   - All “inside view” tests run by the eG agent should be configured with the credentials of a domain administrator.
   - Set the **INSIDE VIEW USING** flag for all the “inside view” tests to **Remote connection to VM (Windows)**.

4. To obtain the “inside view” of Windows VMs **using the ‘eG VM Agent’**, follow the steps given below:

- Install the eG VM Agent on every Windows VM to be monitored; the procedure for installing the eG VM Agent are detailed in 2.2.2 of this document.
- Enable the eG agent to communicate with the port at which the eG VM Agent listens (the default port is 60001).
- Set the **INSIDE VIEW USING** flag for all the “inside view” tests to **eG VM Agent (Windows)**.

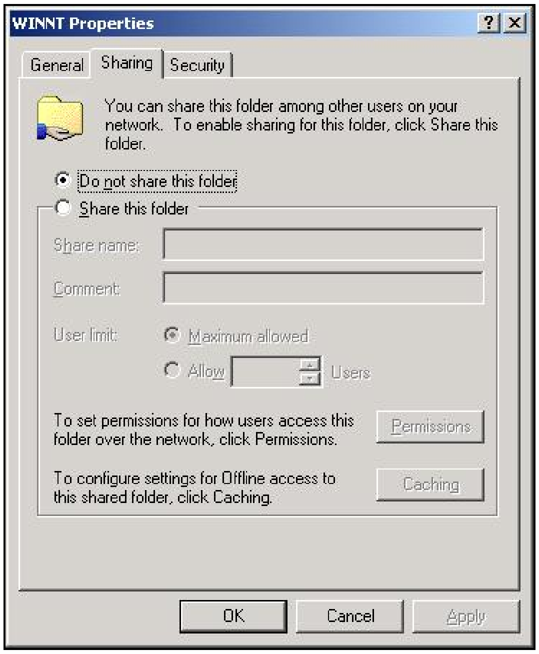
2.2.1 Configuring Windows Virtual Machines to Support the Inside View without the eG VM Agent

For the "inside" view, by default, the eG agent communicates remotely with the virtual machines on the Hyper-V server and collects metrics. To establish this remote connection with Windows VMs, eG Enterprise requires that the eG remote agent (on Windows) 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 (Windows)**.

In addition, the **ADMIN$** share will have to be available on the Windows guests.

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 2.2 appears indicating the same.



You can share this folder among other users on your network. To enable sharing for this folder, click Share this folder.

● Do not share this folder

○ Share this folder

Share name:

Comment:

User limit: ● Maximum allowed

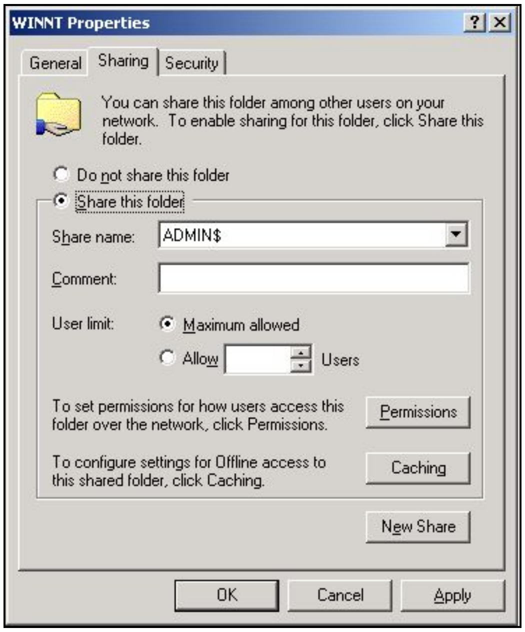
○ Allow Users

To set permissions for how users access this folder over the network, click Permissions.

To configure settings for Offline access to this shared folder, click Caching.

Figure 2.2: The ADMIN$ share does not exist

3. On the other hand, if the **ADMIN$** share pre-exists, Figure 2.3 appears. In such a case, first, remove the **ADMIN$** share by selecting the **Do not share this folder** option from Figure 2.3 and clicking the **Apply** and **OK** buttons. After this, you will have to repeat step 1 of this procedure to open Figure 2.2. Then, proceed as indicated by step 3 onwards.



You can share this folder among other users on your network. To enable sharing for this folder, click Share this folder.

○ Do not share this folder

● Share this folder

Share name: ADMIN$

Comment:

User limit: ● Maximum allowed

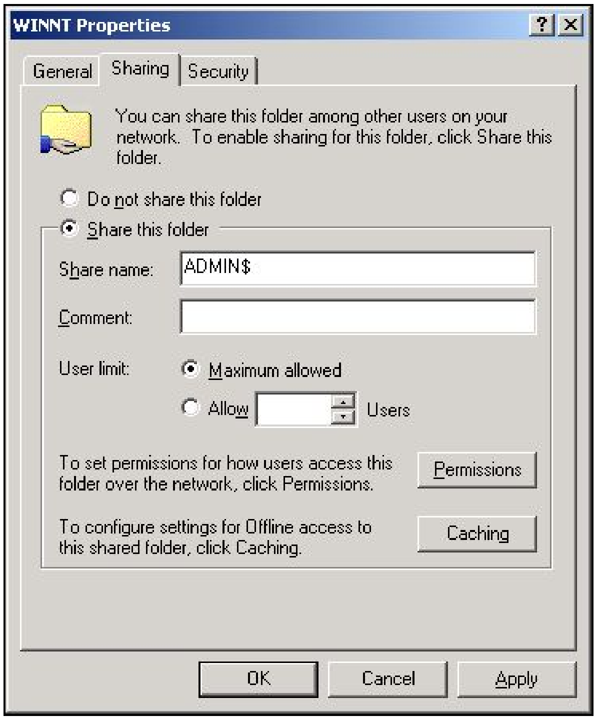
○ Allow Users

To set permissions for how users access this folder over the network, click Permissions.

To configure settings for Offline access to this shared folder, click Caching.

Figure 2.3: Admin$ share pre-exists

4. To create (or re-create) the **ADMIN$** share, select the **Share this folder** option from Figure 2.3, and provide **ADMIN$** share against the **Share name** text box (see Figure 2.4).



You can share this folder among other users on your network. To enable sharing for this folder, click Share this folder.

○ Do not share this folder

● Share this folder

Share name: ADMIN$

Comment:

User limit: ● Maximum allowed

○ Allow Users

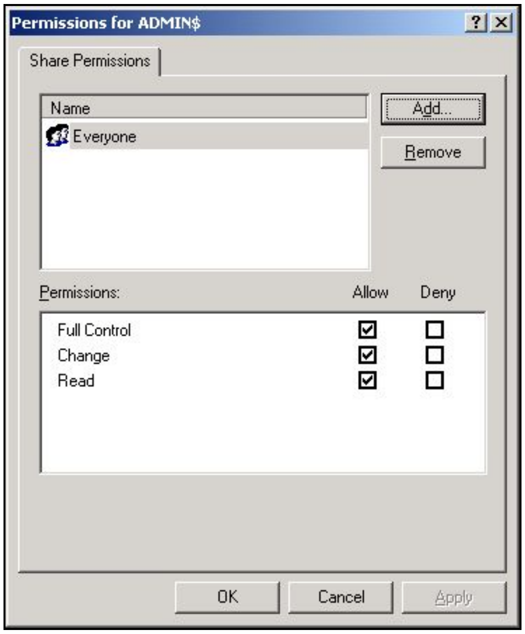
To set permissions for how users access this folder over the network, click Permissions.

To configure settings for Offline access to this shared folder, click Caching.

Figure 2.4: 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 (domain); also, the **credentials of this user should be passed while configuring the eG monitoring capabilities** - i.e., while configuring the Hyper-V tests. To grant the access permissions, click on the **Permissions** button in Figure 2.4.

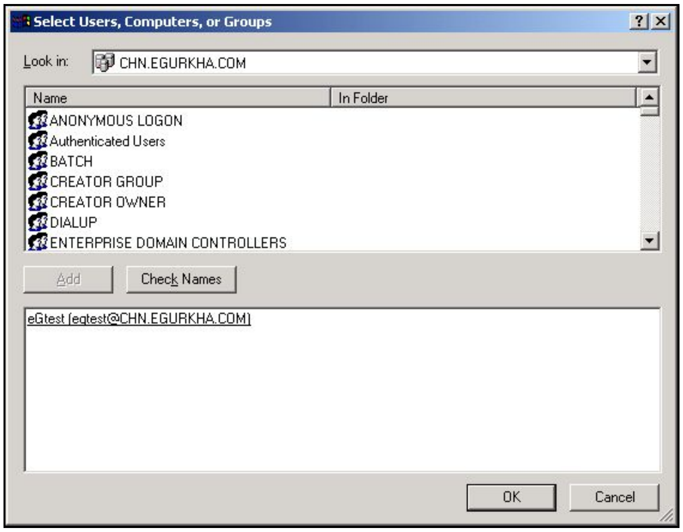
6. By default, the **ADMIN$** share can be accessed by **Everyone** (see Figure 2.5). To grant access rights to a specific domain administratoar, select the **Add** button in Figure 2.5. When Figure 2.6 appears, select the domain to search from the **Look in** list. The valid user accounts configured 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.



Name
Everyone

Permissions:	Allow	Deny
Full Control	☑	□
Change	☑	□
Read	☑	□

Figure 2.5: Clicking the Add button



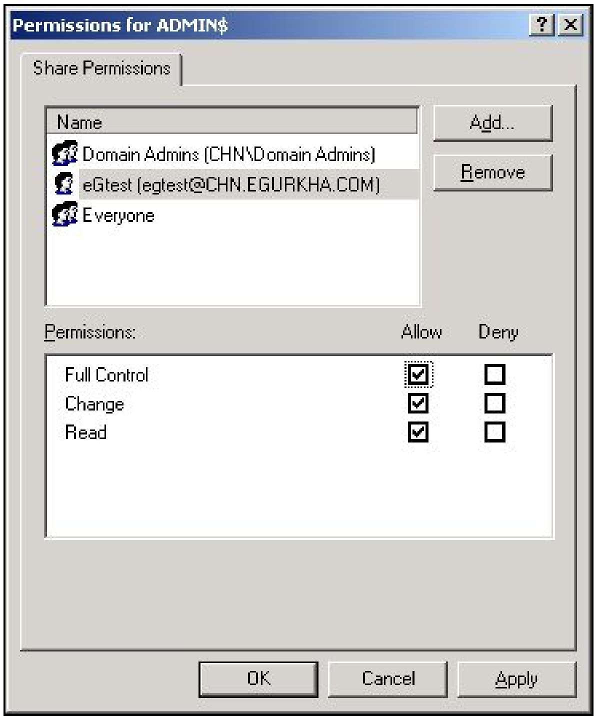
Look in: CHN.EGURKHA.COM

Name	In Folder
ANONYMOUS LOGON	
Authenticated Users	
BATCH	
CREATOR GROUP	
CREATOR OWNER	
DIALUP	
ENTERPRISE DOMAIN CONTROLLERS	

eGtest (egtest@CHN.EGURKHA.COM)

Figure 2.6: Selecting the administrative user to whom access rights are to be granted

7. Finally, click the **OK** button. You will then return to Figure 2.5, where the newly added administrator account will appear (see Figure 2.7).



Permissions for ADMIN$

Share Permissions

Name

- Domain Admins (CHN\Domain Admins)
- eGtest (egtest@CHN.EGURKHA.COM)
- Everyone

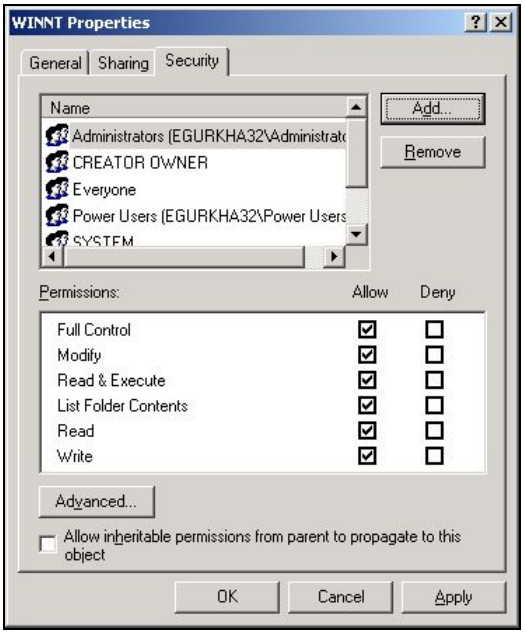
Permissions:	Allow	Deny
Full Control	☑	□
Change	☑	□
Read	☑	□

Figure 2.7: The administrator account granted access permissions

8. Select the newly added administrator account from Figure 2.7, 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 2.7 to register the changes.

10. Once you return to Figure 2.7, click on the **Security** tab (see Figure 2.8) to define the security settings for the **ADMIN$** share.

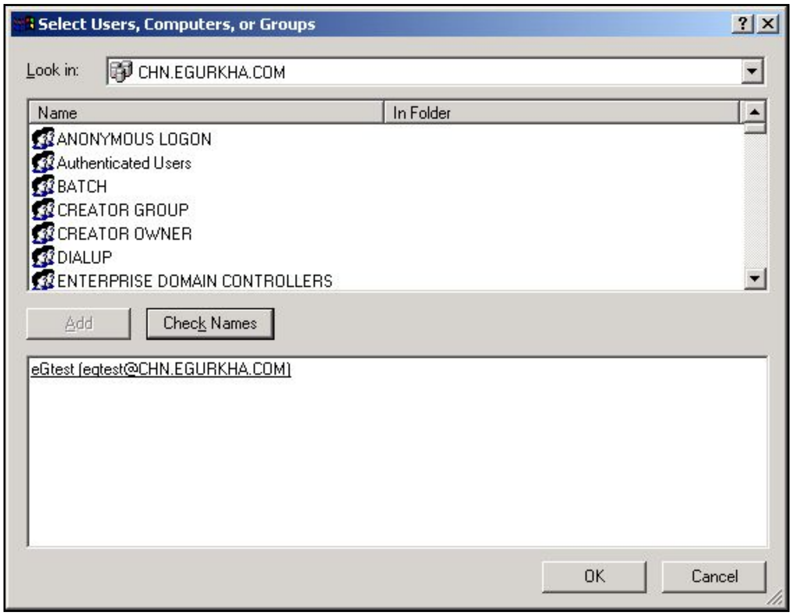


Permissions:	Allow	Deny
Full Control	☑	□
Modify	☑	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

□ Allow inheritable permissions from parent to propagate to this object

Figure 2.8: 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 2.8, pick a domain from the **Look in** list of Figure 2.9, select the said administrator account from the domain users list below, and click the **Add** button (in Figure 2.9) to add the chosen account. Then, click the **OK** button in Figure 2.9.

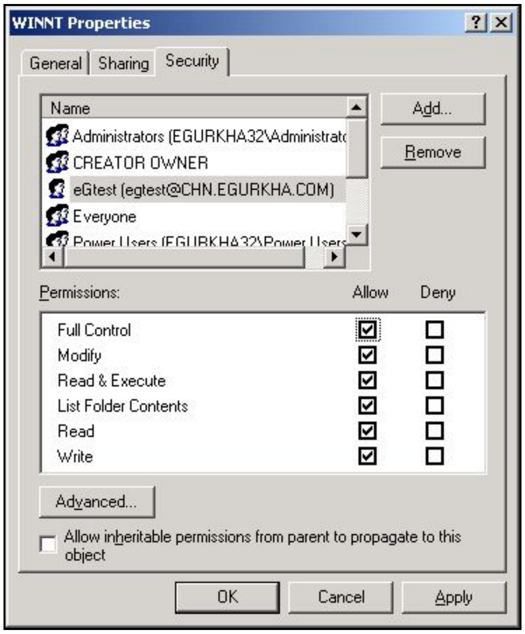


Look in: CHN.EGURKHA.COM

eGtest (egtest@CHN.EGURKHA.COM)

Figure 2.9: Adding the administrator account

12. This will bring you back to Figure 2.8, but this time, the newly added domain administrator account will be listed therein as indicated by Figure 2.10.



Name
- Administrators (EGURKHA32\Administrat
- CREATOR OWNER
- eGtest (egtest@CHN.EGURKHA.COM)
- Everyone
- Power Users (EGURKHA32\Power Users

Permissions:	Allow	Deny
Full Control	☑	□
Modify	☑	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

□ Allow inheritable permissions from parent to propagate to this object

Figure 2.10: The Administrator account in the Security list

13. Finally, click the **Apply** and **OK** buttons in Figure 2.10.

2.2.2 Configuring Windows Virtual Machines to Support theInside View Using the eG VM Agent

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.

2.2.3 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:

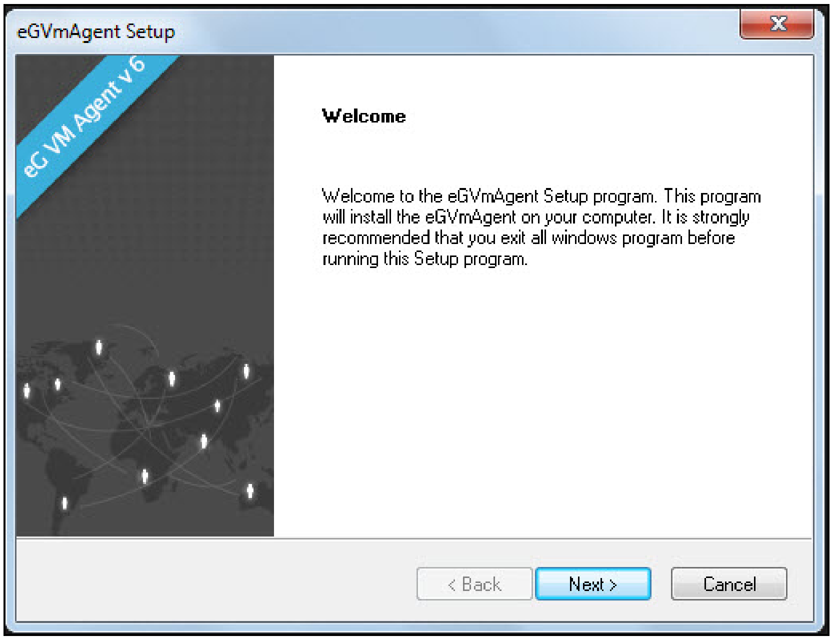
- Manually install the eG VM Agent on every Windows VM using the executable that eG Enterprise provides;
- Bundle the eG VM Agent as part of a template VM, and use this template to create multiple VMs; this way, the eG VM Agent is automatically available in all the VMs that are created using the template;
- 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. For the third option, refer to the *Installing eG VM Agent Using SCCM* document.

2.2.3.1 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 2.11 then appears. Click on the **Next** button in Figure 2.11 to continue.

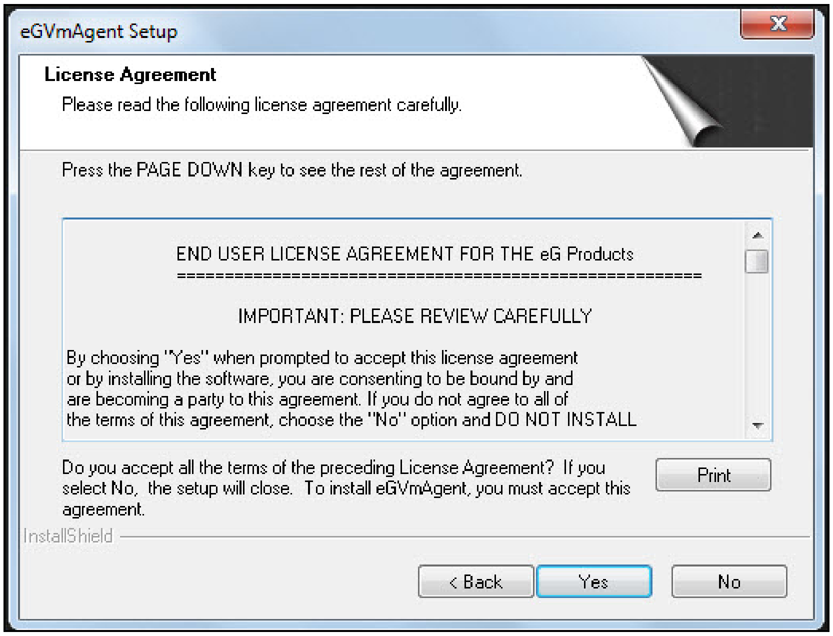


Welcome

Welcome to the eGVmAgent Setup program. This program will install the eGVmAgent on your computer. It is strongly recommended that you exit all windows program before running this Setup program.

Figure 2.11: Welcome screen of the eG VM Agent installation wizard

3. When Figure 2.12 appears, click on **Yes** to accept the displayed license agreement.



License Agreement

Please read the following license agreement carefully.

Press the PAGE DOWN key to see the rest of the agreement.

END USER LICENSE AGREEMENT FOR THE eG Products
==

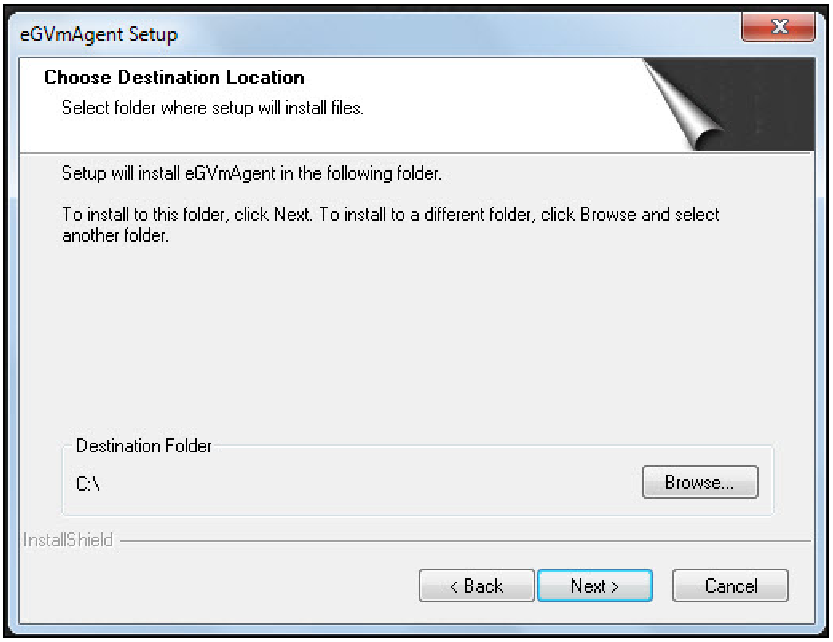
IMPORTANT: PLEASE REVIEW CAREFULLY

By choosing "Yes" when prompted to accept this license agreement or by installing the software, you are consenting to be bound by and are becoming a party to this agreement. If you do not agree to all of the terms of this agreement, choose the "No" option and DO NOT INSTALL

Do you accept all the terms of the preceding License Agreement? If you select No, the setup will close. To install eGVmAgent, you must accept this agreement.

Figure 2.12: Accepting the license agreement

4. Use the **Browse** button in Figure 2.13 to indicate the location in which the agent should be installed, and click the **Next** button to proceed.


eGVmAgent Setup

Choose Destination Location

Select folder where setup will install files.

Setup will install eGVmAgent in the following folder.

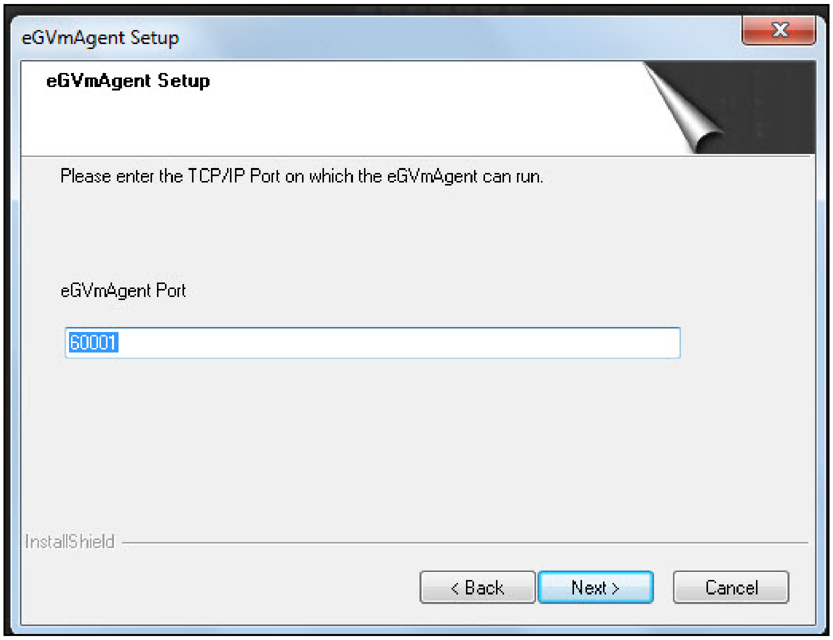
To install to this folder, click Next. To install to a different folder, click Browse and select another folder.

Destination Folder

C:\

Figure 2.13: 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 2.14 to proceed.


eGVmAgent Setup

eGVmAgent Setup

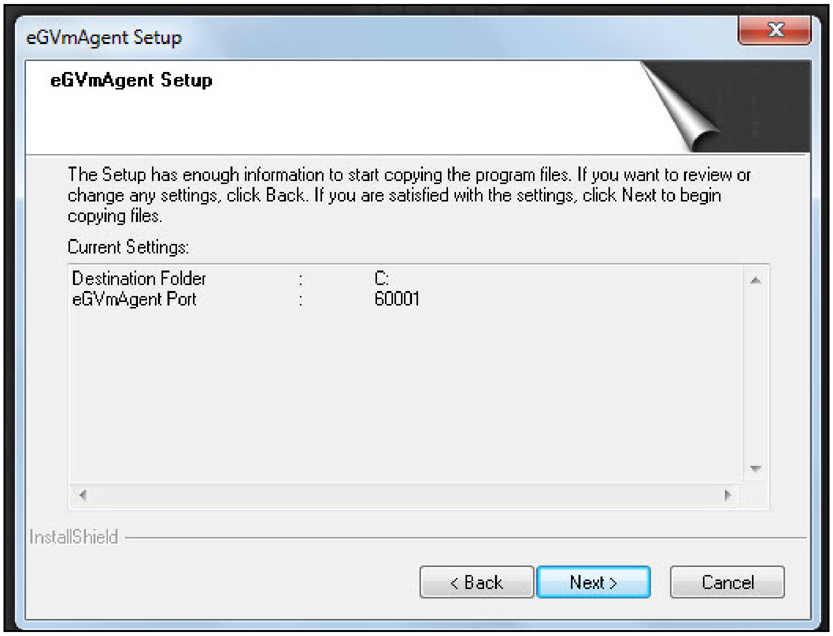
Please enter the TCP/IP Port on which the eGVmAgent can run.

eGVmAgent Port

60001

Figure 2.14: Specifying the VM agent port

6. A summary of your specifications then follows (see Figure 2.15). Click **Next** to proceed.



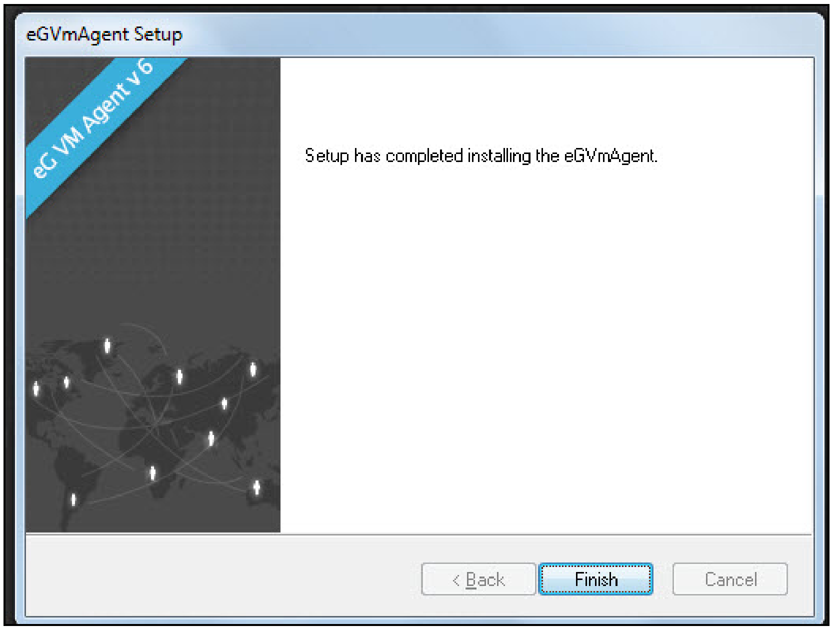
The Setup has enough information to start copying the program files. If you want to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files.

Current Settings:

```
Destination Folder        :        C:
eGVmAgent Port            :        60001
```

Figure 2.15: A summary of your specifications

7. Finally, click the **Finish** button in Figure 2.16 to complete the installation.



Setup has completed installing the eGVmAgent.

Figure 2.16: Finishing the installation

2.2.4 Silent Installation of the eG VM Agent

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

a. Creating silent mode script for eGVmagent installation

b. Installing eGVmAGent in silent mode

Each of these steps have been discussed elaborately below.

2.2.4.1 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.

2.2.4.2 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.

2.2.4.3 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.

The very first time the eG remote agent connects to a Windows VM to collect 'inside view' metrics from it, the eG VM agent on that VM downloads the inside view tests to be executed and the commands to be run for metrics collection. The eG VM agent then runs these commands, collects the metrics, and sends them to the remote agent. During this process, the eG VM Agent also compares the frequencies of all the inside view tests and picks the lowest frequency. At this frequency, the eG VM Agent then automatically runs the downloaded commands and stores the metrics collected in the cache. The next time the eG remote agent contacts the eG VM agent for metrics, it simply pulls the metrics stored in the cache. The eG remote agent then analyzes the metrics and informs the eG manager about the status of the Windows VMs.

2.2.5 Licensing and Benefits of the eG VM Agent

1. The eG VM Agent is not license-controlled. Therefore, you can install and use any number of VM agents in your infrastructure.
2. 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.

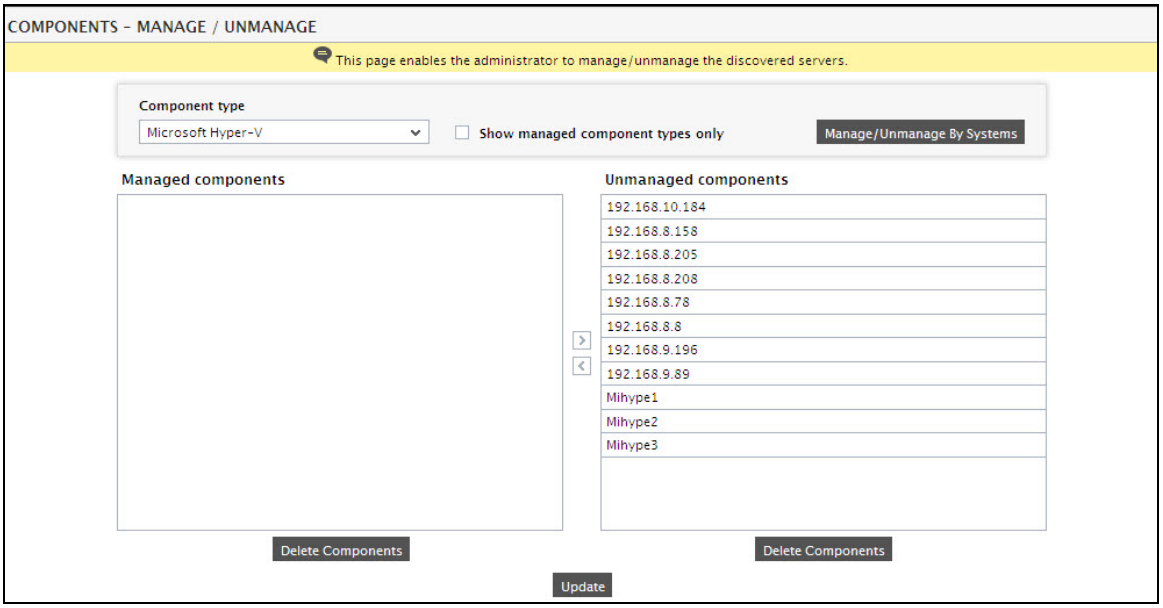
The chapters to come elaborately discuss the *Microsoft Hyper-V* and the *Microsoft Hyper-V VDI* models that eG Enterprise offers.

Chapter 3: Administering eG Enterprise to Monitor a Microsoft Hyper-V Server

To monitor a Microsoft Hyper-V / Hyper-V VDI server using eG, follow the steps below:

1. Log into the eG administrative interface.

2. The eG manager is capable of auto-discovering the Hyper–V / Hyper-V VDI server. If these servers are already discovered, then use the Infrastructure -> Components ->Manage/Unmanage menu to manage them. Otherwise run discovery process using the menu sequence: Infrastructure -> Components -> Discover and manage the Hyper-V servers as detailed in Figure 3.1 to Figure 3.2. While Figure 3.1 and Figure 3.2 depict how to manage an auto-discovered Hyper-V server, Figure 3.3 and Figure 3.4 show how to manage an Hyper-V VDI server using eG. As shown, first select the **Component type**, then pick the servers to be managed from the **Unmanaged Components** list, and click the **<** button to manage the servers. Finally, click the **Update** button.



COMPONENTS - MANAGE / UNMANAGE

This page enables the administrator to manage/unmanage the discovered servers.

Component type: Microsoft Hyper-V

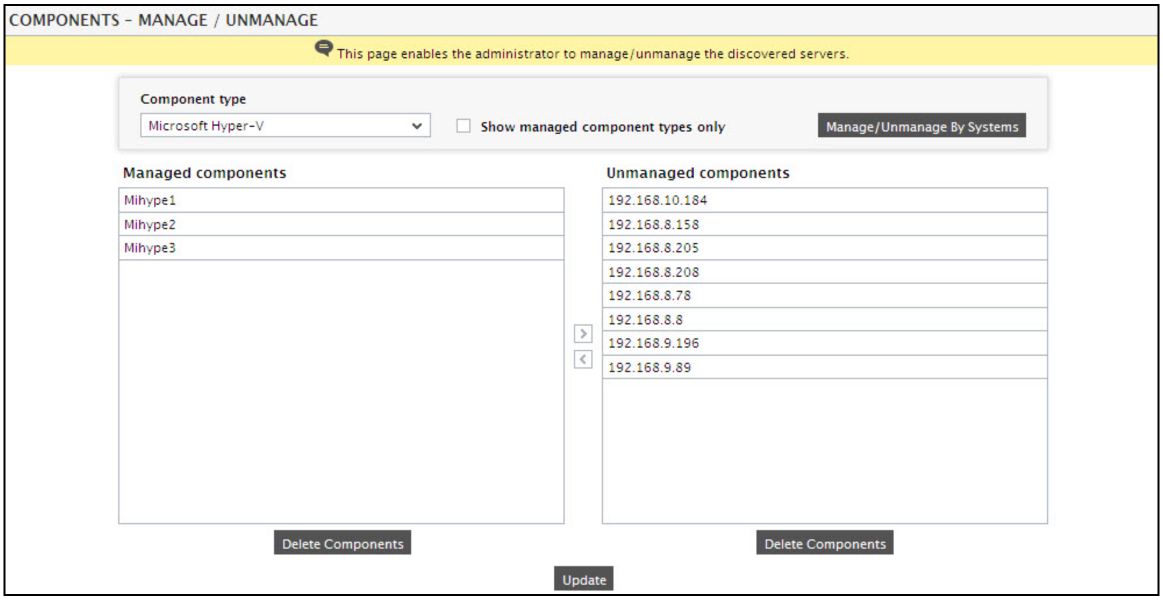
□ Show managed component types only

Managed components

Unmanaged components

192.168.10.184
192.168.8.158
192.168.8.205
192.168.8.208
192.168.8.78
192.168.8.8
192.168.9.196
192.168.9.89
Mihype1
Mihype2
Mihype3

Figure 3.1: Viewing the unmanaged Hyper-V Servers



COMPONENTS - MANAGE / UNMANAGE

This page enables the administrator to manage/unmanage the discovered servers.

Component type: Microsoft Hyper-V

□ Show managed component types only

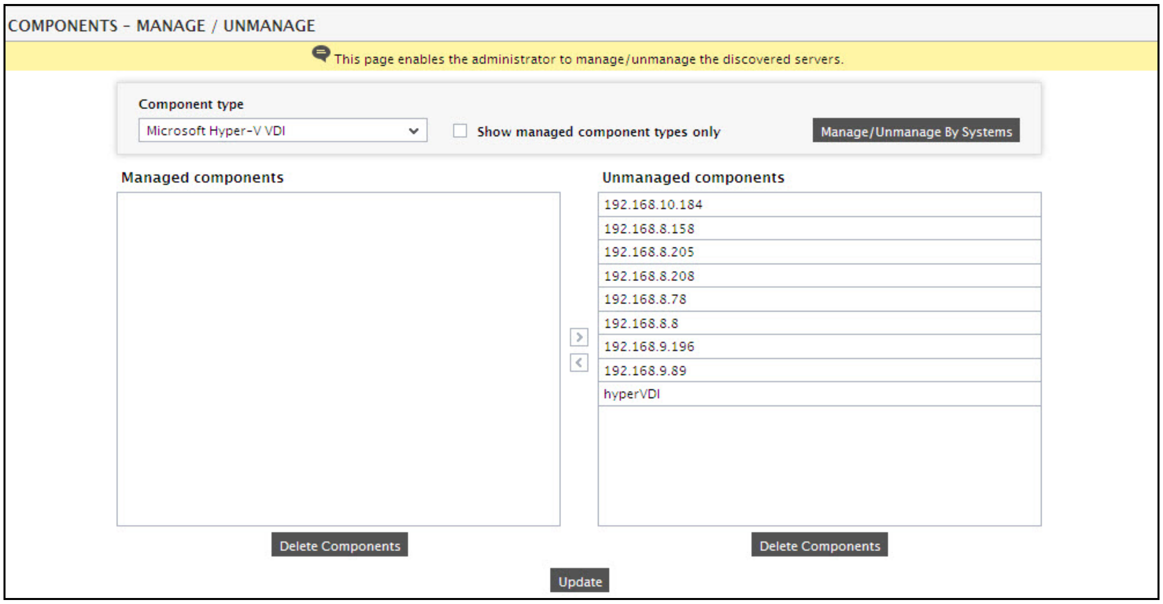
Managed components

- Mihype1
- Mihype2
- Mihype3

Unmanaged components

- 192.168.10.184
- 192.168.8.158
- 192.168.8.205
- 192.168.8.208
- 192.168.8.78
- 192.168.8.8
- 192.168.9.196
- 192.168.9.89

Figure 3.2: Managing the Hyper-V Servers



COMPONENTS - MANAGE / UNMANAGE

This page enables the administrator to manage/unmanage the discovered servers.

Component type: Microsoft Hyper-V VDI

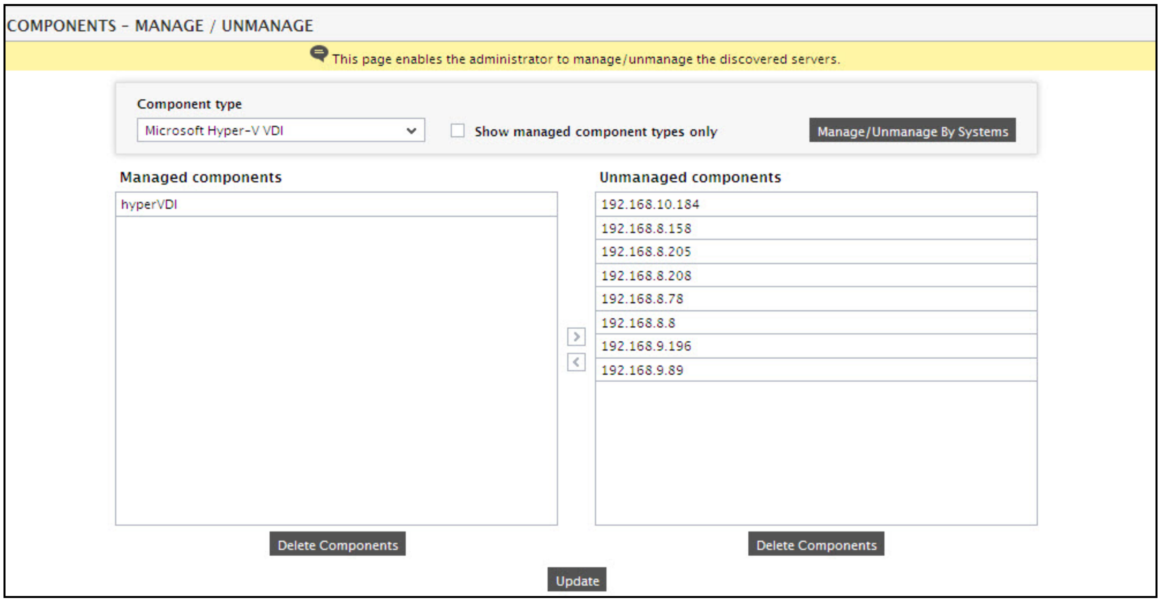
□ Show managed component types only

Managed components

Unmanaged components

- 192.168.10.184
- 192.168.8.158
- 192.168.8.205
- 192.168.8.208
- 192.168.8.78
- 192.168.8.8
- 192.168.9.196
- 192.168.9.89
- hyperVDI

Figure 3.3: Viewing the unmanaged Hyper-V VDI Servers



COMPONENTS - MANAGE / UNMANAGE

This page enables the administrator to manage/unmanage the discovered servers.

Component type: Microsoft Hyper-V VDI

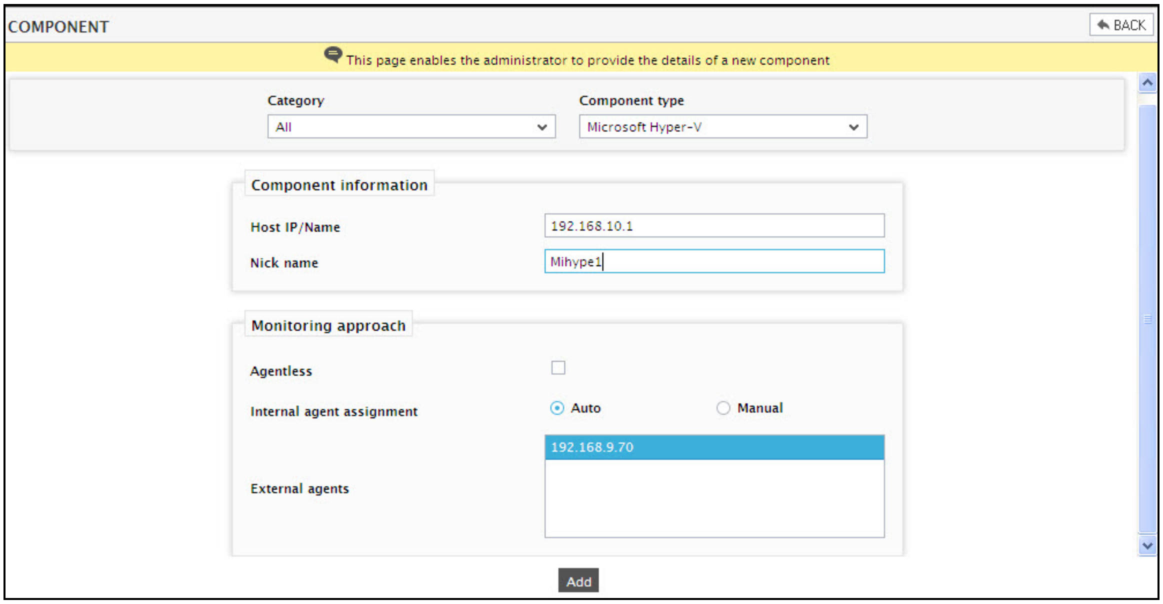
□ Show managed component types only

Managed components:
- hyperVDI

Unmanaged components:
- 192.168.10.184
- 192.168.8.158
- 192.168.8.205
- 192.168.8.208
- 192.168.8.78
- 192.168.8.8
- 192.168.9.196
- 192.168.9.89

Figure 3.4: Managing the Hyper-V VDI Servers

3. Alternatively, you can also manually add the target Hyper-V / Hyper- VDI servers using the **COMPONENTS** page (see Figure 3.5 and Figure 3.6). The components so added are automatically managed. To access this page, follow the Infrastructure -> Components -> Add/Modify menu sequence.



COMPONENT

This page enables the administrator to provide the details of a new component

Category: All

Component type: Microsoft Hyper-V

Component information

Host IP/Name: 192.168.10.1

Nick name: Mihype1

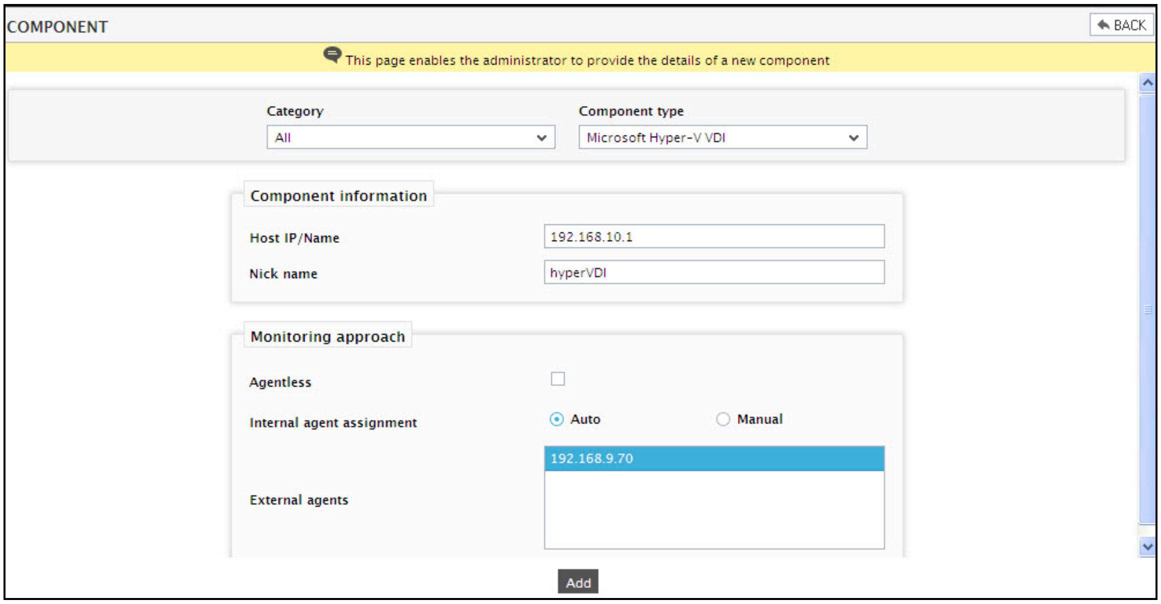
Monitoring approach

Agentless □

Internal agent assignment ● Auto ○ Manual

External agents: 192.168.9.70

Figure 3.5: Adding the Hyper-V Server



This page enables the administrator to provide the details of a new component

Category: All

Component type: Microsoft Hyper-V VDI

Component information

Host IP/Name: 192.168.10.1

Nick name: hyperVDI

Monitoring approach

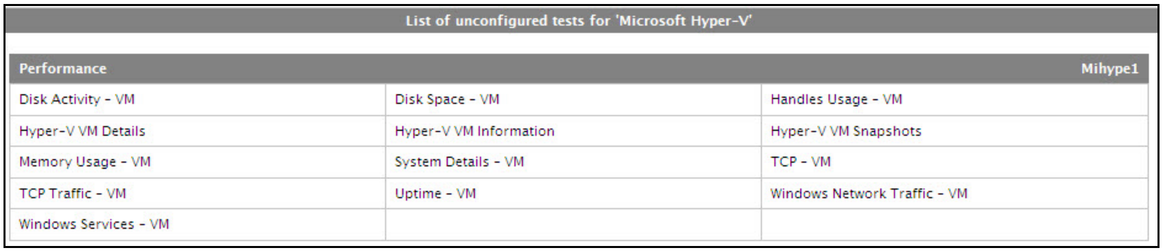
Agentless: □

Internal agent assignment: ● Auto ○ Manual

External agents: 192.168.9.70

Figure 3.6: Adding the Hyper-V VDI Server

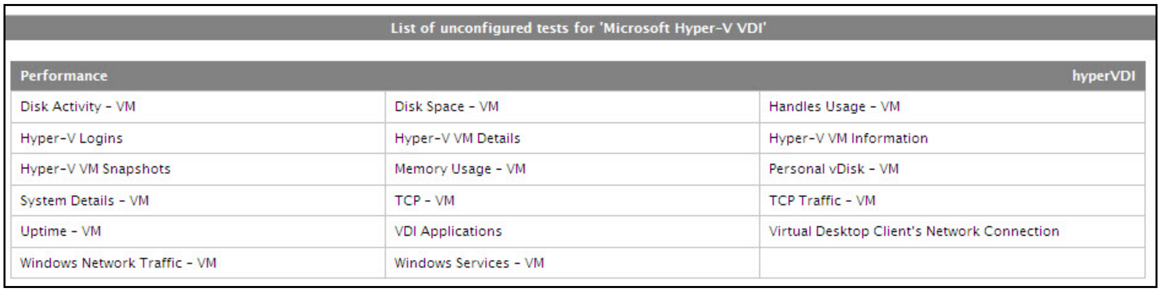
4. When you attempt to sign out, for a Hyper-V server, a list of unconfigured tests appears as in Figure 3.7. For a Hyper-V VDI server, a list of unconfigured tests.



List of unconfigured tests for 'Microsoft Hyper-V'

Performance		Mihype1
Disk Activity - VM	Disk Space - VM	Handles Usage - VM
Hyper-V VM Details	Hyper-V VM Information	Hyper-V VM Snapshots
Memory Usage - VM	System Details - VM	TCP - VM
TCP Traffic - VM	Uptime - VM	Windows Network Traffic - VM
Windows Services - VM		

Figure 3.7: List of tests to be configured for Hyper-V Server

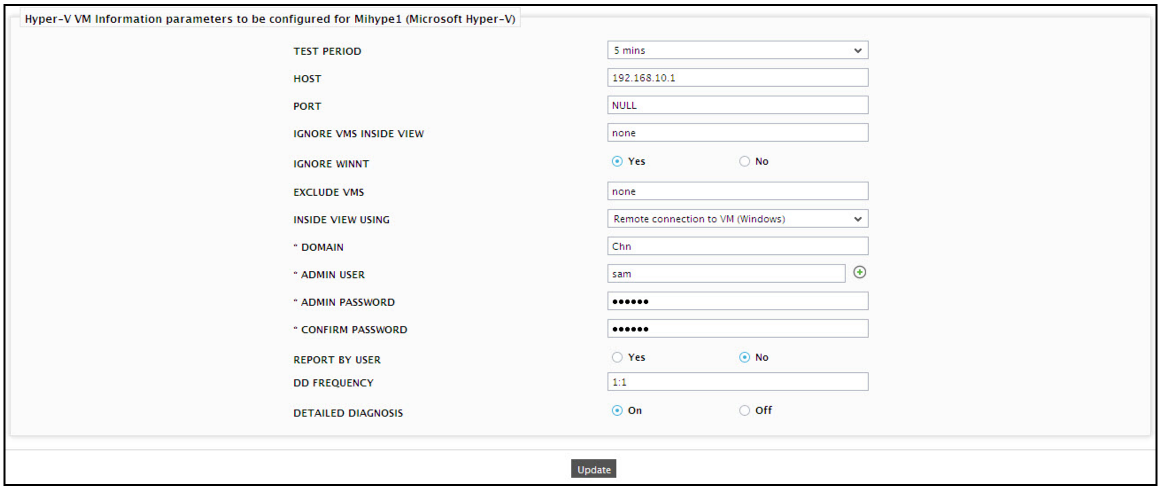


List of unconfigured tests for 'Microsoft Hyper-V VDI'

Performance		hyperVDI
Disk Activity - VM	Disk Space - VM	Handles Usage - VM
Hyper-V Logins	Hyper-V VM Details	Hyper-V VM Information
Hyper-V VM Snapshots	Memory Usage - VM	Personal vDisk - VM
System Details - VM	TCP - VM	TCP Traffic - VM
Uptime - VM	VDI Applications	Virtual Desktop Client's Network Connection
Windows Network Traffic - VM	Windows Services - VM	

Figure 3.8: List of tests to be configured for Hyper-V VDI Server

5. Click on the Hyper-V VM Information test to configure it.



Hyper-V VM Information parameters to be configured for Mihype1 (Microsoft Hyper-V)

TEST PERIOD	5 mins
HOST	192.168.10.1
PORT	NULL
IGNORE VMS INSIDE VIEW	none
IGNORE WINNT	● Yes ○ No
EXCLUDE VMS	none
INSIDE VIEW USING	Remote connection to VM (Windows)
* DOMAIN	Chn
* ADMIN USER	sam
* ADMIN PASSWORD	••••••
* CONFIRM PASSWORD	••••••
REPORT BY USER	○ Yes ● No
DD FREQUENCY	1:1
DETAILED DIAGNOSIS	● On ○ Off

Figure 3.9: Configuring the Hyper-V VM Information test

6. To know how to configure the test, refer to the topic on **Hyper-V VM Information** test.

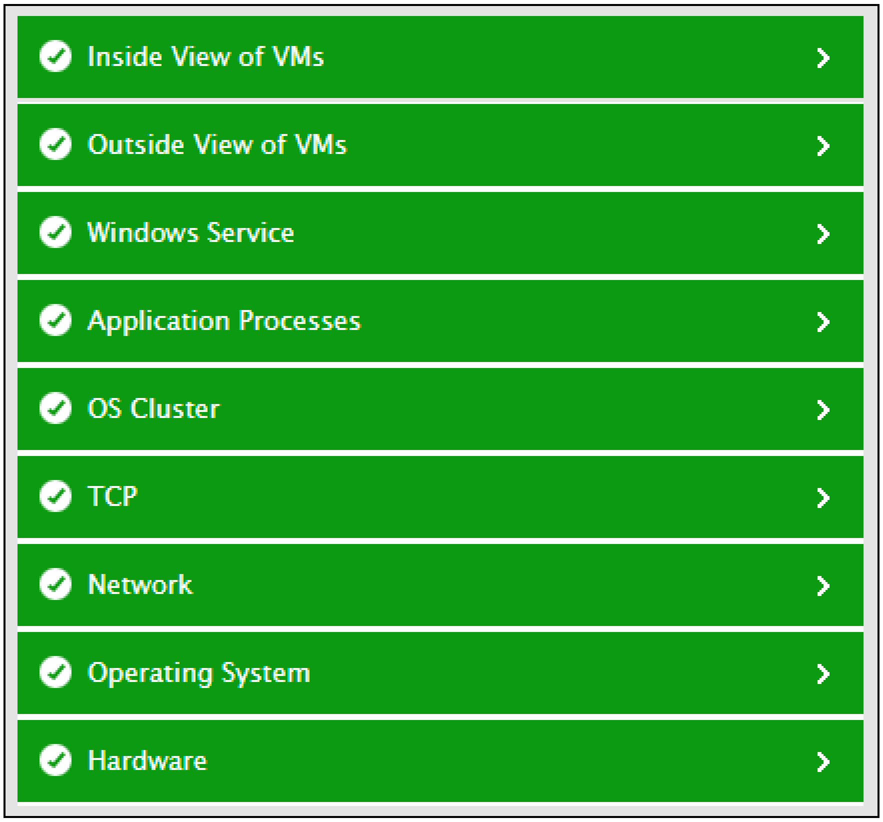
7. After configuring the test, click on the **Update** button in Figure 3.9.

8. Next, signout of the eG administrative interface.

9. Finally, login to the eG monitoring console to view the current status of the Hyper-V / Hyper-V VDI server. You can zoom into the managed Hyper-V / Hyper-V VDI server and view its layer model, tests, and measurements in the console. For more details on the layer model, refer to the topics on Hyper-V and Hyper-V VDI monitoring models.

Chapter 4: The Hyper-V Monitoring Model

eG Enterprise prescribes a specialized *Hyper-V* model (see Figure 4.1) for monitoring Microsoft Hyper-V servers with VMs that host server applications.



✓ Inside View of VMs ›

✓ Outside View of VMs ›

✓ Windows Service ›

✓ Application Processes ›

✓ OS Cluster ›

✓ TCP ›

✓ Network ›

✓ Operating System ›

✓ Hardware ›

Figure 4.1: The layer model of the Hyper-V server

Each layer of Figure 4.1 execute tests that report on key performance parameters pertaining to the *Hyper-V* server. Using these metrics, administrators can find quick and accurate answers for the following questions:

- Are adequate memory and disk resources available on the Hyper-V host operating system?
- Which logical processor is being used excessively? Who is making more use of the CPU resources – the VMs or the hypervisor?
- How many virtual processors does the hypervisor support? How many of these are available to the root partition? Are all processors used optimally, or have abnormal usage trends been

detected with any processor? Which one is it? What is eroding the CPU resources - the guest code or hypervisor code?

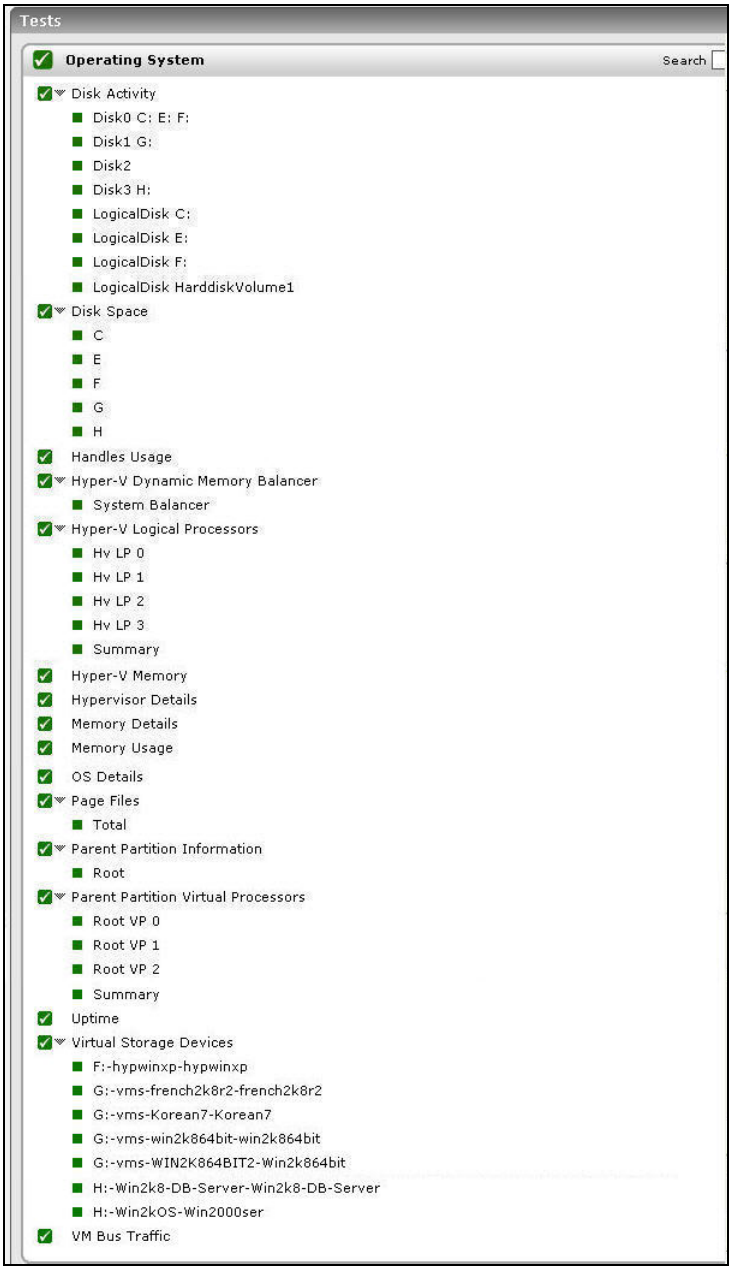
- How many memory pages have been deposited with the root partition?
- Do too many TLB flushes occur on the root partition?
- Are too many TLB large pages been used by the root partition?
- How frequently has the root partition attempted to access a page that is not in the CPU TLB?
- How busy is the root partition? Are hypercalls issued by or instructions completed on the root partition very frequently?
- Is the hypervisor managing memory resources efficiently?
- Is the VMBus able to process interrupts smoothly?
- Which is the busiest network adapter/switch/switch port on the Hyper- V server, in terms of amount of network traffic handled?
- Are all critical Hyper-V processes/services available?
- How are the VMs using each virtual processor assigned them? Is any VM over-utilizing the virtual processors?
- Is any VM currently powered off?
- How many VMs are registered with the Hyper-V servers?
- How much physical memory and disk resources have been allocated to every VM? Which VM has been allocated the maximum memory, CPU, and disk resources?
- Are the network adapters supported by the VM healthy or is too much data being lost on the network adapters?
- Which is the busiest VM on the Hyper- V server, in terms of hypercalls and instructions issued/completed?
- Were any VMs migrated to/from the server? If so, which ones are they, and why were they migrated?
- Are any VMs being deleted on the server?
- Are all VMs currently running, or has any VM been paused for a short while?
- Is any VM inaccessible to users?
- How are the VMs using the allocated resources?

- Is any VM currently experiencing a resource crunch? Are any resource- intensive applications/processes executing on that VM?

The layers depicted by Figure 4.1 and the tests mapped to each are discussed elaborately in the sections to come.

4.1 The Operating System Layer

The tests pertaining to the **Operating System** layer (see Figure 4.2) report on the physical resource usage by the Hyper-V host – i.e., the root partition in particular. The physical disk drive that is experiencing excessive activity, the disk drive that is low on space, logical/virtual processors that are over-utilized can be identified accurately using the tests mapped to this layer.



Tests

☑ Operating System

Search

- ☑ Disk Activity
  - Disk0 C: E: F:
  - Disk1 G:
  - Disk2
  - Disk3 H:
  - LogicalDisk C:
  - LogicalDisk E:
  - LogicalDisk F:
  - LogicalDisk HarddiskVolume1
- ☑ Disk Space
  - C
  - E
  - F
  - G
  - H
- ☑ Handles Usage
- ☑ Hyper-V Dynamic Memory Balancer
  - System Balancer
- ☑ Hyper-V Logical Processors
  - Hv LP 0
  - Hv LP 1
  - Hv LP 2
  - Hv LP 3
  - Summary
- ☑ Hyper-V Memory
- ☑ Hypervisor Details
- ☑ Memory Details
- ☑ Memory Usage
- ☑ OS Details
- ☑ Page Files
  - Total
- ☑ Parent Partition Information
  - Root
- ☑ Parent Partition Virtual Processors
  - Root VP 0
  - Root VP 1
  - Root VP 2
  - Summary
- ☑ Uptime
- ☑ Virtual Storage Devices
  - F:-hypwinxp-hypwinxp
  - G:-vms-french2k8r2-french2k8r2
  - G:-vms-Korean7-Korean7
  - G:-vms-win2k864bit-win2k864bit
  - G:-vms-WIN2K864BIT2-Win2k864bit
  - H:-Win2k8-DB-Server-Win2k8-DB-Server
  - H:-Win2kOS-Win2000ser
- ☑ VM Bus Traffic

Figure 4.2: The tests mapped to the Operating System layer

We will be discussing the Hyper-V-specific tests only.

4.1.1 Hyper-V Memory Test

This test reports how the Hyper-V host uses the physical memory resources available to it, and reveals whether adequate free memory is available on the host or not.

Target of the test : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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. **ENABLE MEMORY DIAGNOSIS -** By default, this flag is set to **No**, indicating that detailed diagnosis will not be available for the Free memory measure reported by this test by default. If you want to view the detailed diagnosis of the Free memory measure - i.e., to view the top 10 processes on the Hyper-V host that are utilizing memory excessively - you can change this flag to **Yes**.
4. **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.

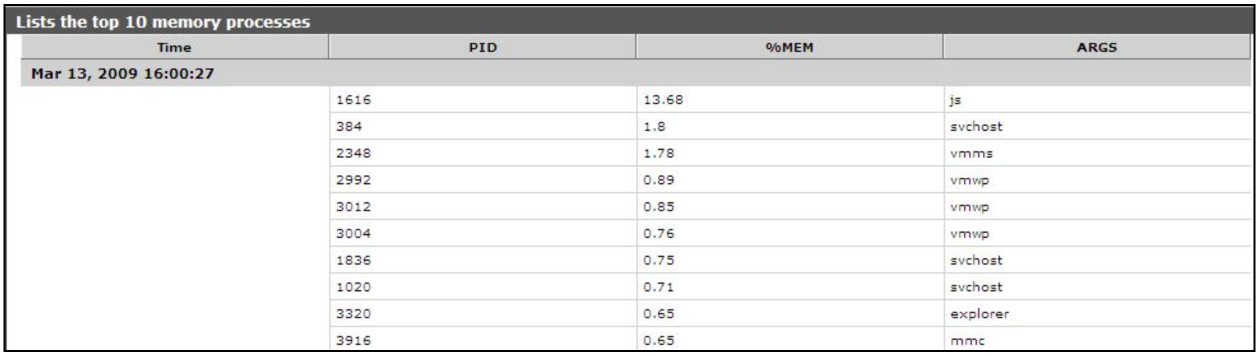
   The ability of this test to provide detailed diagnostic measures is governed by the enable memory diagnosis parameter only. This test therefore, disregards the status of the detailed diagnosis flag when determining whether/not to collect detailed metrics.

Measurements made by the test

Measurements	Description	Measurement Unit	Interpretation
Run queue length	Indicates the instantaneous length of the queue in which threads are waiting for the processor cycle. This	Number	A value consistently greater than 2 indicates that many processes could be simultaneously contending for the processor.

Measurements	Description	Measurement Unit	Interpretation
	length does not include the threads that are currently being executed.		
Number of blocked processes	Indicates the number of processes currently blocked for I/O, paging, etc.	Number	A high value could indicate an I/O problem on the host (e.g., a slow disk).
Swap memory	This measurement 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	Indicates the free memory available currently.	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 host.
Scan rate	Indicates the memory scan rate.	Pages/Sec	A high value is indicative of memory thrashing. Excessive thrashing can be detrimental to application performance.

The detailed diagnosis of the *Free memory* measure lists the top-10 memory-consuming processes on the Hyper-V host. Using this information, you can accurately identify the process that is causing the memory drain on the host.



Lists the top 10 memory processes

Time	PID	%MEM	ARGS
Mar 13, 2009 16:00:27			
	1616	13.68	js
	384	1.8	svchost
	2348	1.78	vmms
	2992	0.89	vmwp
	3012	0.85	vmwp
	3004	0.76	vmwp
	1836	0.75	svchost
	1020	0.71	svchost
	3320	0.65	explorer
	3916	0.65	mmc

Figure 4.3: The top 10 memory consumers on the Hyper-V host

4.1.2 Hyper-V Memory Usage Test

If one/more VMs on a Hyper-V host are over-sized with physical memory resources, it can result in a serious memory contention that may not only affect the host, but also other VMs on the host. By tracking the physical memory allocation to the VMs on a Hyper-V host, administrators can proactively detect over-allocations and can initiate remedial actions before the problem impacts performance. For this, administrators can use the **Hyper-V Memory Usage** test. This test monitors the physical memory allocated to the VMs, and points to those VMs that are allocated more resources than required.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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. **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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Total memory available	Indicates the total physical memory capacity of the host.	MB	
Memory allocated to VMs	Indicates amount of physical memory allocated to VMs.	MB	
Memory used by VMs	Indicates the percentage of physical memory that is allocated to the VMs on the host.	Percent	Ideally, this value should be low. A value close to 100% is indicative of over-allocation of physical memory to the VMs. You can use the detailed diagnosis of this measure to identify which VM has been allocated maximum resources.
Available memory for VMs	Indicates the amount of physical memory unused on the host.	MB	Ideally, the value of this measure should be high. A low value or a consistent decrease in this value is a cause for concern, as it indicates excessive memory usage by the VMs on the host.

4.1.3 Hyper-V Logical Processors Test

A logical processor is a hardware entity, either a processor core or a hyperthread, on which the Hyper-V operating system can schedule a software thread for execution.

The metrics reported by this test enables administrators to figure out the following:

- Which logical processor is being used excessively? Who is making more use of the CPU resources – the VMs or the hypervisor?

- Are the logical processors able to process interrupts well? Is any logical processor experiencing a bottleneck during interrupt processing?
- Has any processor been idle for too long a time? Does that processor receive scheduler interrupts frequently?

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for every logical processor on the Hyper-V 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.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Guest runtime	Indicates the percentage of time guest code is running on this logical processor (LP). For the Summary descriptor, it indicates the average percentage across all logical processors.	Percent	For example, if you have 2 logical processors and one VM running CPU tests you might see the value be 95% for LP(0), 0% for LP(1) and 47.5% for the Summary. From this, you can conclude which processor is being heavily used by the VMs.
Hypervisor runtime	Indicates percentage of time the Hypervisor is running on an LP. For the Summary descriptor, this measure indicates the average percentage across all LPs.	Percent	Ideally, this value should be low. Comparing the value of this measure across LPs will enable you to accurately identify the LP that is being excessively used by the hypervisor.
Idle time	Indicates the	Percent	Ideally, this value should be low.

Measurement	Description	Measurement Unit	Interpretation
	percentage of time the LP is waiting for work. For the Summary descriptor, this measure indicates the average percentage across all LPs.		Comparing the value of this measure across LPs will enable you to accurately identify the LP that is the most idle.
CPU utilization	Indicates the percentage of time this LP was in use.	Percent	This is typically the sum of the Guest runtime and Hypervisor runtime measures. Comparing the value of this measure across LPs will reveal the LP that is being utilized excessively. If the value of this measure is less than 60% consumed, then the LP usage is considered Healthy. A usage level between 60% and 89% consumed, can be considered as a warning. A value between 90% and 100% is indicative of a serious resource contention.
Context switches	Indicates the number of times per second a new Virtual Processor (VP) had been scheduled to a particular Logical Processor (LP). For the Summary descriptor, the value of this measure indicates the total number of VP to LP switches per second.	Switches/Sec	Ideal time context switches of around 1000 for a single guest running are not uncommon. This is due to the fact the VP will “Halt” and allow something else to run if it has no work to do.
Hardware interrupts	Indicates the number of hardware interrupts this	Interrupts/Sec	Hardware interrupts are delivered to the root VP’s corresponding the

Measurement	Description	Measurement Unit	Interpretation
	LP is processing per second. For the Summary descriptor, the value of this measure is the total number of hardware interrupts per second across all LPs.		LP on which it was received. For example a network card will create and interrupt when a packet is received.
Inter processor interrupts received	Indicates the total number of Inter-processor interrupts (IPI) received per second of a given LP. For the Summary descriptor, this is the total number of IPIs received by all LPs.	Interrupts/Sec	IPIs are sent from one processor to another to get the processor to do memory coherency (like TLB, cache, etc.).
Inter processor interrupts sent	Indicates the number of Ips sent per second of a given LP. For the Summary descriptor, this is the total number of IPIs sent by all LPs.	Interrupts/Sec	
Monitor transition cost	This is a current measure of the cost to enter the Hypervisor via an Intercept on a Logical Processor (LP). For the Summary descriptor, it is the total cost across all processors.	Number	Intercepts are like User mode to Kernel Mode context switches except that here it is from the User/Kernel Mode to the Virtual Machine Monitor (VMM) a.k.a the Hypervisor mode. The smaller this value the better. The only real use it has is to figure out the relative performance of processors.
Scheduler interrupts	Indicates the number of scheduler interrupts that	Interrupts/Sec	Scheduler interrupts are sent by the Hypervisor scheduler from one

Measurement	Description	Measurement Unit	Interpretation
	occurred on this LP per second. For the Summary descriptor, the number of scheduler interrupts that occurred across all LPs will be reported as the value of this measure.		Logical Processor (LP) to another to re- evaluate their runlist. The runlist is the list of Virtual Processors (VP) waiting to run on a given LP. This is also a “wake-up” mechanism for an LP that might be sitting idle in a lower power state.

4.1.4 Parent Partition Information Test

The hypervisor creates partitions that are used to isolate guests and host operating systems. A partition is comprised of a physical address space and one or more virtual processors. A parent partition creates and manages child partitions. It contains a virtualization stack, which controls these child partitions. The parent partition is in most occasions also the root partition. It is the first partition that is created and owns all resources not owned by the hypervisor. As the root partition it will handle the loading of and the booting of the hypervisor. It is also required to deal with power management, plug and play and hardware failure events.

The **Parent Partition Information** test monitors the root partition and reports how well the root manages the physical memory resources.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the root partition on the Hyper-V 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.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Address space in the virtual TLB	Indicates the current number of address spaces in the virtual TLB of the partition.	Number	The root partition hosts a data structure called a Translation LookAside Buffer (TLB), which is used to perform Virtual to Physical Address Translation. A TLB has a fixed number of slots containing page table entries, which map virtual addresses onto physical addresses. It is typically a content- addressable memory (CAM), in which the search key is the virtual address and the search result is a physical address. This measure is a good indicator of the size of the TLB.
Virtual processors	Indicates the number of virtual processors present in the root partition currently.	Number	All execution in the root and child partitions happens on Virtual Processors (VPs). At a minimum you will see one VP for each Logical Processor (LP). These account for the root VPs.
Deposited pages	Indicates the number of pages currently deposited into this partition.	Number	For each partition, the hypervisor maintains a memory pool of RAM SPA pages. This pool acts just like a checking account. The amount of pages in the pool is called the balance. Pages are deposited or withdrawn from the pool. When a hypercall that requires memory is made by a partition, the hypervisor withdraws the required memory

Measurement	Description	Measurement Unit	Interpretation
			from the total pool balance of that partition.
Pages used by the virtual TLB	Indicates the number of page tables that are currently present in the virtual TLB of the partition.	Number	
Pages present in the GPA space	Indicates the number of pages currently present in the GPA space of the root partition.	Number	The physical memory that is seen by the hypervisor is called System Physical Address (SPA) space. The pages allocated for the operating system in a child partition are not necessarily contiguous so a remapping takes place to allow the guest to see a contiguous Guest Physical Address (GPA) space. System Physical Address space refers to the physical memory's physical addresses. Guest Physical Address space is the set of pages that are accessed when a guest references a physical address (i.e. when the CR3 register is loaded with the physical address of the page directory). There is one SPA space per machine and one GPA space per child partition. When the operating system is running within a child partition using Hyper-V, the guest page tables reference GPA, although, as far as the child partition operating

Measurement	Description	Measurement Unit	Interpretation
			system knows, this is physical memory. Even though the guest references a GPA page when referring to physical memory, these references have to be converted so the actual memory access is performed on an SPA page.
GPA space modification	Indicates the rate of modifications to the GPA space.	Modifications/Sec	
Virtual TLB flush entries	Indicates the rate of flushes of the entire virtual TLB.	Entries/Sec	When the memory map is changed the entries in the TLB many need to be removed (flushed). TLB flushes can be expensive operations because it may trigger interprocessor interrupts to clear out similar entries on other processors and additional accesses to memory to recompute mappings that were previously in the TLB. Therefore, the value of this measure, should ideally be low.

4.1.5 Parent Partition Virtual Processors Test

A virtual processor is a single logical processor that is exposed to a partition by the hypervisor. Virtual processors can be mapped to any of the available logical processors in the physical computer and are scheduled by the hypervisor to allow you to have more virtual processors than you have logical processors.

This test monitors how well the parent partition uses the virtual processors assigned to it.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the every virtual processor assigned to the root partition of the Hyper-V 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.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Parent partition run time	Indicates the percentage of time spent by this virtual processor (VP) in guest code. For the Summary descriptor, the value of this measure is the total percentage across all VPs.	Percent	Comparing the value of this measure across VPs will accurately indicate which VP is being actively used by the guests.
Hypervisor runtime	Indicates the percentage of time spent by the virtual processor in hypervisor code. For the Summary descriptor, the value of this measure is the total percentage across all VPs.	Percent	Comparing the value of this measure across VPs will accurately indicate which VP is being actively used by the hypervisor.
Parent partition CPU utilization	Indicates the total percentage of time this VP was in use. For the Summary descriptor, this is the average percentage of time for which all VPs were in use.	Percent	This is typically the sum of the Parent partition runtime and Hypervisor runtime measures. Comparing the value of this measure across VPs will reveal the VP that is being utilized excessively.

Measurement	Description	Measurement Unit	Interpretation
Control register accesses	Indicates the number of CPU Control Register accesses per second. For the Summary descriptor, this is rate of CPU control register accesses across all VPs.	Accesses/Sec	Control registers are used to set up address mapping, privilege mode, etc.
CPUID instructions	Indicates the number of CPUID instructions calls per second. For the Summary descriptor, this is rate of CPUID instructions across all VPs.	Instructions/Sec	The CPUID instruction is used to retrieve information on the local CPU's capabilities. Typically, CPUID is only called when the OS / Application first start. Therefore, this value is likely to be 0 most of the time.
Emulated instructions	Indicates the number of emulated instructions completed per second. For the Summary descriptor, this is rate of emulated instructions completed across all VPs.	Instructions/Sec	Some instructions require emulation to complete in the Hypervisor. One such example is APIC access.
HLT instructions	Indicates the number of CPU halts per second on the VP. For the Summary descriptor, this is the total number of CPU halts (per second) across all VPs.	Instructions/Sec	A HLT will cause the hypervisor scheduler to de- schedule the current VP and move to the next VP in the runlist.
Hypercalls	Indicates the number of hypercalls made by guest code on the VP per second. For the	Hypercalls/Sec	Hypercalls are one form of enlightenment. Guest OS's use the enlightenments to more efficiently use the system via the

Measurement	Description	Measurement Unit	Interpretation
	Summary descriptor, this is the total number of hypercalls made on all VPs per second.		hypervisor. TLB flush is an example hypercall. If this value is zero, it is an indication that Integration Components are not installed. New OS's like WS08 can use hypercalls without enlightened drivers. So, hypercalls are only a prerequisite and not a guarantee for not having Integration Components installed.
IO instructions	Indicates the number of CPU in / out instructions executed per second. For the Summary descriptor, this is total number of IO instructions executed on all VPs per second.	Instructions/Sec	Many older or low bandwidth devices use "programmed I/O" via in / out instructions.
Large page TLB fills	Indicates the number of Large Page TLB fills / second. For the Summary descriptor, this is rate of large page TLB fills across all VPs per second.	Fills/Sec	There are two types of TLB entries (and some three). Small TLB which generally means a 4K page and large Page which generally means 2MB. There are fewer Large TLB entries on the order of 8 – 32. A non- zero value for this measures indicates that the root partition is using large pages.
MSR accesses	Indicates the number of Machine Specific Register (MSR) instruction calls per second.For the	Accesses/Sec	There are many types of MSRs such as C-state config, Synthetic Interrupt (Synic) Timers, and control functions such as shutdown.

Measurement	Description	Measurement Unit	Interpretation
	Summary descriptor, this is total number of MSR instruction calls made on all VPs per second.		
MWAIT instructions	Indicates the number of MWAIT instructions per second. For the Summary descriptor, this is the total number of MWAIT instructions executed on all VPs per second.	Instructions/Sec	The mwait (monitored wait) instruction instructs the processor to enter a wait state in which the processor is instructed to monitor the address range between a and b and wait for an event or a store to that address range.
Page fault intercepts	Indicates the number of page faults per second. For the Summary descriptor, this is the total number of page faults on all VPs per second.	Intercepts/Sec	Whenever guest code accesses a page not in the CPU TLB a page fault will occur. This counter is closely correlated with the Large Page TLB Fills measure.
Small page TLB fills	Indicates the number of Small Page TLB fills / second. For the Summary descriptor, this is rate of small page TLB fills across all VPs.	Fills/Sec	There are two types of TLB entries (and some three). Small TLB which generally means a 4K page and large Page which generally means 2MB. There are fewer Large TLB entries on the order of 64 – 1024+.
Total intercepts	Indicates the rate of hypervisor intercept messages. For the Summary descriptor, this is rate at which intercepts occurred across all VPs per	Intercepts/Sec	Whenever a guest VP needs to exit its current mode of running for servicing in the hypervisor, this is called an intercept. Some common causes of intercepts are resolving Guest Physical Address (GPA) to Server Physical Address

Measurement	Description	Measurement Unit	Interpretation
	second.		(SPA) translations, privileged instructions like hlt / cupid / in / out, and the end of the VP's scheduled time slice.
CPU ready time	Indicates the time duration during which this VP was ready to execute the requests to the logical processor but was not able to because of processor contention.	Milliseconds	The value of this measure should typically be low. The more time a VP spends waiting to run, the more lag time there is in responsiveness within the VP.

4.1.6 VM Bus Traffic Test

Child partitions do not have direct access to hardware resources, but instead have a virtual view of the resources, in terms of virtual devices. Any request to the virtual devices is redirected via the VMBus to the devices in the parent partition, which will manage the requests. The VMBus is a logical channel which enables inter- partition communication. The response is also redirected via the VMBus.

Using this test, you can measure the level of activity on the VMBus.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Interrupts received	Indicates the number of interrupts received since the last measurement period.	Number	
Interrupts sent	Indicates the number of interrupts sent since the last measurement period.	Number	
Throttle events	Indicates the total number of times since the last measurement period that any partition has been throttled, which is to say that its interrupts were disabled.	Number	

4.1.7 Hypervisor Status Test

A hypervisor, also called virtual machine monitor (VMM), is a computer hardware platform virtualization software that allows multiple operating systems to run on a host computer concurrently. It runs directly on the host's hardware as a hardware control and guest operating system monitor. A guest operating system thus runs on another level above the hypervisor.

The Hypervisor Status test reports useful statistics revealing the health of the Hyper-V hypervisor.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Logical processors	Indicates the number of cores / HT that the hypervisor is managing currently.	Number	If you have a dual proc quad core without HT you will see this number set to 8. If you also had HT it would be set to 16.
Virtual machines running	Indicates the number of partitions managed by the hypervisor currently.	Number	Each virtual machine on the system is run in a container called a partition. If you have no VMs running this value will be set to 1, because the “host OS” called the “root” in Hyper-V is also running in a partition. So, if you have 2 guest VMs running, this value will be 3 - 2 for each guest VM and 1 for the root.
Virtual processors	Indicates the number of virtual processors on the system currently.	Number	All execution in the root and child partitions (where guest VMs run) happens on Virtual Processors (VPs). At a minimum, you will see one VP for each Logical Processor (LP). These account for the root VPs. You will then see one for each VP you have configured to a guest. Therefore, if you have an 8LP system with 1 guest running with 2 VPs, the count here will be 10.
Monitored notification	Indicates the number of monitored notifications currently registered with the hypervisor.	Number	Monitored notifications are part of an interrupt coalescing technique Hyper- V uses to reduce virtualization overhead. For example, when a guest has data to transmit over the network it could send an interrupt for each packet to

Measurement	Description	Measurement Unit	Interpretation
			the root VP that will actually do the I/O, or it can send one interrupt to let the root know data is starting to flow. This counter is an indication of the number of "flows" of interrupts being set to the root and guests.
Total pages	Indicates the current number of bootstrap and deposited pages in the hypervisor.	Number	The Hypervisor needs memory in order to keep track of Virtual Processors, Guest Virtual address to System Physical Address translation entries in the virtual TLB, etc. Therefore, the total pages keep track of the total amount of memory the Hypervisor is using for management or partitions. A page is 4KBytes. This is not the total amount used to support a guest. You would also need to get this by looking at the size of the worker process (vmwp.exe) and account for memory in vid. Total Pages can change based on what guests VMs are running. Here is an example of how the Hypervisor gets memory - A user want to start a VM. To achieve this, the vid makes a hypercall to the Hypervisor via winhv.sys to create a partition. In order to create VPs, vTLBs, etc., the Hypervisor needs memory. Hence, it makes a call to the root via winhv.sys. Winhv.sys

Measurement	Description	Measurement Unit	Interpretation
			then allocates memory from the root and makes a hypercall to deposit memory and then the whole process unwinds and the partition create completes.

4.1.8 Hyper-V Dynamic Memory Balancer Test

Dynamic Memory is a new feature of Hyper-V™ that enables Hyper-V hosts to dynamically adjust the amount of memory available to virtual machines in response to changing workloads. Instead of assigning a specific amount of memory to a virtual machine, the administrator instead configures a range of memory, memory priority and other settings that Hyper-V then uses to determine how much memory to allocate to the virtual machine in real time. The benefits of Dynamic Memory include higher virtual machine consolidation ratios and increased flexibility for managing virtualized workloads.

By closely monitoring the amount of memory the Hyper-V host dynamically allocates and releases from VMs, you can understand the memory needs of virtual machines and the memory pressure on the host. The **Hyper-V Dynamic Memory Balancer** test enables this monitoring and the consequent analysis.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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.

Measurements reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Added memory	Indicates the amount of	MB	Hyper-V host and the enlightened

Measurement	Description	Measurement Unit	Interpretation
	physical memory added to the VMs during the last measurement period.		VM communicate through the VMBus (the server use Virtual Service Provider and the client use Virtual Service Consumer) to determine the current memory needs of the VM. If the workload of the VM increases and need more memory – then memory is dynamically added to the VM. If the workload decreases (or other VMs have higher memory priority)– the memory is dynamically removed from the VM.
Removed memory	Indicates the amount of physical memory removed from the VMs during the last measurement period.	MB	
Available memory	Indicates the amount of physical memory remaining unused on the host.	MB	A very low value of this measure is a cause of concern. This is because, it indicates that the physical memory of the host has been overcommitted; if too much paging occurs at this juncture, performance will plummet.
Average pressure	Indicates the average memory pressure on the host.	Percent	Dynamic Memory determines the amount of memory needed by a virtual machine by calculating something called memory pressure. To perform this calculation, Hyper- V looks at the total committed memory of the guest operating system running in the virtual machine and then calculates pressure as the ratio of how much memory the virtual machine wants to how much it has. The amount of memory that Hyper-V then assigns to the virtual

Measurement	Description	Measurement Unit	Interpretation
			machine equals total committed memory plus some additional memory to be used as a buffer. However, Dynamic Memory does not guarantee that the total committed memory amount is always assigned to the virtual machine. Neither does Dynamic Memory guarantee that the additional memory amount configured as a buffer value is always assigned to the virtual memory. This is because the actual amount of memory assigned to a virtual machine depends upon the memory pressure being exerted upon the host by the memory needs of other virtual machines running on the host. A very high value of this measure therefore indicates that the VMs are exerting too much memory pressure on the host, probably owing to a severe memory contention on the VMs. As long as this number is under 100, you can conclude that there is enough memory on the Hyper-V host to service your virtual machines. Ideally, this value should be at 80 or lower. The closer this gets to 100, the closer you are to running out of memory. Once this number goes over 100 then you can pretty much guarantee that you have virtual

Measurement	Description	Measurement Unit	Interpretation
			machines that are paging in the guest operating system.
Memory add operations	Indicates the number of memory addition operations performed during the last measurement period.	Number	
Memory remove operations	Indicates the number of memory removal operations performed during the last measurement period.	Number	

4.1.9 Virtual Storage Devices Test

Virtual hard disk (VHD) is a disk image file format for storing the complete contents of a hard drive. It replicates an existing hard drive and includes all data and structural elements.

On the Hyper-V management operating system, virtual hard disks can have a capacity of up to 2040 gigabytes and can be of any of the following types:

- Fixed: A fixed virtual hard disk is a disk that occupies physical disk space on the management operating system equal to the maximum size of the disk, regardless of whether a virtual machine requires the disk space. A fixed virtual hard disk takes longer to create than other types of disks because the allocated size of the .vhd file is determined when it is created. This type of virtual hard disk provides improved performance compared to other types because fixed virtual hard disks are stored in a contiguous block on the management operating system.
- Dynamically expanding: A dynamically expanding virtual hard disk is a disk in which the size of the .vhd file grows as data is written to the disk. This type provides the most efficient use of disk space. You will need to monitor the available disk space to avoid running out of disk space on the management operating system.
- Differencing: A differencing virtual hard disk stores the differences from the virtual hard disk on the management operating system. This allows you to isolate changes to a virtual machine and keep a virtual hard disk in an unchanged state. The differencing disk on the management operating

system can be shared with virtual machines and, as a best practice, must remain read-only. If it is not read-only, the virtual machine‘s virtual hard disk will be invalidated.

The performance of a virtual hard disk is judged by the speed with which it processes I/O requests. Slowdowns in I/O processing can cause read-write requests to the virtual hard disks to queue up, thereby increasing the I/O load on the virtual hard disks and degrading the overall performance of the VMs using those virtual hard disks. Moreover, since virtual hard disk files (.vhd files) are typically stored in the physical disks of the Hyper-V host, excessive I/O activity on the virtual hard disk will also impact the performance of the corresponding physical disks.

Using the **Virtual Storage Devices** test, administrators can periodically monitor the I/O activity on each virtual hard disk assigned to every VM on Hyper-V. This way, probable delays in the processing of I/O requests can be proactively detected, and the physical disks and VMs that may be impacted by these latencies can be isolated.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each virtual disk hosted by a physical disk and assigned to a 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.

Measurements reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Read throughput	Indicates the rate at which data is read from this virtual hard disk that is stored by this physical disk and used by this VM.	Bytes/Sec	While an abnormal increase in the value of these measures could indicate a high level of read-write activity on a virtual hard disk, a consistent decrease in the value of these measures could be indicative of a processing bottleneck probably caused by slowdowns in a virtual hard disk.

Measurement	Description	Measurement Unit	Interpretation
			When a physical disk experiences slowdowns, you can compare the value of these measures across all virtual hard disks that are located on that physical disk, so that you can instantly identify which virtual hard disk is contributing to the processing delay.
Write throughput	Indicates the rate at which data is written to this virtual hard disk that is stored by this physical disk and used by this VM.	Bytes/Sec	
Total throughput	Indicates the rate at which data is read from/written to this virtual hard disk that is stored by this physical disk and used by this VM.	KBps	
Read IOPS	Indicates the rate at which data reads are performed on this virtual hard disk that is stored by this physical disk and used by this VM.	Reads/sec	
Write IOPS	Indicates the rate at which data writes are performed on this virtual hard disk that is stored by this physical disk and used by this VM.	Writes/sec	
Total IOPS	Indicates the total number of data reads and data writes performed per second on this virtual hard disk that is stored by this physical disk and used	Operations/sec	

Measurement	Description	Measurement Unit	Interpretation
	by this VM.		
Errors count	Indicates the number of error events triggered on this virtual hard disk that is stored by this physical disk and used by this VM.	Number	
Flush count	Represents the total number of flush operations that have occurred on this virtual device.	Number	
Read count	Indicates the number of read operations performed on this virtual hard disk that is stored by this physical disk and used by this VM.	Number	
Write count	Indicates the number of write operations performed on this virtual hard disk that is stored by this physical disk and used by this VM.	Number	

4.1.10 Hyper-V Memory Reserve Test

As Hyper-V can dynamically allocate memory to virtual machines on demand, the host needs to ensure that some memory is kept for itself. Without which, Hyper-V may allocate a lot of memory to the VMs, starving the host of adequate memory resources. As a result, the host may start performing poorly. To avoid this, administrators need to time and again check the amount of memory that the host has set aside for its use. This check can be easily performed using the **Hyper-V Memory Reserve** test. This test periodically reports the amount of memory that the host has reserved for itself, thus enabling administrators to periodically check whether the host has sufficient memory for its own operations.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V host being monitored

Configurable parameters for the test

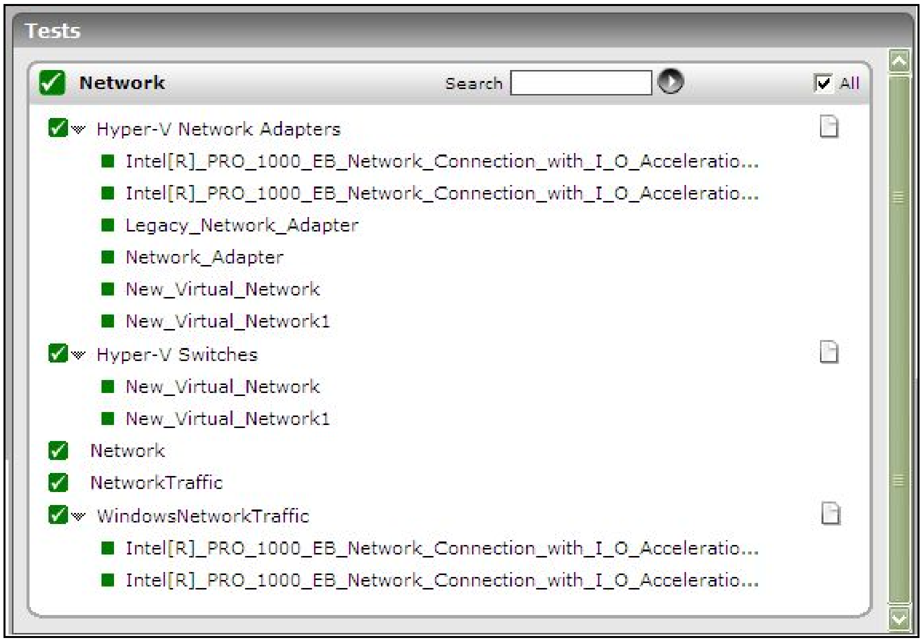
1. **TEST PERIOD -** How often should the test be executed
2. **HOST -** The host for which the test is to be configured.

Measurements reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Host memory reserve	Indicates the amount of memory reserved for the host.	MB	A high value is desired for this measure. If not, the host's performance may suffer, which in time, will affect the VMs' performance as well.

4.2 The Network Layer

The tests mapped to this layer indicate whether the Hyper-V server is available over the network or not, and if so, how quickly it responds to requests. In addition, the tests measure the level of network traffic handled by the adapters, switches, and switch ports supported by the Hyper-V server.



Tests

☑ Network Search ☑ All

☑ Hyper-V Network Adapters
- Intel[R]_PRO_1000_EB_Network_Connection_with_I_O_Acceleratio...
- Intel[R]_PRO_1000_EB_Network_Connection_with_I_O_Acceleratio...
- Legacy_Network_Adapter
- Network_Adapter
- New_Virtual_Network
- New_Virtual_Network1

☑ Hyper-V Switches
- New_Virtual_Network
- New_Virtual_Network1

☑ Network

☑ NetworkTraffic

☑ WindowsNetworkTraffic
- Intel[R]_PRO_1000_EB_Network_Connection_with_I_O_Acceleratio...
- Intel[R]_PRO_1000_EB_Network_Connection_with_I_O_Acceleratio...

Figure 4.4: The tests mapped to the Network layer

4.2.1 Hyper-V Network Adapters Test

There are two types of network adapters available for Hyper-V: a network adapter and a legacy network adapter. For the network adapter to work, integration services must be installed, which is part of the Hyper-V installation. If integration services cannot be installed because of the version of the operating system, the network adapter cannot be used. Instead, you need to add a legacy network adapter that emulates an Intel 21140-based PCI Fast Ethernet Adapter and works without installing a virtual machine driver. A legacy network adapter also supports network- based installations because it includes the ability to boot to the Pre-Execution Environment (PXE boot). The legacy network adapter is also required if a virtual machine needs to boot from a network. You will need to disable the network adapter after the PXE boot.

Hyper- V allows guest computers to share the same physical network adapter. It is therefore necessary for administrators to monitor how each of the physical network adapters are used by both the guest VMs and the host operating system, so that they can accurately determine which adapter is experiencing high usage levels, and also figure out where the network resources are spent more – at the VM-level or at the host operating system-level?

To determine how the Hyper- V operating system utilizes these network adapters, periodically execute the Hyper-V Network Adapters test.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each network adapter available to the Hyper-V 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.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Broadcast packets received	Indicates the total number of packets received per second by this adapter.	Packets/Sec	
Broadcast packets sent	Indicates the total number of packets sent per second by this network adapter.	Packets/Sec	
Data received	Indicates the rate at which bytes were received by this network adapter.	Mbps	
Data transmitted	Indicates the rate at which bytes of data were sent by this network adapter.	Mbps	
Packets received	Indicates the total number of packets received by this network adapter per second.	Packets/Sec	
Packets sent	Indicates the total number of packets sent by this network adapter	Packets/Sec	

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>per second.</td><td></td><td></td></tr>
<tr><td>Is enabled?</td><td>Indicates whether/not this network adapter is enabled.</td><td></td><td>The values that this measure reports and the numeric values that correspond to them have been discussed in the table below:
<table>
<tr><th>State</th><th>Numeric Value</th></tr>
<tr><td>No</td><td>0</td></tr>
<tr><td>Yes</td><td>1</td></tr>
</table>
<b>Note</b>:
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate whether/not this network adapter is enabled. The graph of this measure however, represents the same using the numeric equivalents only.</td></tr>
<tr><td>Uplink status:</td><td>Indicates the current uplink status of this network adapter.</td><td></td><td>The values that this measure reports and the numeric values that correspond to them have been discussed in the table below:
<table>
<tr><th>State</th><th>Numeric Value</th></tr>
<tr><td>Down</td><td>0</td></tr>
<tr><td>Degraded</td><td>1</td></tr>
<tr><td>Up</td><td>2</td></tr>
</table>
<b>Note</b>:
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current uplink status of this network adapter. The</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			graph of this measure however, represents the same using the numeric equivalents only.

4.2.2 Hyper-V Switches Test

A virtual switch can be attached to one and only one physical NIC. Each Virtual / Legacy NIC plugs into a virtual switch. This test gives details on what the switch is doing and the flows of sends / receives it handles.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each virtual switch available to the Hyper-V 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.

Measurements reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Broadcast packets received	Indicates the total number of packets received per second by this switch.	Packets/Sec	
Broadcast packets sent	Indicates the total number of packets sent per second by the switch.	Packets/Sec	

Measurement	Description	Measurement Unit	Interpretation
Data received	Indicates the rate at which bytes were received by the switch.	Mbps	
Data transmitted	Indicates the rate at which bytes of data were sent by the switch.	Mbps	
Packets received	Indicates the total number of packets received by the switch per second.	Packets/Sec	
Packets sent	Indicates the total number of packets sent by the switch per second.	Packets/Sec	

4.2.3 Hyper-V Switch Ports Test

This test reports the network traffic flowing into and out of every virtual switch port (i.e., virtual NIC) on the Hyper- V host. 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 Hyper-V as the **Component type**, Performance as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each virtual switch port available to the Hyper-V 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.

Measurements reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Broadcast packets received	Indicates the total number of packets received per second by this port.	Packets/Sec	
Broadcast packets sent	Indicates the total number of packets sent per second by the switch.	Packets/Sec	
Data received	Indicates the rate at which bytes were received by the port.	Mbps	
Data transmitted	Indicates the rate at which bytes of data were sent by the port.	Mbps	
Packets received	Indicates the total number of packets received by the port per second.	Packets/Sec	
Packets sent	Indicates the total number of packets sent by the port per second.	Packets/Sec	

4.2.4 Windows Team Network Traffic Test

NIC teaming, also known as Load Balancing/Failover (LBFO), allows multiple network adapters to be placed into a team for the purposes of

- bandwidth aggregation, and/or
- traffic failover to maintain connectivity in the event of a network component failure.

The architecture of the Windows NIC teaming solution is as follows:

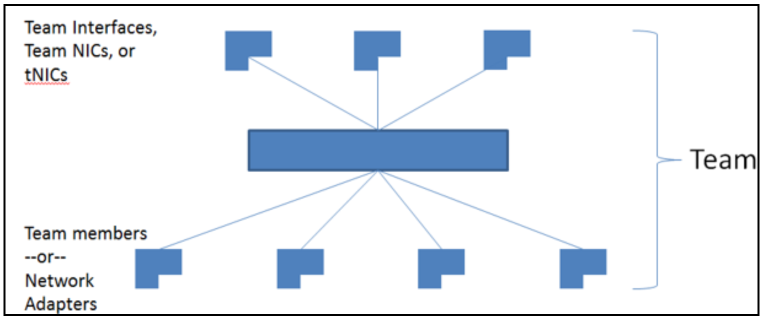


Figure 4.5: NIC teaming architecture

One or more physical NICs are connected into the NIC teaming solution common core, which then presents one or more virtual adapters (team NICs [tNICs] or team interfaces) to the operating system. There are a variety of algorithms that distribute outbound traffic between the NICs.

Regardless of the algorithm used, if a single member of a team goes down, then the other active members of that team may end up handling more traffic than their configuration allows. This can increase packet drops and significantly degrade network performance. Administrators should therefore be promptly alerted if even a single member of a team becomes unavailable.

In addition, administrators should closely monitor the bandwidth usage of a team, so that the adequacy of the collective bandwidth resources of the team members can be evaluated, and team capacity expanded if required.

The Windows Team Network Traffic test helps administrators achieve all of the above. This test monitors the status (up/down) of each NIC team, and alerts administrators if even a single member of a team is unavailable. Additionally, the test reports the bandwidth usage of each team and points to those teams that are experiencing a bandwidth contention. The network throughput of each team and packet discards are reported, so that administrators can quickly identify the team with a poor throughput. In the process, the test throws a spotlight on capacity constraints and performance deficiencies in teams, so that administrators can effectively plan the future capacity of their teams and initiate measures to improve network throughput and performance.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each NIC team

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.

Measurements reported by the test:

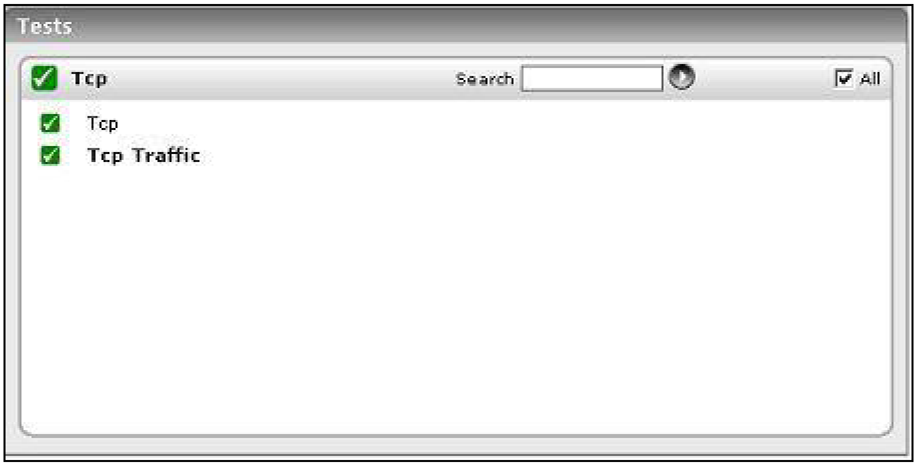
<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Status</td><td>Indicates the current status of this team.</td><td></td><td>This measure reports the value Up, if all members of a team are up and running. The value Down is reported if even one member of a team is down.
The numeric values that correspond to the measure values mentioned above are as follows:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Up</td><td>1</td></tr><tr><td>Down</td><td>0</td></tr></table>
<b>Note:</b>
This test typically reports the Measure Values listed in the table above to indicate the status of a team. In the graph of this measure however, status is represented using the numeric equivalents only.</td></tr>
<tr><td>Incoming traffic</td><td>Indicates the rate at which data (including framing characters) was received by this team.</td><td>Mbps</td><td></td></tr>
<tr><td>Outgoing traffic</td><td>Indicates the rate at which data (including framing characters) was sent by this team.</td><td>Mbps</td><td></td></tr>
<tr><td>Total traffic</td><td>Indicates the rate at which (including framing characters) data was handled by this team.</td><td>Mbps</td><td>This is a good indicator of the throughput of a team.</td></tr>
<tr><td>Max bandwidth</td><td>Represents an estimate of the capacity of the team.</td><td>Mbps</td><td></td></tr>
<tr><td>Bandwidth usage</td><td>Indicates the percentage</td><td>Percent</td><td>A value close to 100% is a cause for</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
	of Max bandwidth used by this team.		concern, as it indicates that the team is about to exhaust the available bandwidth resources. You may want to consider add more NICs to the team to expand its capacity.
Output queue length	Indicates the length of the output packet queue for this team.	Number	A long output queue could indicate that the team members are unable to transmit packets as quickly as they are being sent to it, causing many outgoing packets to queue up for transmission. This in turn could be owing to inadequate processing power / bandwidth with the team. Consider increasing bandwidth by adding more NICs or try fine-tuning the load-balancing algorithm.
Outbound packet discards	Indicates the number of outbound packets that were discarded by this team.	Number	If the value of both or either of these measure is abnormally high for a team, it could imply that one/more NICs in the team are malfunctioning. It could also point to issues with buffer space.
Inbound packet discards	Indicates the number of inbound packets that were discarded by this team.		
Outbound packet errors	Indicates the number of outbound packets that could not be transmitted by this team due to errors.	Number	If packet errors are high, there may be an issue with a malfunctioning NIC. Alternatively, you may want to check your protocol stack and make sure it's working properly and hasn't become corrupted.
Inbound packet errors	Indicates the number of inbound packets that could not be transmitted to this team due to errors.	Number	
Packets received	Indicates the rate at which packets were received by this team.	KB/Sec	An abnormally high value for this measures could indicate excessive traffic on the network. Make sure that your team is configured with adequate bandwidth resources to handle the load. If not, then, you may want to

Measurement	Description	Measurement Unit	Interpretation
			consider expanding the team capacity by adding more NICs to the team.
Packets sent	Indicates the rate at which packets were transmitted by this team.	KB/Sec	

4.3 The TCP Layer

The tests mapped to this layer measure the TCP connectivity of the Hyper-V host.


Tests

☑ Tcp Search ☑ All

☑ Tcp

☑ Tcp Traffic

Figure 4.6: The Tcp layer

Since both the tests mapped to this layer have already been discussed in the *Monitoring Unix and Windows Servers* document, let us proceed to take a look at the next layer.

4.4 The OS Cluster Layer

The tests mapped to this layer measure the health and performance of the **Failover Cluster Service** (if enabled) on the Windows server that is hosting the monitored Hyper- V server. Additionally, the test also monitors I/O traffic to each Cluster Shared Volume (CSV) the Hyper-V cluster supports, and points you to overloaded CSVs. The CSV Cache is also monitored and administrators proactively alerted to ineffective cache usage .

The tests related to the Failover Cluster Service of Windows have already been discussed elaborately in the Monitoring Unix and Windows Servers document. This section will therefore discuss the CSV-related tests alone.

Note:

The tests mapped to the **OS Cluster** layer run only in the agent-based mode. This is why, you need to install an eG agent on at least one node in the cluster to enable these tests to report cluster-level metrics. For best results however, it is recommended that you install an eG agent on each node in the cluster; this way, even if one node goes down due to any reason, cluster health can continue to be monitored using the agents on the other nodes.

4.4.1 Cluster Shared Volumes Test

A CSV is a disk or pool of disks which is accessible by each node in a Hyper-V cluster as if it were a logical disk on the system. Each node in the cluster willl be able to connect to the CSV simultaneously. This allows you to have a common storage location for the VM disk and machine configuration which can be passed to another node in the event of a node failure, without the need for manually mounting a volume or copying files.

To use CSV, a Hyper-V VM is configured and the associated virtual hard disk(s) are created on or copied to a CSV disk. Multiple VHDs can be placed on a CSV that in turn are associated with multiple VMs which can be running on different nodes in the cluster.

Since multiple VMs access a CSV simultaneously, the high availability of the CSV is crucial to the high uptime of the VMs. Administrators should hence be able to promptly detect the unavailability of a CSV, identify the VMs that will be impacted by the same, and initiate measures to bring the CSV back up before it causes any permanent damage to VM operations. Also, the I/O load on the CSV is bound to increase with the count of VMs sharing it! For maximizing CSV and VM performance, administrators should make sure that I/O load is always evenly distributed across the CSVs. To keep an eye on the state of and I/O load on each CSV, and to instantly identify unavailable and/or overloaded CSVs, administrators can use the **Cluster Shared Volumes** test.

This test auto-discovers the CSVs, and for each CSV, reports its current state. In the process, the test promptly alerts you if a CSV goes down! Additionally, the test closely monitors the I/O load on each CSV, measures the rate at which every CSV processes the load, and thus points to those CSVs that are overloaded or are experiencing processing bottlenecks.

Note:

This test is only applicable to Microsoft Hyper-V servers running Windows 2012 (or above).

Target of the test : A Hyper-V / Hyper-V VDI server running Windows 2012

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for every CSV on the server

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.

Measurements reported by the test

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Volume state</td><td>Indicates the current state of this CSV.</td><td></td><td>The values that this measure can report and their corresponding numeric values are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Description</th><th>Numeric Value</th></tr>
<tr><td>Active</td><td>In this state all I/O are proceeding as normal.</td><td>100</td></tr>
<tr><td>Paused</td><td>In this state volume will pause any new I/O and down- level state is cleaned.</td><td>60</td></tr>
<tr><td>Initializing</td><td>In this state all files are invalidated and all IOs except volume IOs are failing</td><td>50</td></tr>
<tr><td>Draining</td><td>In this state volume will pause any new I/O, but down- level files are still opened and some down-</td><td>30</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>Measure Value</th><th>Description</th><th>Numeric Value</th></tr>
<tr><td></td><td>level IOs might be still in process.</td><td></td></tr>
<tr><td>Down</td><td>In this state volume will pause any new I/O. The down- level state is already reapplied.</td><td>10</td></tr>
</table>
<p>Note:</p>
<p>By default, this test reports the Measure Values displayed in the table above to indicate CSV state. In the graph of this measure however, the state is indicated using the numeric equivalents only.</p>
</td></tr>
<tr><td>Direct read throughput</td><td>Indicates the rate at which this CSV reads data from the disk in the Direct I/O Mode or in the Block Level Redirected I/O Mode.</td><td>Kbps</td><td>
<p>These measures include both Direct I/O and Block Level Redirected I/O. In Direct Mode, I/O operations from the application on the cluster node can be sent directly to the storage. It therefore, bypasses the NTFS or ReFS volume stack. In Block level redirected Mode, I/O passes through the local CSVFS proxy file system stack and is written directly to Disk.sys on the coordinator node. As a result it avoids traversing the NTFS/ReFS file system stack twice.</p>
<p>The technologies that let CSV-enabled volumes operate require one cluster node that's responsible for the coordination of file access. This cluster</p>
</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			node is called the coordinator node, with each individual LUN having its own coordinator node. If the node being monitored is a co-ordinator node, then these measures include the following: • the rate at which this CSV reads/writes (as the case may be) data directly to the storage, in the Direct I/O Mode. • the rate at which this CSV reads/writes I/O redirected by all slave nodes in the cluster directly to the storage, in the Block Level Redirected I/O Mode. If the node being monitored is a non-coordinator node, then these measures include the following: • the rate at which this CSV reads/writes (as the case may be) data directly to the storage, in the Direct I/O Mode. • the rate at which this CSV reads/writes (as the case may be) I/O to the disk by redirecting the I/O to the coordinator node, in the Block Level Redirected I/O Mode.
Direct write throughput	Indicates the rate at which this CSV writes data to the disk in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Kbps	
Total direct throughput	Indicates the rate at which this CSV reads data from and writes data to the disk in the Direct I/O Mode or in	Kbps	This is a good indicator of the level of direct I/O activity on a CSV. By comparing the value of this measure across CSVs, you can figure out which CSV is experiencing maximum direct

Measurement	Description	Measurement Unit	Interpretation
	the Block Level Redirected I/O Mode.		traffic. If this max value is abnormally high for that CSV, you may want to investigate the reasons for the same.
Redirected read throughput	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV reads data from the physical disk via NTFS, in the File System Redirected Mode. If the node being monitored is a non-coordinator node, then this measure indicates the rate at which this CSV readsdata from the disk by redirecting the I/O to the co-ordinator node via SMB, in the File System Redirected Mode.	Kbps	The technologies that let CSV-enabled volumes operate require one cluster node that's responsible for the coordination of file access. This cluster node is called the coordinator node, with each individual LUN having its own coordinator node. That node can be any of your cluster hosts, with each host having an equal chance of being given the job. While this responsibility doesn't come into play often—typically, Hyper-V interacts with its disk files directly, not necessarily through a coordinator node— it's important for certain types of actions. One of those actions is copying VHD files to a LUN. Hyper- V transparently redirects the file copy through the coordinator node. I/O redirection can also occur if slave nodes in a cluster are unable to access the disk directly. In this case, the slave nodes will redirect the I/O to the co-ordinator node via the SMB Client protocol. The coordinator node then processes the redirected I/I/O it receives using the SMB Server protocol . This redirection is performed in the File System Redirected Mode only. In File System Redirected Mode, I/O on a

Measurement	Description	Measurement Unit	Interpretation
			cluster node is redirected at the top of the CSV pseudo-file system stack over SMB to the disk. This traffic is written to the disk via the NTFS or ReFS file system stack on the coordinator node. From this, we can conclude that for a CSV attached to a co-ordinator node, the value of the *Redirected read throughput* measure will represent the rate at which the read I/Os redirected by all slave nodes in the cluster are received and processed by this CSV in the File System Redirected Mode. For a CSV on a slave/non-coordinator node, the value of this measure will indicate the rate at which that CSV redirected the read I/Os to the coordinator node and read data from the disk. In case of a slave node, the value of this measure will also include the rate at which VHD files are read from that CSV to be written/copied to a CSV on the coordinator node. The value of the *Redirected write throughput* measure for a CSV attached to a coordinator node will include: • the rate at which the write I/Os redirected by all slave nodes in the cluster are received and processed by this CSV in the File System Redirected Mode. • the rate at which the VHD files are copied to the LUN; For a slave/non-coordinator node on the other hand, the value of the *Redirected write throughput* measure will represent only the rate at which that CSV redirects write I/Os to the coordinator node and writes data to the disk, in the File System Redirected Mode.

Measurement	Description	Measurement Unit	Interpretation
Redirected write throughput	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV writes data to the physical disk via NTFS, in the File System Redirected Mode. If the node being monitored is a non-coordinator node, then this measure indicates the rate at which this CSV writes data to the disk by redirecting the I/O to the co-ordinator node via SMB, in the File System Redirected Mode.	Kbps	
Redirected total throughput	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV writes data to the physical disk via NTFS, in the File System Redirected Mode. If the node being monitored is a non-	Kbps	This is the sum of the values of the *Redirected read throughput* and *Redirected write throughput* measures. This is a good indicator of the level of redireced I/O activity on a CSV. By comparing the value of this measure across CSVs, you can figure out which CSV is experiencing maximum redirected traffic. If this max value is abnormally high for that CSV, you may want to investigate the reasons for the same.

Measurement	Description	Measurement Unit	Interpretation
	coordinator node, then this measure indicates the rate at which this CSV writes data to the disk by redirecting the I/O to the co-ordinator node via SMB, in theFile System Redirected Mode.		
Read throughput	Indicates the rate at which data was read by this CSV, both directly and via redirection - i.e.,in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.	Kbps	This is the sum of the values of the *Direct read throughput* and *Redirected read throughput* measures.
Write throughput	Indicates the rate at which data was written by this CSV, both directly and via redirection - i.e.,in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.	Kbps	This is the sum of the values of the *Direct write throughput* and *Redirected write throughput* measures.
Throughput	Indicates the rate at which data was read and written by this CSV, both directly	Kbps	This is the sum of the values of the *Read throughput* and *Write throughput* measures.

Measurement	Description	Measurement Unit	Interpretation
	and via redirection - i.e.,in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.		This is a good indicator of the level of I/O activity on a CSV. By comparing the value of this measure across CSVs, you can figure out which CSV is experiencing maximum traffic. If this max value is abnormally high for that CSV, you may want to investigate the reasons for the same.
Direct read rate	Indicates the rate at which this CSV performs disk reads in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Reads/Sec	These measures include both Direct I/O and Block Level Redirected I/O. In Direct Mode, I/O operations from the application on the cluster node can be sent directly to the storage. It therefore, bypasses the NTFS or ReFS volume stack. In Block level redirected Mode, I/O passes through the local CSVFS proxy file system stack and is written directly to Disk.sys on the coordinator node. As a result it avoids traversing the NTFS/ReFS file system stack twice. If the node being monitored is a co-ordinator node, then these measures will include the following: • the rate at which this CSV performs reads/writes (as the case may be) directly on the storage, in the Direct I/O Mode. • the rate at which this CSV services read/write (as the case may be) requests redirected to it by all slave nodes in the cluster, in the Block Level Redirected I/O Mode.

Measurement	Description	Measurement Unit	Interpretation
			If the node being monitored is a non-coordinator node, then these measures will include the following: • the rate at which this CSV performs read/write (as the case may be) operations directly on the storage, in the Direct I/O Mode. • the rate at which this CSV performs read/write (as the case may be) operations on the storage by redirecting read/write requests to the coordinator node, in the Block Level Redirected I/O Mode.
Direct write rate	Indicates the rate at which this CSV performs disk writes in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Writes/Sec	
Total direct IOPS	Indicates the rate at which this CSV performs IOPS in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Operations/Sec	This is a good indicator of the level of I/O activity on the CSV in the Direct I/O Mode or in the Block Level Redirected I/O Mode.
Redirected read rate	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV reads from the disk via NTFS, in the File System Redirected Mode. If the node being monitored is a non- coordinator node, then this measure indicates	Reads/Sec	

Measurement	Description	Measurement Unit	Interpretation
	the rate at which this CSV reads from the disk by redirecting the read requests to the co-ordinator node via SMB, in the File System Redirected Mode.		
Redirected write rate	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV writes to the disk via NTFS, in the File System Redirected Mode. If the node being monitored is a non-coordinator node, then this measure indicates the rate at which this CSV writes to the disk by redirecting the write requests to the co-ordinator node via SMB, in the File System Redirected Mode.	Writes/Sec	
Total redirected IOPS	Indicates the rate at which I/O reads and writes were performed by this	Operations/Sec	This is a good indicator of the level of I/O activity in the File System Redirected Mode.

Measurement	Description	Measurement Unit	Interpretation
	CSV on the disk via NTFS, in the File System .Redirected Mode.		
Read IOPS	Indicates the rate at which read I/O operations were performed on this CSV, both directly and via redirection - i.e., in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.	Reads/Sec	The value of this measure is the sum of the values of the *Direct read rate* and *Redirected read rate* measures.
Write IOPS	Indicates the rate at which write I/O operations were performed on this CSV, both directly and via redirection - i.e., in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.	Writes/Sec	The value of this measure is the sum of the values of the *Direct write rate* and *Redirected write rate* measures.
IOPS	Indicates the rate at which read and write I/O operations were performed on this CSV, both directly and via redirection -	Operations/Sec	The value of this measure is the sum of the values of the *Read IOPS* and *Write IOPS* measures. This is a good indicator of the level of I/O activity on the CSV.

Measurement	Description	Measurement Unit	Interpretation
	i.e., in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.		
Direct read latency	Indicates the average latency between the time a read request is sent to this CSV and when its response is received, in the Direct I/O Mode or Block Level Redirected I/O Mode.	Secs	If I/Os are sent using Block Level Redirected I/O alone, then the value of this measure will be close to the value of the *Read latency* measure of the *SMB Client Share* test for that CSV.
Direct write latency	Indicates the average latency between the time a write request is sent to this CSV and when its response is received, , in the Direct I/O Mode or Block Level Redirected I/O Mode.	Secs	If I/Os are sent using Block Level Redirected I/O alone, then the value of this measure will be close to the value of the *Write latency* measure of the *SMB Client Share* test for that CSV.
Total direct latency	Indicates the latency of I/O operations performed in Direct I/O Mode or Block Level Redirected I/O Mode since the last time of data collection.	Secs	A low value is desired for this measure.

Measurement	Description	Measurement Unit	Interpretation
Redirected read latency	If the node being monitored is a coordinator node, this measure indicates the average latency between the time a read request redirected by a slave node is received by this CSV in the File System Redirected Mode, and when its response is sent. If the node being monitored is a non-coordinator node, this measure indicates the average latency between the time a read request is redirected by this CSV to the coordinator node in the File System Redirected Mode, and when its response is received.	Secs	
Redirected write latency	If the node being monitored is a coordinator node, this measure indicates the average latency between the time a read request	Secs	

Measurement	Description	Measurement Unit	Interpretation
	redirected by a slave node is received by this CSV in the File System Redirected Mode, and when its response is sent. If the node being monitored is a non-coordinator node, this measure indicates the average latency between the time a read request is redirected by this CSV to the coordinator node in the File System Redirected Mode, and when its response is received.		
Total redirected latency	Indicates the latency of I/O operations redirected to this CSV in the File System Redirected Mode since the last time of data collection.	Secs	Ideally, the value of this measure should be low. Compare the value of this measure across CSVs to know which CSV is taking the maximum time to process I/O in the File System Redirected Mode.
Read latency	Indicates the average latency of read operations performed by this CSV on the disk, both directly and	Secs	The value of this measure is the sum of the values of the*Direct read latency* and *Redirected read latency* measures.

Measurement	Description	Measurement Unit	Interpretation
	via redirection - i.e., in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.		
Write latency	Indicates the average latency of write operations performed by this CSV on the disk, both directly and via redirection - i.e., in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.	Secs	The value of this measure is the sum of the values of the *Direct write latency* and *Redirected write latency* measures.
Latency	Indicates the average latency of both read and write operations performed by this CSV on the disk, both directly and via redirection - .i.e., in the Direct I/O, Block Level Redirected I/O, and File System Redirected I/O Modes.	Secs	The value of this measure is the sum of the values of the *Read latency* and *Write latency* measures. A low value is desired for this measure. A consistent rise in the value of the measure is an indicator of a processing bottleneck on the CSV.
Direct read queue length	Indicates the number of read I/Os currently outstanding on this CSV, in the Direct I/O or Block Level	Number	

Measurement	Description	Measurement Unit	Interpretation
	Redirected I/O Mode.		
Direct write queue length	Indicates the number of write I/Os currently outstanding on this CSV, in the Direct I/O or Block Level Redirected I/O Mode.	Number	
Total direct queue length	Indicates the number of read and write I/Os that are currently pending processing on this CSV, in the Direct I/O or Block Level Redirected I/O Mode.	Number	A zero value is desired for this measure. A consistent increase in the value of this measure is a cause for concern, as it indicates that the CSV is having difficulty processing /O requests in the Direct I/O or Block Level Redirected I/O Mode.
Redirected read queue length	Indicates the number of reads currently outstanding on this CSV, in the File System Redirected Mode.	Number	
Redirected write queue length	Indicates the number of writes currently outstanding on this CSV, in the File System Redirected Mode.	Number	
Total redirected queue length	Indicates the number of writes currently outstanding on this CSV, in the File System Redirected Mode.	Number	A zero value is desired for this measure. A consistent increase in the value of this measure is a cause for concern, as it indicates that the CSV is having difficulty processing I/O requests in the File System Redirected I/O Mode.

Measurement	Description	Measurement Unit	Interpretation
Read queue length	Indicates the count of read operations that are currently outstanding on this CSV, in the Direct I/O, Block Redirected I/O, and File System Redirected I/O Modes.	Number	
Write queue length	Indicates the count of write operations that are currently outstanding on this CSV, in the Direct I/O, Block Redirected I/O, and File System Redirected I/O Modes.	Number	
Queue length	Indicates the count of read and write operations that are currently outstanding on this CSV, in the Direct I/O, Block Redirected I/O, and File System Redirected I/O Modes.	Number	A zero value is desired for this measure. A consistent increase in the value of this measure is a cause for concern, as it indicates that the CSV is having difficulty processing I/O requests.

4.4.2 Cluster Shared Volume Spaces Test

A CSV is a disk or pool of disks which is accessible by each node in a Hyper-V cluster as if it were a logical disk on the system. Each node in the cluster willl be able to connect to the CSV simultaneously. This allows you to have a common storage location for the VM disk and machine configuration which can be passed to another node in the event of a node failure, without the need for manually mounting a volume or copying files.

To use CSV, a Hyper-V VM is configured and the associated virtual hard disk(s) are created on or copied to a CSV disk. Multiple VHDs can be placed on a CSV that in turn are associated with multiple VMs which can be running on different nodes in the cluster.

Since multiple VMs access a CSV simultaneously, the I/O load on the CSV is bound to increase with the count of VMs sharing it! For maximizing CSV and VM performance, administrators should make sure that I/O load is always evenly distributed across the CSVs. To keep an eye on the I/O load on each CSV and to instantly identify overloaded CSVs, administrators can use the **Cluster Shared Volume Spaces** test.

This test auto-discovers the CSVs and closely monitors the I/O load on each CSV, measures the rate at which every CSV processes the load, and thus points to those CSVs that are overloaded or are experiencing processing bottlenecks.

Note:

This test is only applicable to Microsoft Hyper-V servers running Windows 2008.

Target of the test : A Hyper-V / Hyper-V VDI server running Windows 2008

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for every CSV on the server

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.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Read throughput	Indicates the rate at which this CSV reads data from the disk in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Kbps	These measures include both Direct I/O and Block Level Redirected I/O. In Direct Mode, I/O operations from the application on the cluster node can be sent directly to the storage. It therefore, bypasses the NTFS or

Measurement	Description	Measurement Unit	Interpretation
			ReFS volume stack. In Block level redirected Mode, I/O passes through the local CSVFS proxy file system stack and is written directly to Disk.sys on the coordinator node. As a result it avoids traversing the NTFS/ReFS file system stack twice. The technologies that let CSV-enabled volumes operate require one cluster node that's responsible for the coordination of file access. This cluster node is called the coordinator node, with each individual LUN having its own coordinator node. If the node being monitored is a co-ordinator node, then these measures include the following: • the rate at which this CSV reads/writes (as the case may be) data directly to the storage, in the Direct I/O Mode. • the rate at which this CSV reads/writes I/O redirected by all slave nodes in the cluster directly to the storage, in the Block Level Redirected I/O Mode. If the node being monitored is a non-coordinator node, then these measures include the following: • the rate at which this CSV reads/writes (as the case may be) data directly to the storage, in the Direct I/O Mode. • the rate at which this CSV reads/writes (as the case may

Measurement	Description	Measurement Unit	Interpretation
Write throughput	Indicates the rate at which this CSV writes data to the disk in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Kbps	
Throughput	Indicates the rate at which this CSV reads data from and writes data to the disk in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Kbps	The value of this measure is the sum of the values of the *Read throughput* and *Write throughput* measures. This is a good indicator of the level of direct I/O activity on a CSV. By comparing the value of this measure across CSVs, you can figure out which CSV is experiencing maximum direct traffic. If this max value is abnormally high for that CSV, you may want to investigate the reasons for the same.
Redirected read throughput	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV reads data from the physical disk via NTFS, in the File System Redirected Mode. If the node being monitored is a non-coordinator node, then this measure indicates the rate at which this CSV reads	Kbps	The technologies that let CSV-enabled volumes operate require one cluster node that's responsible for the coordination of file access. This cluster node is called the coordinator node, with each individual LUN having its own coordinator node. That node can be any of your cluster hosts, with each host having an equal chance of being given the job. While this responsibility doesn't come into play often— typically, Hyper-V interacts with its disk files directly, not necessarily through a

Measurement	Description	Measurement Unit	Interpretation
	data from the disk by redirecting the I/O to the co-ordinator node via SMB, in the File System Redirected Mode.		coordinator node—it's important for certain types of actions. One of those actions is copying VHD files to a LUN. Hyper- V transparently redirects the file copy through the coordinator node. I/O redirection can also occur if slave nodes in a cluster are unable to access the disk directly. In this case, the slave nodes will redirect the I/O to the co- ordinator node via the SMB Client protocol. The coordinator node then processes the redirected I/I/O it receives using the SMB Server protocol . This redirection is performed in the File System Redirected Mode only. In File System Redirected Mode, I/O on a cluster node is redirected at the top of the CSV pseudo-file system stack over SMB to the disk. This traffic is written to the disk via the NTFS or ReFS file system stack on the coordinator node. From this, we can conclude that for a CSV attached to a co- ordinator node, the value of the *Redirected read throughput* measure will represent the rate at which the read I/Os redirected by all slave nodes in the cluster are received and processed by this CSV in the File System Redirected Mode. For a

Measurement	Description	Measurement Unit	Interpretation
			CSV on a slave/non- coordinator node, the value of this measure will indicate the rate at which that CSV redirected the read I/Os to the coordinator node and read data from the disk. In case of a slave node, the value of this measure will also include the rate at which VHD files are read from that CSV to be written/copied to a CSV on the coordinator node. The value of the *Redirected write throughput* measure for a CSV attached to a coordinator node will include: • the rate at which the write I/Os redirected by all slave nodes in the cluster are received and processed by this CSV in the File System Redirected Mode. • the rate at which the VHD files are copied to the LUN; For a slave/non-coordinator node on the other hand, the value of the *Redirected write throughput* measure will represent only the rate at which that CSV redirects write I/Os to the coordinator node and writes data to the disk, in the File System Redirected Mode.
Redirected write throughput	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV writes data to the physical disk via NTFS, in the File System Redirected Mode. If the node being monitored is a non-coordinator node, then this measure indicates the rate at which this CSV writes data to the disk by redirecting the I/O to the co-ordinator node via SMB, in the File System Redirected Mode.	Kbps	
Redirected throughput	If the node being monitored is a co-	Kbps	This is the sum of the values of the *Redirected read throughput* and

Measurement	Description	Measurement Unit	Interpretation
	ordinator node, then this measure indicates the rate at which this CSV writes data to the physical disk via NTFS, in the File System Redirected Mode. If the node being monitored is a non-coordinator node, then this measure indicates the rate at which this CSV writes data to the disk by redirecting the I/O to the co-ordinator node via SMB, in theFile System Redirected Mode.		*Redirected write throughput* measures. This is a good indicator of the level of redireced I/O activity on a CSV. By comparing the value of this measure across CSVs, you can figure out which CSV is experiencing maximum redirected traffic. If this max value is abnormally high for that CSV, you may want to investigate the reasons for the same.
CSV Read IOPS	Indicates the number of disk reads performed by this CSV in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Number	These measures include both Direct I/O and Block Level Redirected I/O. In Direct Mode, I/O operations from the application on the cluster node can be sent directly to the storage. It therefore, bypasses the NTFS or ReFS volume stack. In Block level redirected Mode, I/O passes through the local CSVFS proxy file system stack and is written directly to Disk.sys on the coordinator node. As a result it avoids traversing the NTFS/ReFS file system stack twice. If the node being monitored is a co-ordinator node, then these

Measurement	Description	Measurement Unit	Interpretation
			measures will include the following: • the number of read/write (as the case may be)operations performed by this CSV directly on the storage, in the Direct I/O Mode. • the number of read/write (as the case may be) operations performed by this CSV in response to read/write requests redirected to it by all slave nodes in the cluster, in the Block Level Redirected I/O Mode. If the node being monitored is a non-coordinator node, then these measures will include the following: • the number of read/write operations (as the case may be) performed by this CSV directly on the storage, in the Direct I/O Mode. • the number of read/write operations (as the case may be) performed by this CSV by redirecting read/write requests to the coordinator node, in the Block Level Redirected I/O Mode.
CSV Write IOPS	Indicates the number of write operations performed by this CSV in the Direct I/O Mode or in the Block Level Redirected I/O Mode.	Writes/Sec	
CSV IOPS	Indicates the total number of I/O operations performed by this CSV in the	Operations/Sec	The value of this measure is the sum of the values of the *CSV Read IOPS* and *CSV Write IOPS* measures.

Measurement	Description	Measurement Unit	Interpretation
	Direct I/O Mode or in the Block Level Redirected I/O Mode.		This is a good indicator of the level of I/O activity on the CSV in the Direct I/O Mode or in the Block Level Redirected I/O Mode.
Redirected read IOPS	If the node being monitored is a co-ordinator node, then this measure indicates the number of read operations performed by this CSV via NTFS, in the File System Redirected Mode. If the node being monitored is a non-coordinator node, then this measure indicates the number of read operations performed by this CSV by redirecting the read requests to the co-ordinator node via SMB, in the File System Redirected Mode.	Number	
Redirected write IOPS	If the node being monitored is a co-ordinator node, then this measure indicates the number of CSV writes to the disk via NTFS, in the File System Redirected	Number	

Measurement	Description	Measurement Unit	Interpretation
	Mode. If the node being monitored is a non-coordinator node, then this measure indicates the number of CSV writes to the disk by redirecting the write requests to the co-ordinator node via SMB, in the File System Redirected Mode.		
Redirected IOPS	Indicates the number of I/O reads and writes performed by this CSV on the disk via NTFS, in the File System .Redirected Mode.	Number	The value of this measure is the sum of the values of the *Redirected read IOPS* and *Redirected Write IOPS* measures. This is a good indicator of the level of I/O activity in the File System Redirected Mode.

4.4.3 Cluster Shared Volume Manager Test

One of the key modules of the CSV file system stack is the CSV Volume Manager. This is the driver that makes sure that the CSVs are showed as local volumes. This driver sits in the volume layer and processes I/O in the Direct mode and/or in the Block Level Redirected I/O Mode.

In Direct Mode, I/O operations from the application on the cluster node can be sent directly to the storage via the Cluster Volume Manager. It therefore, bypasses the NTFS or ReFS volume stack.

In Block Level Redirected I/O Mode, I/O passes through the CSV Volume Manager in the local CSVFS proxy file system stack and is written directly to Disk.sys on the coordinator node.

By closely monitoring the I/O traffic to a CSV Volume Manager, administrators will be able to gauge the level of I/O activity on the corresponding CSV. In the process, overloaded CSVs can be isolated and the nature of I/O traffic contributing to the load - i.e., whether Direct I/O or Block Level Redirected I/O - can be pinpointed. This is exactly what the **Cluster Shared Volume Manager** test. For each CSV, this test reports the rate of direct and redirected reads and writes to that CSV. This way, the test sheds light on the I/O load on every CSV, pinpoints overloaded CSVs, and thus reveals irregularities in load balancing across CSVs.

Target of the test : A Hyper-V / Hyper-V VDI server running Windows 2012

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for every CSV on the server

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.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Direct read throughput	Indicates the rate at which this CSV reads data from the disk in the Direct I/O Mode.	Kbps	
Direct write throughput	Indicates the rate at which this CSV writes data to the disk in the Direct I/O Mode.	Kbps	
Total direct throughput	Indicates the rate at which this CSV reads data from and writes data to the disk in the Direct I/O Mode.	Kbps	The value of this measure is the sum of the values of the *Direct read throughput* and *Direct write throughput* measure. This is a good indicator of the level of direct I/O activity on a CSV. By comparing the value of this measure across CSVs, you can figure out which CSV is experiencing maximum direct traffic. If this max value is abnormally high for that CSV, you

Measurement	Description	Measurement Unit	Interpretation
			may want to investigate the reasons for the same.
Redirected read throughput	Indicates the rate at which this CSV reads data from the disk by redirecting the I/O to the co- ordinator node via SMB in the Block Level Redirected I/O mode.	Kbps	The technologies that let CSV-enabled volumes operate require one cluster node that's responsible for the coordination of file access. This cluster node is called the coordinator node, with each individual LUN having its own coordinator node. I/O redirection can also occur if slave nodes in a cluster are unable to access the disk directly. In this case, the CSV Volume Manager on the slave nodes will redirect the I/O to the co-ordinator node via the SMB Client protocol. The coordinator node then processes the redirected I/O it receives using the SMB Server protocol. The Cluster Shared Volume Manager performs this redirection in the Block Level Redirected Redirected Mode only. From this, we can conclude that for a CSV attached to a slave/non- coordinator node, the value of the *Redirected read throughput* measure will indicate the rate at which that CSV redirected the read I/Os to the coordinator node and read data

Measurement	Description	Measurement Unit	Interpretation
			from the disk, in the Block Level Redirected I/O Mode. For a slave/non-coordinator node, the value of the *Redirected write throughput* measure will represent only the rate at which that CSV redirects write I/Os to the coordinator node and writes data to the disk, in the Block Level Redirected I/O Mode. For a Coordinator Node on the other hand, these measures denote the rate at which the Coordinator Node services read/write I/Os (as the case may be) redirected to it by all slave nodes, in the Block Level Redirected I/O Mode.
Redirected write throughput	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV writes data to the physical disk, in the Block Level Redirected I/O Mode. If the node being monitored is a non- coordinator node, then this measure indicates the rate at which this CSV writes data to the disk by redirecting the I/O to the co- ordinator node via SMB, in the Block Level Redirected I/O Mode.	Kbps	
Redirected total throughput	Indicates the rate at which this CSV writes data to the disk by redirecting the I/O to the co- ordinator node via SMB, in the Block Level Redirected I/O Mode.	Kbps	This is the sum of the values of the *Redirected read throughput* and *Redirected write throughput* measures. This is a good indicator of the level of redirected I/O activity on a CSV. By comparing the value of this measure across CSVs, you can figure out which CSV is experiencing maximum redirected traffic. If this max value is abnormally high for that CSV, you may want to investigate the reasons for the same.

Measurement	Description	Measurement Unit	Interpretation
Read throughput	Indicates the rate at which data was read by this CSV, both directly and via redirection - i.e.,in the Direct I/O and Block Level Redirected I/O modes.	Kbps	This is the sum of the values of the *Direct read throughput* and *Redirected read throughput* measures.
Write throughput	Indicates the rate at which data was written by this CSV, both directly and via redirection - i.e.,in the Direct I/O and Block Level Redirected I/O modes.	Kbps	This is the sum of the values of the *Direct write throughput* and *Redirected write throughput* measures.
Throughput	Indicates the rate at which data was read and written by this CSV, both directly and via redirection - i.e.,in the Direct I/O and Block Level Redirected I/O modes.	Kbps	This is the sum of the values of the *Read throughput* and *Write throughput* measures. This is a good indicator of the level of I/O activity on a CSV. By comparing the value of this measure across CSVs, you can figure out which CSV is experiencing maximum traffic. If this max value is abnormally high for that CSV, you may want to investigate the reasons for the same.
Direct read rate	Indicates the rate at which this CSV performs disk reads in the Direct I/O Mode.	Reads/Sec	
Direct write rate	Indicates the rate at	Writes/Sec	

Measurement	Description	Measurement Unit	Interpretation
	which this CSV performs disk writes in the Direct I/O Mode.		
Total direct IOPS	Indicates the rate at which this CSV performs IOPS in the Direct I/O Mode.	Operations/Sec	This is a good indicator of the level of I/O activity on the CSV in the Direct I/O Mode.
Redirected read rate	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV reads from the disk, in the Block Level Redirected I/O Mode. If the node being monitored is a non-coordinator node, then this measure indicates the rate at which this CSV reads from the disk by redirecting the read requests to the co-ordinator node via SMB, in the Block Level Redirected I/O Mode.	Reads/Sec	
Redirected write rate	If the node being monitored is a co-ordinator node, then this measure indicates the rate at which this CSV writes to the disk, in the Block Level Redirected	Writes/Sec	

Measurement	Description	Measurement Unit	Interpretation
	I/O Mode. If the node being monitored is a non- coordinator node, then this measure indicates the rate at which this CSV writes to the disk by redirecting the write requests to the co- ordinator node via SMB, in the Block Level Redirected I/O Mode.		
Total redirected IOPS	Indicates the rate at which I/O reads and writes were performed by this CSV on the disk, in the Block Level Redirected I/O Mode.	Operations/Sec	This is a good indicator of the level of I/O activity in the Block Level Redirected I/O Mode.
Read IOPS	Indicates the rate at which read I/O operations were performed on this CSV, both directly and via redirection - i.e., in the Direct I/O and Block Level Redirected I/O, Modes.	Reads/Sec	The value of this measure is the sum of the values of the *Direct read rate* and *Redirected read rate* measures.
Write IOPS	Indicates the rate at which write I/O operations were performed on this CSV, both directly and via redirection - i.e., in the Direct I/O and Block Level Redirected I/O	Writes/Sec	The value of this measure is the sum of the values of the *Direct write rate* and *Redirected write rate* measures.

Measurement	Description	Measurement Unit	Interpretation
	I/O Modes.		
IOPS	Indicates the rate at which read and write I/O operations were performed on this CSV, both directly and via redirection - i.e., in the Direct I/O and Block Level Redirected I/O.	Operations/Sec	The value of this measure is the sum of the values of the *Read IOPS* and *Write IOPS* measures. This is a good indicator of the level of I/O activity on the CSV.

4.4.4 Cluster Shared Volume Caches Test

Cluster Shared Volumes (CSV) Cache, also known as CSV Block Cache, is a feature which allows you to allocate system memory (RAM) as a write-through cache.

The CSV Cache provides caching of read-only unbuffered I/O. This can improve performance for applications such as Hyper-V, which conducts unbuffered I/O when accessing a VHD or VHDX file. That means that the server will cache virtual hard disk reads in RAM and hit that RAM instead of accessing the more latent disks on which the CSV is stored. Unbuffered I/O's are operations which are not cached by the Windows Cache Manager. What CSV Block Cache delivers is caching which can boost the performance of read requests, with write-through for no caching of write requests.

If enough memory is not allocated to the CSV block cache, then the cache reads will significantly drop, virtual hard disk reads will rise, and consequently, VM performance will deteriorate. If this is to be avoided, then administrators should continuously track cache usage, detect ineffective usage, and fine-tune the cache configuration so as to optimize cache usage and enhance VM performance. This is what the Cluster Shared Volume Caches test helps administrators achieve!!!

This test monitors the CSV Block Cache and reports the number and rate of reads from the cache. By keeping an eye on these usage metrics over time, administrators can proactively detect a consistent drop in cache hits, investigate the reasons for the same, and promptly initiate measures to improve cache usage and safeguard VM performance.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for every CSV on the server

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.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Cache read rate	Indicates the rate at which data is read from the cache.	KB/Sec	A consistent drop in the value of this measure could indicate a bottleneck in read request processing by the cache. Check whether adequate memory is available to the cache for servicing the read requests it receives. If not, increase the value of the **BlockCacheSize** cluster property to ensure that sufficient memory is available to the cache at all times. This property allows you to define how much memory (in megabytes) you wish to reserve for the CSV Cache on each node in the cluster. If a value of 512 is defined, then 512 MB of system memory will be reserved on each node in the Failover Cluster.
Cache reads	Indicates the number of times read requests were serviced by the cache.	Number	A steady decrease in the value of this measure is a cause for concern, as it indicates that cache misses are on a rise. This in turn may cause VM performance to deteriorate. To avoid this, check whether the BlockCacheSize cluster property has been set right. Typically, this property has to be set after considering the level of I/O activity on the cluster. Where there is a high

Measurement	Description	Measurement Unit	Interpretation
			level of read activity, sufficient memory should be available to the cache, so that cache reads are always high. In such environments, you may want to increase the value of the BlockCacheSize property. Cache usage can also be optimized using the EnableBlockCache property. This is a private property of the cluster Physical Disk resource. It allows you to enable/disable caching on an individual disk. This gives the flexibility to configure cache for read intensive VMs running on some disks, while allowing to disable and prevent random I/O on other disks from purging the cache.

4.4.5 SMB Server Share Performance Test

CSV uses Server Message Block (SMB) protocol to redirect traffic using File System Redirected IO or Block Level Redirected IO to the Coordinator Node. For a non-Coordinator Node to redirect I/O to the Coordinator Node, SMB shares are required on the Coordinator Node. There is a hidden admin share that is created for CSV, shared as ClusterStorage$. This share is created by the cluster to facilitate remote administration. You should use it in the scenarios where you would normally use an admin share on any other volume (such as D$). There are also a couple of hidden shares that are used by the CSV. These shares are used only on the Coordinator Node. Other nodes either do not have these shares or these shares are not used: Each Cluster Shared Volume hosted on a Coordinator Node creates a share with a name that looks like a GUID. This is used by CsvFs (CSV File System) to communicate with the hidden CSV NTFS stack on the Coordinator Node. This share points to the hidden NTFS volume used by CSV. Metadata and the File System Redirected IO are flowing to the Coordinator Node using this share. On the Coordinator Node you also will see a share with the name CSV$. This share is used to forward Block Level Redirected IO to the Coordinator Node. There is only one CSV$ share on every Coordinator Node. Users are not expected to use these shares – they are access-controlled so that only Local System and Failover Cluster Identity user (CLIUSR) have access to the share.

All of these shares are temporary – information about these shares is not in any persistent storage, and when node reboots they will be removed from the Server Service. Cluster takes care of creating the shares every time during CSV start up.

In addition, users can create folders on any node in the cluster and share them with one/more other nodes in the cluster, so that users connecting from such nodes can read from and/or write to the folder at will. Such folders are also called SMB shares.

If a user to a VM in a Hyper-V cluster complains of slowness when accessing the VM, then administrators may want to know if SMB shares on that cluster node are receiving I/O traffic from other nodes, and if so, whether/not that transmission is what is taking time and slowing down the user access! This is exactly what the SMB Server Share Performance test helps administrators achieve! For a Coordinator Node, this test auto-discovers the following:

- SMB shares on the Coordinator Node to which the slave nodes redirect I/O in the File System Redirected IO or Block Level Redirected IO mode;
- SMB shares (if any) created on the Coordinator Node to which one/more other slave nodes in the cluster send I/O.

For a non-coordinator node, this test auto-discovers the SMB shares (if any) on that node to which other nodes in the cluster send I/O.

The test then monitors the traffic to each SMB server share and reveals which share is seeing maximum traffic and how latent the transmission is.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for every SMB server share on the monitored node

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.

Measurements reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Read throughput	Indicates the rate at which data is being read from this share.	Kbps	
Write throughput	Indicates the rate at which data is being written to this share.	Kbps	
Throughput	Indicates the rate at which data is being read from and written to this share.	Kbps	The value of this measure is the sum of the values of the *Read throughput* and *Write throughput* measures. This is a good indicator of the level of traffic to a share. By comparing the value of this measure across shares, you can identify the share that is seeing the maximum traffic. If this max value is abnormally high for a share, you may want to investigate the reasons for the same.
Read rate	Indicates the rate at which read requests are being sent to this share.	Requests/Sec	
Write rate	Indicates the rate at which write requests are being sent to this share.	Requests/Sec	
IOPS	Indicates the rate at which read and write requests are being sent to this share.	Requests/Sec	The value of this measure is the sum of the values of the *Read rate* and *Write rate* measures. This is a good indicator of the level of traffic to a share. By comparing the value of this measure across

Measurement	Description	Measurement Unit	Interpretation
			shares, you can identify the share that is seeing the maximum traffic. If this max value is abnormally high for a share, you may want to investigate the reasons for the same.
Read latency	Indicates the average latency between when a read request is sent to this share and when a response is received from this share.	Secs	Ideally, the value of this measure should be low.
Write latency	Indicates the average latency between when a write request is sent to this share and when a response is received from this share.	Secs	Ideally, the value of this measure should be low.
Latency	Indicates the average latency of both read and write operations performed by this share.	Secs	A low value is desired for this measure. A consistent rise in the value of the measure is an indicator of a processing bottleneck on the share. The value of this measure is the sum of the values of the *Read latency* and *Write latency* measures.
Read IO	Indicates the average number of bytes read from this share.	Number	
Write IO	Indicates the average number of bytes written	Number	

Measurement	Description	Measurement Unit	Interpretation
	to this share.		
IO reads and writes	Indicates the average number of bytes read from and written to this share.	Number	The value of this measure is the sum of the values of the *Read IO* and *Write IO* measures. This is a good indicator of the workload of a share.
Read queue length	Indicates the count of read operations that are currently outstanding on this share.	Number	
Write queue length	Indicates the count of write operations that are currently outstanding on this share.	Number	
Queue length	Indicates the count of read and write operations that are currently outstanding on this share.	Number	The value of this measure is the sum of the values of the *Read queue length* and *Write queue length* measures. A zero value is desired for this measure. A consistent increase in the value of this measure is a cause for concern, as it indicates that the share is having difficulty processing I/O requests.

4.4.6 SMB Client Share Performance Test

CSV uses Server Message Block (SMB) protocol to redirect traffic using File System Redirected IO or Block Level Redirected IO to the Coordinator Node. For a non-Coordinator Node to redirect I/O to the Coordinator Node, SMB shares are required on the Coordinator Node. There is a hidden admin share that is created for CSV, shared as ClusterStorage$. This share is created by the cluster to facilitate remote administration. You should use it in the scenarios where you would normally use an admin share on any other volume (such as D$). There are also a couple of hidden shares that are used by the CSV. These shares are used only on the Coordinator Node. Other nodes either do not have these shares or these shares are not used: Each Cluster Shared Volume hosted on a Coordinator Node creates a share with a name that looks like a GUID. This is used by CsvFs (CSV File System) to communicate with the hidden CSV NTFS stack on the Coordinator Node. This share points to the hidden NTFS volume used by CSV. Metadata and the File System Redirected IO are flowing to the Coordinator Node using this share. On the Coordinator Node you also will see a share with the name CSV$. This share is used to forward Block Level Redirected IO to the Coordinator Node. There is only one CSV$ share on every Coordinator Node. Users are not expected to use these shares – they are access-controlled so that only Local System and Failover Cluster Identity user (CLIUSR) have access to the share.

All of these shares are temporary – information about these shares is not in any persistent storage, and when node reboots they will be removed from the Server Service. Cluster takes care of creating the shares every time during CSV start up.

In addition, users can create folders on any node in the cluster and share them with one/more other nodes in the cluster, so that users connecting from such nodes can read from and/or write to the folder at will. Such folders are also called SMB shares.

If a user to a VM in a Hyper- V cluster complains of slowness when accessing the VM, then administrators may want to know if that cluster node is transmitting I/O to a remote SMB share, and if so, whether/not that transmission is what is taking time and slowing down the user access! This is exactly what the SMB Client Share Performance test helps administrators achieve! For a non-coordinator node, this test auto-discovers the following:

- SMB shares on the Coordinator Node to which that node redirects I/Os in the File System Redirected IO or Block Level Redirected IO mode;
- SMB shares (if any) created on any node in the cluster, to which the monitored non-coordinator node has access.

For a Coordinator Node, this test auto-discovers only the SMB shares (if any) created on other nodes in the cluster to which the Coordinator Node sends I/O.

The test then monitors the traffic to each SMB client share and reveals which share is seeing maximum traffic and how latent the transmission is.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for every SMB client share to which the monitored node sends I/O

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.

Measurements reported by the test:

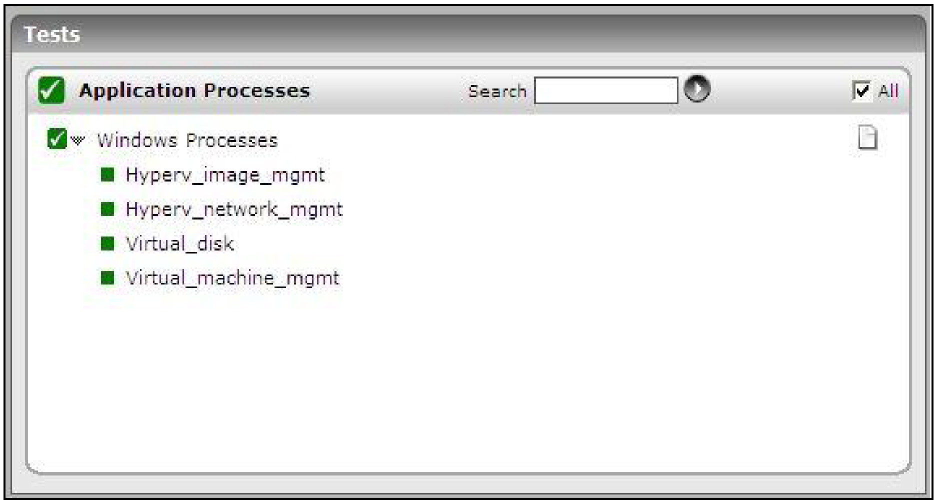
Measurement	Description	Measurement Unit	Interpretation
Read throughput	Indicates the rate at which data is being read from this share.	Kbps	
Write throughput	Indicates the rate at which data is being written to this share.	Kbps	
Throughput	Indicates the rate at which data is being read from and written to this share.	Kbps	The value of this measure is the sum of the values of the *Read throughput* and *Write throughput* measures. This is a good indicator of the level of traffic to a share. By comparing the value of this measure across shares, you can identify the share that is seeing the maximum traffic. If this max value is abnormally high for a share, you may want to investigate the reasons for the same.
Read rate	Indicates the rate at which read requests are being sent to this share.	Requests/Sec	
Write rate	Indicates the rate at which write requests are being sent to this share.	Requests/Sec	
IOPS	Indicates the rate at which read and write	Requests/Sec	The value of this measure is the sum of the values of the *Read rate*

Measurement	Description	Measurement Unit	Interpretation
	requests are being sent to this share.		and *Write rate* measures. This is a good indicator of the level of traffic to a share. By comparing the value of this measure across shares, you can identify the share that is seeing the maximum traffic. If this max value is abnormally high for a share, you may want to investigate the reasons for the same.
Read latency	Indicates the average latency between when a read request is sent to this share and when a response is received from this share.	Secs	Ideally, the value of this measure should be low.
Write latency	Indicates the average latency between when a write request is sent to this share and when a response is received from this share.	Secs	Ideally, the value of this measure should be low.
Latency	Indicates the average latency of both read and write operations performed by this share.	Secs	A low value is desired for this measure. A consistent rise in the value of the measure is an indicator of a processing bottleneck on the share. The value of this measure is the sum of the values of the *Read latency* and *Write latency* measures.
Read IO	Indicates the average	Number	

Measurement	Description	Measurement Unit	Interpretation
	number of bytes read from this share.		
Write IO	Indicates the average number of bytes written to this share.	Number	
IO reads and writes	Indicates the average number of bytes read from and written to this share.	Number	The value of this measure is the sum of the values of the *Read IO* and *Write IO* measures. This is a good indicator of the workload of a share.
Read queue length	Indicates the count of read operations that are currently outstanding on this share.	Number	
Write queue length	Indicates the count of write operations that are currently outstanding on this share.	Number	
Queue length	Indicates the count of read and write operations that are currently outstanding on this share.	Number	The value of this measure is the sum of the values of the *Read queue length* and *Write queue length* measures. A zero value is desired for this measure. A consistent increase in the value of this measure is a cause for concern, as it indicates that the share is having difficulty processing I/O requests.

4.5 The Application Processes Layer

The *WindowsProcesses* layer monitors the processes critical to the smooth functioning of the Hyer-V server, and also measures the resource footprint of these key processes.



Tests

☑ Application Processes Search ☑ All

☑ Windows Processes

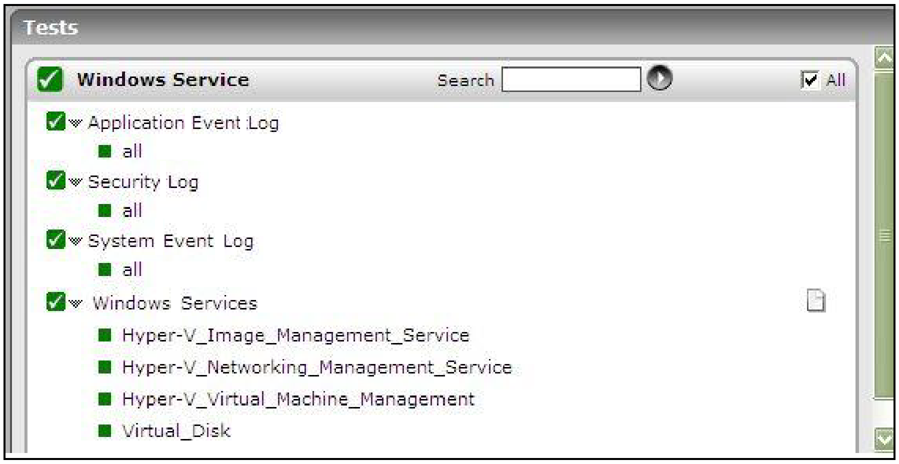
- Hyperv_image_mgmt
- Hyperv_network_mgmt
- Virtual_disk
- Virtual_machine_mgmt

Figure 4.7: The tests associated with the Application Processes layer

Since these tests too have been dealt with in the *Monitoring Unix and Windows Servers* document, let us proceed to the *Windows Services* layer

4.6 The Windows Service Layer

The *Windows Service* layer captures the applications, system, and security errors/warning events logged in the Windows event logs , and also reveals whether the core Hyper-V services are currently available or not. In addition, the layer also monitors useful Hyper-V-specific event logs and captures errors related virtual machine configurations, Hyper-V clustering, Hyper-V integration and virtual machine management services, and Hyper-V worker processes.



Tests

☑ Windows Service Search ☑ All

☑ Application Event Log

- all

☑ Security Log

- all

☑ System Event Log

- all

☑ Windows Services

- Hyper-V_Image_Management_Service
- Hyper-V_Networking_Management_Service
- Hyper-V_Virtual_Machine_Management
- Virtual_Disk

Figure 4.8: The tests mapped to the Windows Service layer

Since most of the tests mapped to this layer have been elaborately discussed in the *Monitoring Unix and Windows Servers* document, we will only be discussing the Hyper-V event log tests in this section.

4.6.1 Hyper-V Config Admin Log Test

The *Hyper-V Config* logs help troubleshoot issues related to virtual machine configuration files. For instance, if you have a missing or corrupt virtual machine configuration file, the entries in the *Hyper-V Config* logs will shed light on it. Using the Hyper-V Config Admin Log test, you can be alerted if any error/warning event is captured by the *Hyper-V Config logs* and can view the complete details of these events without accessing the event logs for it.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the **FILTER** configured

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 –** Refers to the port used by the EventLog Service. Here it is null.
4. **LOGTYPE** – Refers to the type of event logs to be monitored. The default value is Microsoft-Windows-Hyper-V-Config-Admin.
5. **POLICY BASED FILTER** - Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options:

   - Manually specify the event sources, IDs, and descriptions in the **FILTER** text area, or,
   - Select a specification from the predefined filter policies listed in the **FILTER** box

   For explicit, manual specification of the filter conditions, select the **NO** option against the **POLICY BASED FILTER** field. This is the default selection. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **YES** option against the **POLICY BASED FILTER** field.

6. **FILTER** - If the **POLICY BASED FILTER** flag is set to **NO**, then a **FILTER** text area will appear, wherein you will have to specify the event sources, event IDs, and event descriptions to be monitored. This specification should be of the following format: *{Displayname}:{event_sources_to_be_included}:*

{event_ sources_ to_ be_ excluded}: {event_ IDs_ to_ be_ included}: {event_ IDs_ to_ be_ excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}. For example, assume that the **FILTER** text area takes the value, *OS_ events:all:Browse,Print:all:none:all:none*. Here:

- *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI;
- *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the event sources are monitored, specify *none*.
- In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The *all* in our example represents that all the event IDs need to be considered while monitoring.
- Similarly, the *none* (following *all* in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs as a comma-separated list. Likewise, specifying *all* makes sure that all the event IDs are excluded from monitoring.
- The *all* which follows implies that all events, regardless of description, need to be included for monitoring. To exclude all events, use *none*. On the other hand, if you provide a comma-separated list of event descriptions, then the events with the specified descriptions will alone be monitored. Event descriptions can be of any of the following forms - desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
- In the same way, you can also provide a comma-separated list of event descriptions to be excluded from monitoring. Here again, the specification can be of any of the following forms: desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. In our example however, none is specified, indicating that no event descriptions are to be excluded from monitoring. If you use all instead, it would mean that all event descriptions are to be excluded from monitoring.

By default, the **FILTER** parameter contains the value: *all:all:none:all:none:all:none*. Multiple filters are to be separated by semi-colons (;).

Note:

The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window.

On the other hand, if the **POLICY BASED FILTER** flag is set to **YES**, then a **FILTER** list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is built into the policy in the following format:

{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}

To monitor a specific combination of event sources, event IDs, and event descriptions, you can choose the corresponding filter policy from the **FILTER** list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link appears just above the test configuration section, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one. The changed policy or the new policy can then be associated with the test by selecting the policy name from the **FILTER** list box in this page.

7. **USEWMI** - The eG agent can either use WMI to extract event log statistics or directly parse the event logs using event log APIs. If the **USEWMI** flag is **YES**, then WMI is used. If not, the event log APIs are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the WinMgmt process to shoot up. On such systems, set the **USEWMI** parameter value to **NO**. **On the other hand, when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the USEWMI flag should always be set to 'Yes'.**
8. **STATELESS ALERTS** - Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the EventLog test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To

make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the **STATELESS ALERTS** flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.

9. **EVENTS DURING RESTART** - By default, the **EVENTS DURING RESTART** flag is set to **Yes**. This ensures that whenever the agent is stopped and later started, the events that might have occurred during the period of non-availability of the agent are included in the number of events reported by the agent. Setting the flag to **No** ensures that the agent, when restarted, ignores the events that occurred during the time it was not available.

10. **DDFORINFORMATION** – eG Enterprise also provides you with options to restrict the amount of storage required for event log tests. Towards this end, the **DDFORINFORMATION** and **DDFORWARNING** flags have been made available in this page. By default, both these flags are set to **YES**, indicating that by default, the test generates detailed diagnostic measures for information events and warning events. If you do not want the test to generate and store detailed measures for information events, set the **DDFORINFORMATION** flag to **No**.

11. **DDFORWARNING –** To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **NO**.

12. **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**.

13. **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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Error messages	This refers to the number of error events that were generated.	Number	A very low value (zero) indicates that no problems exist in the virtual machine configuration files. An increasing trend or high value indicates the existence of problems. Please check the Hyper-V COnfig Logs in the Event Log Viewer for more details.
Information messages	This refers to the number of information events generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed. Please check the Hyper-V Config Logs in the Event Log Viewer for more details.
Warnings	This refers to the number of warnings that were generated when the test was last executed.	Number	A high value of this measure indicates v problems with virtual machine configuration files that may not have an immediate impact, but may cause future problems. Please check the Hyper-V Config Logs in the Event Log Viewer for more details.
Critical messages	Indicates the number of critical events that were generated when the test was last executed.	Number	A critical event is one that a virtual machine configuration file cannot automatically recover from. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**.

Measurement	Description	Measurement Unit	Interpretation
			A very low value (zero) indicates good health. An increasing trend or high value indicates the existence of fatal/irrepairable problems. The detailed diagnosis of this measure describes all the critical events that were generated during the last measurement period. Please check the Hyper-V Config logs in the Event Log Viewer for more details.
Verbose messages	Indicates the number of verbose events that were generated when the test was last executed.	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues better. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the Hyper-V Config Logs in the Event Log Viewer for more details.

4.6.2 Hyper-V High Availability Admin Log Test

The *Hyper-V High Availability logs* shed light on the actions and changes that take place because of Hyper-V clustering. Using the **Hyper-V High Availability Admin Log** test, you can be alerted if any error/warning event is captured by the *Hyper-V High Availability logs* and can view the complete details of these events without accessing the event logs for it.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the **FILTER** configured

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 –** Refers to the port used by the EventLog Service. Here it is null.
4. **LOGTYPE** – Refers to the type of event logs to be monitored. The default value is Microsoft-Windows-Hyper-V-High-Availability-Admin.
5. **POLICY BASED FILTER** - Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options:

   - Manually specify the event sources, IDs, and descriptions in the **FILTER** text area, or,
   - Select a specification from the predefined filter policies listed in the **FILTER** box

   For explicit, manual specification of the filter conditions, select the **NO** option against the **POLICY BASED FILTER** field. This is the default selection. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **YES** option against the **POLICY BASED FILTER** field.

6. **FILTER** - If the **POLICY BASED FILTER** flag is set to **NO**, then a **FILTER** text area will appear, wherein you will have to specify the event sources, event IDs, and event descriptions to be monitored. This specification should be of the following format: *{Displayname}:{event_sources_to_be_included}:{event_ sources_ to_ be_ excluded}: {event_ IDs_ to_ be_ included}: {event_ IDs_ to_ be_ excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}*. For example, assume that the **FILTER** text area takes the value, *OS_ events:all:Browse,Print:all:none:all:none*. Here:

   - *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI;
   - *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the

event sources are monitored, specify *none*.

- In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The *all* in our example represents that all the event IDs need to be considered while monitoring.
- Similarly, the *none* (following *all* in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs as a comma-separated list. Likewise, specifying *all* makes sure that all the event IDs are excluded from monitoring.
- The all which follows implies that all events, regardless of description, need to be included for monitoring. To exclude all events, use none. On the other hand, if you provide a comma-separated list of event descriptions, then the events with the specified descriptions will alone be monitored. Event descriptions can be of any of the following forms - desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
- In the same way, you can also provide a comma-separated list of event descriptions to be excluded from monitoring. Here again, the specification can be of any of the following forms: desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. In our example however, none is specified, indicating that no event descriptions are to be excluded from monitoring. If you use all instead, it would mean that all event descriptions are to be excluded from monitoring.

By default, the **FILTER** parameter contains the value: *all:all:none:all:none:all:none*. Multiple filters are to be separated by semi-colons (;).

Note:

The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window.

On the other hand, if the **POLICY BASED FILTER** flag is set to **YES**, then a **FILTER** list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is built into the policy in the following format:

{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}

To monitor a specific combination of event sources, event IDs, and event descriptions, you can choose the corresponding filter policy from the **FILTER** list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link appears just above the test configuration section, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one. The changed policy or the new policy can then be associated with the test by selecting the policy name from the **FILTER** list box in this page.

7. **USEWMI** - The eG agent can either use WMI to extract event log statistics or directly parse the event logs using event log APIs. If the **USEWMI** flag is **YES**, then WMI is used. If not, the event log APIs are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the WinMgmt process to shoot up. On such systems, set the **USEWMI** parameter value to **NO**. **On the other hand, when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the USEWMI flag should always be set to 'Yes'.**
8. **STATELESS ALERTS -** Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the EventLog test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the **STATELESS ALERTS** flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.
9. **EVENTS DURING RESTART -** By default, the **EVENTS DURING RESTART** flag is set to **Yes**. This ensures that whenever the agent is stopped and later started, the events that might have occurred during the period of non-availability of the agent are included in the number of events reported by the agent. Setting the flag to **No** ensures that the agent, when restarted, ignores the events that occurred during the time it was not available.

10. **DDFORINFORMATION** – eG Enterprise also provides you with options to restrict the amount of storage required for event log tests. Towards this end, the **DDFORINFORMATION** and **DDFORWARNING** flags have been made available in this page. By default, both these flags are set to **YES**, indicating that by default, the test generates detailed diagnostic measures for information events and warning events. If you do not want the test to generate and store detailed measures for information events, set the **DDFORINFORMATION** flag to **No**.

11. **DDFORWARNING** – To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **NO**.

12. **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**.

13. **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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Error messages	This refers to the number of error events that were generated.	Number	A very low value (zero) indicates that no problems exist in Hyper-V clustering. An increasing trend or high value indicates the existence of problems. Please check the *Hyper-V High*

Measurement	Description	Measurement Unit	Interpretation
			Availability Logs in the Event Log Viewer for more details.
Information messages	This refers to the number of information events generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed. Please check the *Hyper- V High Availability Logs* in the Event Log Viewer for more details.
Warning	This refers to the number of warnings that were generated when the test was last executed.	Number	A high value of this measure indicates problems with Hyper- V clustering that may not have an immediate impact, but may cause future problems. Please check the *Hyper- V High Availability Logs* in the Event Log Viewer for more details.
Critical messages	Indicates the number of critical events that were generated when the test was last executed.	Number	A critical event is one that Hyper-V cannot automatically recover from. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. A very low value (zero) indicates good health. An increasing trend or high value indicates the existence of fatal/irrepairable problems. The detailed diagnosis of this measure describes all the critical events that were generated during the last measurement period.

Measurement	Description	Measurement Unit	Interpretation
			Please check the *Hyper-V High Availability logs* in the Event Log Viewer for more details.
Verbose messages	Indicates the number of verbose events that were generated when the test was last executed.	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues better. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the *Hyper-V High Availability Logs* in the Event Log Viewer for more details.

4.6.3 Hyper-V Integration Admin Log Test

The *Hyper-V Integration logs* serve as useful source of information for analyzing errors, warnings, and general details related to the Hyper-V integration services. Using the **Hyper-V Integration Admin Log** test, you can be alerted if any error/warning event is captured by the Hyper-V Integration logs and can view the complete details of these events without accessing the event logs for it.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the **FILTER** configured

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 –** Refers to the port used by the EventLog Service. Here it is null.

4. **LOGTYPE** – Refers to the type of event logs to be monitored. The default value is Microsoft-Windows-Hyper-V-Integration-Admin.

5. **POLICY BASED FILTER** - Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options:

   - Manually specify the event sources, IDs, and descriptions in the **FILTER** text area, or,
   - Select a specification from the predefined filter policies listed in the **FILTER** box

   For explicit, manual specification of the filter conditions, select the **NO** option against the **POLICY BASED FILTER** field. This is the default selection. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **YES** option against the **POLICY BASED FILTER** field.

6. **FILTER** - If the **POLICY BASED FILTER** flag is set to **NO**, then a **FILTER** text area will appear, wherein you will have to specify the event sources, event IDs, and event descriptions to be monitored. This specification should be of the following format: *{Displayname}:{event_sources_to_be_included}:{event_ sources_ to_ be_ excluded}: {event_ IDs_ to_ be_ included}: {event_ IDs_ to_ be_ excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}*. For example, assume that the **FILTER** text area takes the value, *OS_ events:all:Browse,Print:all:none:all:none*. Here:

   - *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI;
   - *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the event sources are monitored, specify *none*.
   - In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The *all* in our example represents that all the event IDs need to be considered while monitoring.
   - Similarly, the *none* (following *all* in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs as a comma-separated list. Likewise, specifying *all* makes sure that all the event IDs are excluded from monitoring.
   - The all which follows implies that all events, regardless of description, need to be included

for monitoring. To exclude all events, use none. On the other hand, if you provide a comma-separated list of event descriptions, then the events with the specified descriptions will alone be monitored. Event descriptions can be of any of the following forms - desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.

- In the same way, you can also provide a comma-separated list of event descriptions to be excluded from monitoring. Here again, the specification can be of any of the following forms: desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. In our example however, none is specified, indicating that no event descriptions are to be excluded from monitoring. If you use all instead, it would mean that all event descriptions are to be excluded from monitoring.

By default, the **FILTER** parameter contains the value: *all:all:none:all:none:all:none*. Multiple filters are to be separated by semi-colons (;).

Note:

The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window.

On the other hand, if the **POLICY BASED FILTER** flag is set to **YES**, then a **FILTER** list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is built into the policy in the following format:

{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}

To monitor a specific combination of event sources, event IDs, and event descriptions, you can choose the corresponding filter policy from the **FILTER** list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link appears just above the test configuration section, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one. The changed policy or the new policy can then be associated with the test by selecting the policy name from the **FILTER** list box in this page.

7. **USEWMI** - The eG agent can either use WMI to extract event log statistics or directly parse the event logs using event log APIs. If the **USEWMI** flag is **YES**, then WMI is used. If not, the event log APIs are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the WinMgmt process to shoot up. On such systems, set the **USEWMI** parameter value to **NO**. **On the other hand, when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the USEWMI flag should always be set to 'Yes'.**

8. **STATELESS ALERTS -** Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the EventLog test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the **STATELESS ALERTS** flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.

9. **EVENTS DURING RESTART -** By default, the **EVENTS DURING RESTART** flag is set to **Yes**. This ensures that whenever the agent is stopped and later started, the events that might have occurred during the period of non-availability of the agent are included in the number of events reported by the agent. Setting the flag to **No** ensures that the agent, when restarted, ignores the events that occurred during the time it was not available.

10. **DDFORINFORMATION –** eG Enterprise also provides you with options to restrict the amount of storage required for event log tests. Towards this end, the **DDFORINFORMATION** and **DDFORWARNING** flags have been made available in this page. By default, both these flags are set to **YES**, indicating that by default, the test generates detailed diagnostic measures for information events and warning events. If you do not want the test to generate and store detailed measures for information events, set the **DDFORINFORMATION** flag to **No**.

11. **DDFORWARNING –** To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **NO**.

12. **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**.

13. **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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Error messages	This refers to the number of error events that were generated.	Number	A very low value (zero) indicates that no problems exist in integration services. An increasing trend or high value indicates the existence of problems. Please check the Hyper- V Integration Logs in the Event Log Viewer for more details.
Information messages	This refers to the number of information events generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed. Please check the *Hyper- V Integration Logs* in the Event Log Viewer for more details.

Measurement	Description	Measurement Unit	Interpretation
Warnings	This refers to the number of warnings that were generated when the test was last executed.	Number	A high value of this measure indicates problems with Hyper-V integration services that may not have an immediate impact, but may cause future problems. Please check the *Hyper-V Integration Logs* in the Event Log Viewer for more details.
Critical messages	Indicates the number of critical events that were generated when the test was last executed.	Number	A critical event is one that the Hyper-V integration services cannot automatically recover from. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. A very low value (zero) indicates good health. An increasing trend or high value indicates the existence of fatal/irrepairable problems. The detailed diagnosis of this measure describes all the critical events that were generated during the last measurement period. Please check the *Hyper-V Integration logs* in the Event Log Viewer for more details.
Verbose messages	Indicates the number of verbose events that were generated when the test was last executed.	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues better.

Measurement	Description	Measurement Unit	Interpretation
			This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the *Hyper- V Integration Logs* in the Event Log Viewer for more details.

4.6.4 Hyper-V VMMS Storage Log Test

The *Hyper-V VMMS logs* capture storage-related issues. Using the **Hyper-V VMMS Storage Log** test, you can be alerted if any error/warning event is captured by the *Hyper-V VMMS logs* and can view the complete details of these events without accessing the event logs for it.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the **FILTER** configured

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 –** Refers to the port used by the EventLog Service. Here it is null.
4. **LOGTYPE** – Refers to the type of event logs to be monitored. The default value is Microsoft-Windows-Hyper-V-VMMS-Storage.
5. **POLICY BASED FILTER** - Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options:

   - Manually specify the event sources, IDs, and descriptions in the **FILTER** text area, or,

- Select a specification from the predefined filter policies listed in the **FILTER** box

For explicit, manual specification of the filter conditions, select the **NO** option against the **POLICY BASED FILTER** field. This is the default selection. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **YES** option against the **POLICY BASED FILTER** field.

6. **FILTER** - If the **POLICY BASED FILTER** flag is set to **NO**, then a **FILTER** text area will appear, wherein you will have to specify the event sources, event IDs, and event descriptions to be monitored. This specification should be of the following format: *{Displayname}:{event_sources_to_be_included}:{event_ sources_ to_ be_ excluded}: {event_ IDs_ to_ be_ included}: {event_ IDs_ to_ be_ excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}*. For example, assume that the **FILTER** text area takes the value, *OS_ events:all:Browse,Print:all:none:all:none*. Here:

- *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI;
- *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the event sources are monitored, specify *none*.

- In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The *all* in our example represents that all the event IDs need to be considered while monitoring.
- Similarly, the *none* (following *all* in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs as a comma-separated list. Likewise, specifying *all* makes sure that all the event IDs are excluded from monitoring.
- The all which follows implies that all events, regardless of description, need to be included for monitoring. To exclude all events, use none. On the other hand, if you provide a comma-separated list of event descriptions, then the events with the specified descriptions will alone be monitored. Event descriptions can be of any of the following forms - desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
- In the same way, you can also provide a comma-separated list of event descriptions to be excluded from monitoring. Here again, the specification can be of any of the following forms: desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while

a trailing '*' signifies any number of trailing characters. In our example however, none is specified, indicating that no event descriptions are to be excluded from monitoring. If you use all instead, it would mean that all event descriptions are to be excluded from monitoring.

By default, the **FILTER** parameter contains the value: *all:all:none:all:none:all:none*. Multiple filters are to be separated by semi-colons (;).

Note:

The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window.

On the other hand, if the **POLICY BASED FILTER** flag is set to **YES**, then a **FILTER** list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is built into the policy in the following format:

{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}

To monitor a specific combination of event sources, event IDs, and event descriptions, you can choose the corresponding filter policy from the **FILTER** list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link appears just above the test configuration section, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one. The changed policy or the new policy can then be associated with the test by selecting the policy name from the **FILTER** list box in this page.

7. **USEWMI** - The eG agent can either use WMI to extract event log statistics or directly parse the event logs using event log APIs. If the **USEWMI** flag is **YES**, then WMI is used. If not, the event log APIs are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the WinMgmt process to shoot up. On such systems, set the **USEWMI** parameter value to **NO**. **On the other hand, when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the USEWMI flag should always be set to 'Yes'.**
8. **STATELESS ALERTS -** Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a

measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the EventLog test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the **STATELESS ALERTS** flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.

9. **EVENTS DURING RESTART -** By default, the **EVENTS DURING RESTART** flag is set to **Yes**. This ensures that whenever the agent is stopped and later started, the events that might have occurred during the period of non-availability of the agent are included in the number of events reported by the agent. Setting the flag to **No** ensures that the agent, when restarted, ignores the events that occurred during the time it was not available.

10. **DDFORINFORMATION** – eG Enterprise also provides you with options to restrict the amount of storage required for event log tests. Towards this end, the **DDFORINFORMATION** and **DDFORWARNING** flags have been made available in this page. By default, both these flags are set to **YES**, indicating that by default, the test generates detailed diagnostic measures for information events and warning events. If you do not want the test to generate and store detailed measures for information events, set the **DDFORINFORMATION** flag to **No**.

11. **DDFORWARNING –** To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **NO**.

12. **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**.

13. **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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Error messages	This refers to the number of error events that were generated.	Number	A very low value (zero) indicates that no problems exist in the storage. An increasing trend or high value indicates the existence of problems. Please check the *Hyper-V VMMS Logs* in the Event Log Viewer for more details.
Information messages	This refers to the number of information events generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed. Please check the *Hyper-V VMMS Logs* in the Event Log Viewer for more details.
Warnings	This refers to the number of warnings that were generated when the test was last executed.	Number	A high value of this measure indicates problems with storage that may not have an immediate impact, but may cause future problems. Please check the Hyper-V VMMS Logs in the Event Log Viewer for more details.
Critical messages	Indicates the number of	Number	A critical event is one that the

Measurement	Description	Measurement Unit	Interpretation
	critical events that were generated when the test was last executed.		storage cannot automatically recover from. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. A very low value (zero) indicates good health. An increasing trend or high value indicates the existence of fatal/irrepairable problems. The detailed diagnosis of this measure describes all the critical events that were generated during the last measurement period. Please check the *Hyper-V VMMS logs* in the Event Log Viewer for more details.
Verbose messages	Indicates the number of verbose events that were generated when the test was last executed.	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues better. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the *Hyper-V VMMS logs* in the Event Log Viewer for more details.

4.6.5 Hyper-V VMMS Admin Log Test

The *Hyper-V VMMS logs* capture events related to the virtual machine management services. Using the **Hyper-V VMMS Admin Log** test, you can be alerted if any error/warning event is captured by the *Hyper-V VMMS logs* and can view the complete details of these events without accessing the event logs for it.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the **FILTER** configured

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** – Refers to the port used by the EventLog Service. Here it is null.
4. **LOGTYPE** – Refers to the type of event logs to be monitored. The default value is Microsoft-Windows-Hyper-V-VMMS-Admin.
5. **POLICY BASED FILTER -** Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options:

   - Manually specify the event sources, IDs, and descriptions in the **FILTER** text area, or,
   - Select a specification from the predefined filter policies listed in the **FILTER** box

   For explicit, manual specification of the filter conditions, select the **NO** option against the **POLICY BASED FILTER** field. This is the default selection. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **YES** option against the **POLICY BASED FILTER** field.

6. **FILTER** - If the **POLICY BASED FILTER** flag is set to **NO**, then a **FILTER** text area will appear, wherein you will have to specify the event sources, event IDs, and event descriptions to be monitored. This specification should be of the following format: *{Displayname}:{event_sources_to_be_included}:{event_ sources_ to_ be_ excluded}: {event_ IDs_ to_ be_ included}: {event_ IDs_ to_ be_ excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}*. For example, assume that the **FILTER** text area takes the value, *OS_ events:all:Browse,Print:all:none:all:none*. Here:

- *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI;
- *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the event sources are monitored, specify *none*.
- In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The *all* in our example represents that all the event IDs need to be considered while monitoring.
- Similarly, the *none* (following *all* in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs as a comma-separated list. Likewise, specifying *all* makes sure that all the event IDs are excluded from monitoring.
- The all which follows implies that all events, regardless of description, need to be included for monitoring. To exclude all events, use none. On the other hand, if you provide a comma-separated list of event descriptions, then the events with the specified descriptions will alone be monitored. Event descriptions can be of any of the following forms - desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
- In the same way, you can also provide a comma-separated list of event descriptions to be excluded from monitoring. Here again, the specification can be of any of the following forms: desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. In our example however, none is specified, indicating that no event descriptions are to be excluded from monitoring. If you use all instead, it would mean that all event descriptions are to be excluded from monitoring.

By default, the **FILTER** parameter contains the value: *all:all:none:all:none:all:none*. Multiple filters are to be separated by semi-colons (;).

Note:

The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window.

On the other hand, if the **POLICY BASED FILTER** flag is set to **YES**, then a **FILTER** list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is built into the policy in the following format:

{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}

To monitor a specific combination of event sources, event IDs, and event descriptions, you can choose the corresponding filter policy from the **FILTER** list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link appears just above the test configuration section, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one. The changed policy or the new policy can then be associated with the test by selecting the policy name from the **FILTER** list box in this page.

7. **USEWMI -** The eG agent can either use WMI to extract event log statistics or directly parse the event logs using event log APIs. If the **USEWMI** flag is **YES**, then WMI is used. If not, the event log APIs are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the WinMgmt process to shoot up. On such systems, set the **USEWMI** parameter value to **NO**. **On the other hand, when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the USEWMI flag should always be set to 'Yes'**.

8. **STATELESS ALERTS -** Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the EventLog test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the **STATELESS ALERTS** flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.

9. **EVENTS DURING RESTART** - By default, the **EVENTS DURING RESTART** flag is set to **Yes**. This ensures that whenever the agent is stopped and later started, the events that might have occurred during the period of non-availability of the agent are included in the number of events reported by the agent. Setting the flag to **No** ensures that the agent, when restarted, ignores the events that occurred during the time it was not available.

10. **DDFORINFORMATION** – eG Enterprise also provides you with options to restrict the amount of storage required for event log tests. Towards this end, the **DDFORINFORMATION** and **DDFORWARNING** flags have been made available in this page. By default, both these flags are set to **YES**, indicating that by default, the test generates detailed diagnostic measures for information events and warning events. If you do not want the test to generate and store detailed measures for information events, set the **DDFORINFORMATION** flag to **No**.

11. **DDFORWARNING** – To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **NO**.

12. **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**.

13. **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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Error messages	This refers to the number of error events that were generated.	Number	A very low value (zero) indicates that no problems exist in the virtual machine management services.

Measurement	Description	Measurement Unit	Interpretation
			An increasing trend or high value indicates the existence of problems. Please check the *Hyper-V VMMS Logs* in the Event Log Viewer for more details.
Information messages	This refers to the number of information events generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed. Please check the *Hyper-V VMMS Logs* in the Event Log Viewer for more details.
Warnings	This refers to the number of warnings that were generated when the test was last executed.	Number	A high value of this measure indicates problems with Hyper-V VMMS that may not have an immediate impact, but may cause future problems. Please check the *Hyper-V VMMS Logs* in the Event Log Viewer for more details.
Critical messages	Indicates the number of critical events that were generated when the test was last executed.	Number	A critical event is one that the Hyper- V VMMS cannot automatically recover from. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. A very low value (zero) indicates good health. An increasing trend or high value indicates the existence of fatal/irrepairable problems.

Measurement	Description	Measurement Unit	Interpretation
			The detailed diagnosis of this measure describes all the critical events that were generated during the last measurement period. Please check the *Hyper-V VMMS logs* in the Event Log Viewer for more details.
Verbose messages	Indicates the number of verbose events that were generated when the test was last executed.	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues better. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the *Hyper-V VMMS logs* in the Event Log Viewer for more details.

4.6.6 Hyper-V Worker Admin Log Test

The *Hyper-V Worker logs* file events related to the worker processes used for the actual running of the virtual machines. Using the **Hyper-V Worker Admin Log** test, you can be alerted if any error/warning event is captured by the *Hyper-V Worker logs* and can view the complete details of these events without accessing the event logs for it.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the **FILTER** configured

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 –** Refers to the port used by the EventLog Service. Here it is null.
4. **LOGTYPE** – Refers to the type of event logs to be monitored. The default value is Microsoft-Windows-Hyper-V-Worker-Admin.
5. **POLICY BASED FILTER** - Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options:

   - Manually specify the event sources, IDs, and descriptions in the **FILTER** text area, or,
   - Select a specification from the predefined filter policies listed in the **FILTER** box

   For explicit, manual specification of the filter conditions, select the **NO** option against the **POLICY BASED FILTER** field. This is the default selection. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **YES** option against the **POLICY BASED FILTER** field.

6. **FILTER** - If the **POLICY BASED FILTER** flag is set to **NO**, then a **FILTER** text area will appear, wherein you will have to specify the event sources, event IDs, and event descriptions to be monitored. This specification should be of the following format: *{Displayname}:{event_sources_to_be_included}:{event_ sources_ to_ be_ excluded}: {event_ IDs_ to_ be_ included}: {event_ IDs_ to_ be_ excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}*. For example, assume that the **FILTER** text area takes the value, *OS_ events:all:Browse,Print:all:none:all:none*. Here:

   - *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI;
   - *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the event sources are monitored, specify *none*.
   - In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The *all* in our example represents that all the event IDs need to be considered while monitoring.
   - Similarly, the *none* (following *all* in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs

as a comma-separated list. Likewise, specifying *all* makes sure that all the event IDs are excluded from monitoring.

- The all which follows implies that all events, regardless of description, need to be included for monitoring. To exclude all events, use none. On the other hand, if you provide a comma-separated list of event descriptions, then the events with the specified descriptions will alone be monitored. Event descriptions can be of any of the following forms - desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
- In the same way, you can also provide a comma-separated list of event descriptions to be excluded from monitoring. Here again, the specification can be of any of the following forms: desc*, or desc, or *desc*,or desc*, or desc1*desc2, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. In our example however, none is specified, indicating that no event descriptions are to be excluded from monitoring. If you use all instead, it would mean that all event descriptions are to be excluded from monitoring.

By default, the **FILTER** parameter contains the value: *all:all:none:all:none:all:none*. Multiple filters are to be separated by semi-colons (;).

Note:

The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window.

On the other hand, if the **POLICY BASED FILTER** flag is set to **YES**, then a **FILTER** list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is built into the policy in the following format:

{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}

To monitor a specific combination of event sources, event IDs, and event descriptions, you can choose the corresponding filter policy from the **FILTER** list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link appears just above the test configuration section, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one.

The changed policy or the new policy can then be associated with the test by selecting the policy name from the **FILTER** list box in this page.

7. **USEWMI** - The eG agent can either use WMI to extract event log statistics or directly parse the event logs using event log APIs. If the **USEWMI** flag is **YES**, then WMI is used. If not, the event log APIs are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the WinMgmt process to shoot up. On such systems, set the **USEWMI** parameter value to **NO**. **On the other hand, when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the USEWMI flag should always be set to 'Yes'.**

8. **STATELESS ALERTS -** Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the EventLog test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the **STATELESS ALERTS** flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.

9. **EVENTS DURING RESTART -** By default, the **EVENTS DURING RESTART** flag is set to **Yes**. This ensures that whenever the agent is stopped and later started, the events that might have occurred during the period of non-availability of the agent are included in the number of events reported by the agent. Setting the flag to **No** ensures that the agent, when restarted, ignores the events that occurred during the time it was not available.

10. **DDFORINFORMATION** – eG Enterprise also provides you with options to restrict the amount of storage required for event log tests. Towards this end, the **DDFORINFORMATION** and **DDFORWARNING** flags have been made available in this page. By default, both these flags are set to **YES**, indicating that by default, the test generates detailed diagnostic measures for information events and warning events. If you do not want the test to generate and store detailed measures for information events, set the **DDFORINFORMATION** flag to **No**.

11. **DDFORWARNING** – To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **NO**.

12. **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**.

13. **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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Error messages	This refers to the number of error events that were generated.	Number	A very low value (zero) indicates that no problems exist in the worker processes. An increasing trend or high value indicates the existence of problems. Please check the *Hyper-V Worker Logs* in the Event Log Viewer for more details.
Information messages	This refers to the number of information events generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed. Please check the *Hyper-V Worker*

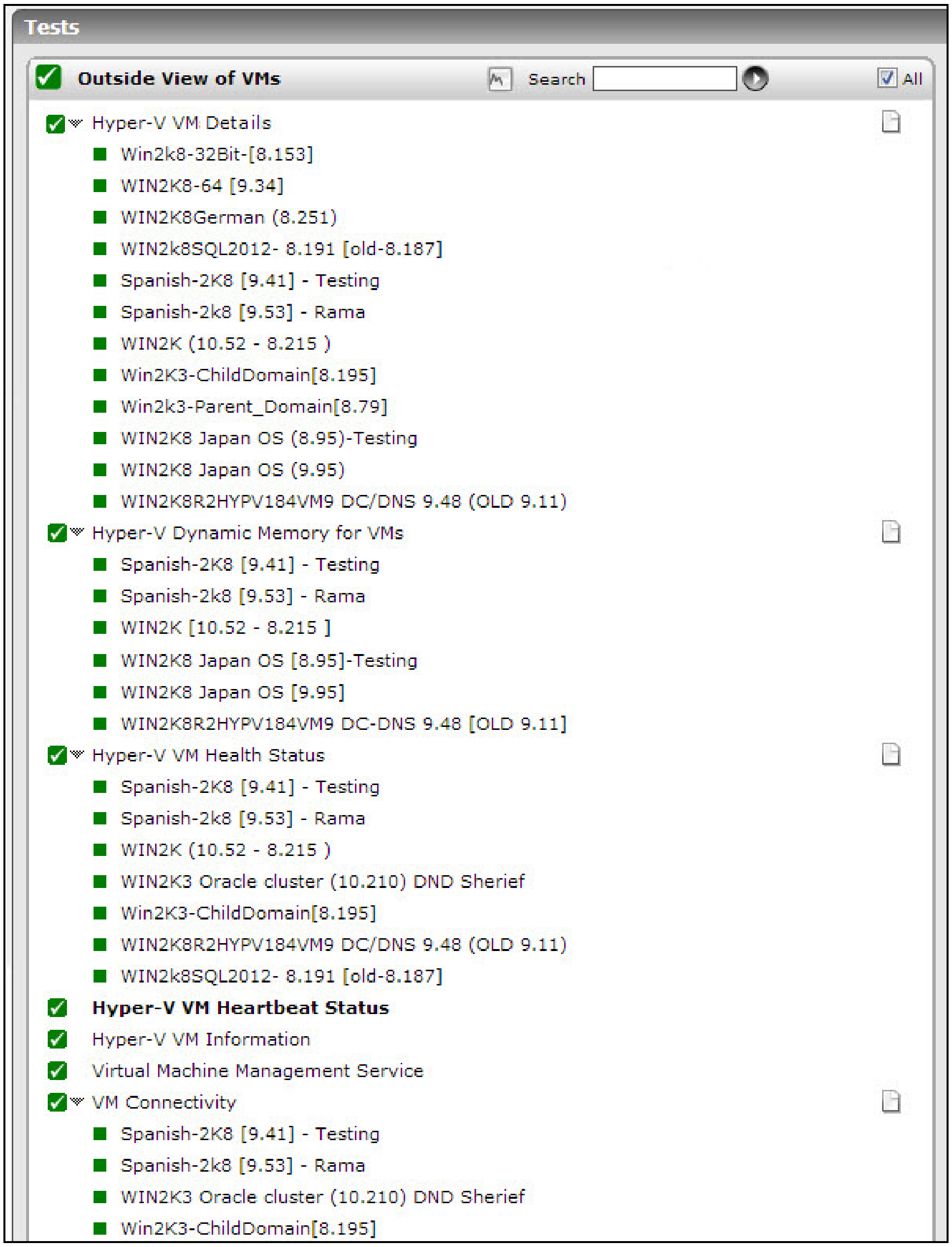
Measurement	Description	Measurement Unit	Interpretation
			Logs in the Event Log Viewer for more details.
Warnings	This refers to the number of warnings that were generated when the test was last executed.	Number	A high value of this measure indicates problems with worker processes that may not have an immediate impact, but may cause future problems. Please check the *Hyper-V Worker Logs* in the Event Log Viewer for more details.
Critical messages	Indicates the number of critical events that were generated when the test was last executed.	Number	A critical event is one that the worker processes cannot automatically recover from. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. A very low value (zero) indicates good health. An increasing trend or high value indicates the existence of fatal/irrepairable problems. The detailed diagnosis of this measure describes all the critical events that were generated during the last measurement period. Please check the *Hyper-V Worker logs* in the Event Log Viewer for more details.
Verbose messages	Indicates the number of verbose events that were generated when	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues

Measurement	Description	Measurement Unit	Interpretation
	the test was last executed.		better. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the *Hyper-V Worker logs* in the Event Log Viewer for more details.

4.7 Outside View of VMs Layer

To be able to accurately assess the physical resource usage on the Hyper-V server, and to precisely identify the root-cause for any sudden/consistent resource drains on the server, the knowledge of the resource utilization of the host operating system and the root partition alone might not suffice. Administrators also need to know how each VM on the server uses the available physical resources, so that resource-intensive VMs can be promptly isolated.

Using the tests associated with the **Outside View of VMs** layer reports the powered-on status of every VM and also reveals the relative resource usage of the VMs, thereby pointing administrators to the source of a physical resource contention on the server – is it the host operating system or is it one/more of the VMs?



Tests

☑ Outside View of VMs Search ☑ All

- ☑ Hyper-V VM Details
  - 🟩 Win2k8-32Bit-[8.153]
  - 🟩 WIN2K8-64 [9.34]
  - 🟩 WIN2K8German (8.251)
  - 🟩 WIN2k8SQL2012- 8.191 [old-8.187]
  - 🟩 Spanish-2K8 [9.41] - Testing
  - 🟩 Spanish-2k8 [9.53] - Rama
  - 🟩 WIN2K (10.52 - 8.215)
  - 🟩 Win2K3-ChildDomain[8.195]
  - 🟩 Win2k3-Parent_Domain[8.79]
  - 🟩 WIN2K8 Japan OS (8.95)-Testing
  - 🟩 WIN2K8 Japan OS (9.95)
  - 🟩 WIN2K8R2HYPV184VM9 DC/DNS 9.48 (OLD 9.11)
- ☑ Hyper-V Dynamic Memory for VMs
  - 🟩 Spanish-2K8 [9.41] - Testing
  - 🟩 Spanish-2k8 [9.53] - Rama
  - 🟩 WIN2K [10.52 - 8.215]
  - 🟩 WIN2K8 Japan OS [8.95]-Testing
  - 🟩 WIN2K8 Japan OS [9.95]
  - 🟩 WIN2K8R2HYPV184VM9 DC-DNS 9.48 [OLD 9.11]
- ☑ Hyper-V VM Health Status
  - 🟩 Spanish-2K8 [9.41] - Testing
  - 🟩 Spanish-2k8 [9.53] - Rama
  - 🟩 WIN2K (10.52 - 8.215)
  - 🟩 WIN2K3 Oracle cluster (10.210) DND Sherief
  - 🟩 Win2K3-ChildDomain[8.195]
  - 🟩 WIN2K8R2HYPV184VM9 DC/DNS 9.48 (OLD 9.11)
  - 🟩 WIN2k8SQL2012- 8.191 [old-8.187]
- ☑ **Hyper-V VM Heartbeat Status**
- ☑ Hyper-V VM Information
- ☑ Virtual Machine Management Service
- ☑ VM Connectivity
  - 🟩 Spanish-2K8 [9.41] - Testing
  - 🟩 Spanish-2k8 [9.53] - Rama
  - 🟩 WIN2K3 Oracle cluster (10.210) DND Sherief
  - 🟩 Win2K3-ChildDomain[8.195]

Figure 4.9: The tests linked to the Outside View of VMs layer

4.7.1 Hyper-V VM Details Test

This test monitors the amount of the physical server's resources that each guest on a Hyper-V 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.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for every guest operating on the monitored Hyper-V server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 4.7.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)':** On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*
7. **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.

8. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.
9. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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. **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.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Is VM powered on? :</td><td>Whether the virtual machine is Hyper-V server host or not.</td><td></td><td>While the test reports a wide variety of other metrics too for virtual machines that are alive, only the <i>Powered on status</i> is indicated for virtual machines that are currently not available.
If this measure reports the value <i>On</i>, it indicates that the guest is up and running. The value Off could indicate that the guest has been powered-off; it could also indicate that the guest has moved to a different Hyper-V server.
The numeric values that correspond to each of the powered-on states discussed above are listed in the table below:
<table><tr><th>State</th><th>Value</th></tr><tr><td>On</td><td>1</td></tr><tr><td>Off</td><td>0</td></tr></table>
<b>Note:</b>
By default, this measure reports the values On or Off to indicate the status of a VM. The graph of this measure however, represents the status of a VM using the numeric equivalents - 0 or 1.</td></tr>
<tr><td>Virtual CPU allocated to VM</td><td>Indicates the number of processors currently present in this VM.</td><td>Number</td><td>All execution in the root and child partitions (where guest VMs run) happens on Virtual Processors (VPs). At a minimum, you will see one VP for each Logical Processor (LP). These account for the root VPs. You will then see one for each VP you have configured to a guest. Therefore, if you have an 8LP system with 1 guest</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			running with 2 VPs, the count here will be 10.
Virtual CPU Utilization of VM	Indicates the percentage of time spent by the virtual processor assigned to this VM in guest and hypervisor code.	Percent	This measure serves as an effective indicator of how resource-intensive a particular VM is on a specific Hyper-V server.
Virtual machine runtime	Indicates the percentage of time spent by the virtual processor in guest code.	Percent	Comparing the value of the Virtual machine runtime and Hypervisor runtime measures for every VM will reveal where the virtual processors of the VM have spent more time – in processing guest code or in processing hypervisor code?
Hypervisor runtime	Indicates the percentage of time the virtual processor spend in hypervisor code.	Percent	
Memory allocated to VM	Indicates the amount of physical memory currently allocated to this VM.	MB	
Data transmitted by VM	Indicates the number of bytes per second sent over the network adapters supported by this VM.	Mbps	
Data received by VM	Indicates the number of bytes per second sent over the network adapters supported by this VM.	Mbps	
Data dropped by VM	Indicates the number of bytes dropped on the network adapter since the last measurement period.	MB	Ideally, this value should be very low. A high value could be indicative of a network bottleneck.
Disk reads by VM	Indicates the number of bytes read per second from the disks attached to the IDE controller.	MB/Sec	These measures are good indicators of the activity on the disks attached to the IDE controller.
Disk writes by VM	Indicates the the number of bytes written per second to the disks	MB/Sec	

Measurement	Description	Measurement Unit	Interpretation
	attached to the IDE controller.		
Deposited pages	Indicates the number of memory pages currently deposited into the partition.	Number	For each partition, the hypervisor maintains a memory pool of RAM SPA pages. This pool acts just like a checking account. The amount of pages in the pool is called the balance deposited or withdrawn from the pool. When a hypercall that requires memory is made by a partition, the hypervisor withdraws the required memory from the total pool balance of that partition. When the balance available in the pool is less, then more memory pages are deposited in the pool. A very high value of this measure therefore, indicates that the balance in the pool maintained for this partition is dwindling. This is a cause for concern.
Hypercall	Indicates the rate of hypercalls made by this guest's code on the virtual processor.	Hypercalls/Sec	Hypercalls are one form of enlightenment. Guest OS's use the enlightenments to more efficiently use the system via the hypervisor. TLB flush is an example hypercall. If this value is zero, it is an indication that Integration Components are not installed. New OS's like WS08 can use hypercalls without enlightened drivers. So, hypercalls are only a prerequisite and not a guarantee for not having Integration Components installed.
Control register accesses	Indicates the rate of control register accesses by this guest on its virtual processors.	Accesses/Sec	Control registers are used to set up address mapping, privilege mode, etc.

Measurement	Description	Measurement Unit	Interpretation
HLT instructions	Indicates the rate of HLT instructions executed by t his guest on its virtual processors.	Instructions/Sec	A HLT will cause the hypervisor scheduler to de-schedule the current VP and move to the next VP in the runlist.
Emulated instruction	Indicates the rate of emulated instructions while executing guest code on the virtual processor.	Instructions/Sec	
MWAIT instructions	Indicates the rate of MWAIT instructions executed by this guest on its virtual processors.	Instructions/Sec	The MWAIT (monitored wait) instruction instructs the processor to enter a wait state in which the processor is instructed to monitor the address range between a and b and wait for an event or a store to that address range.
CPUID instructions	Indicates the rate of CPUID instructions executed by this guest on its virtual processors.	Instructions/Sec	The CPUID instruction is used to retrieve information on the local CPU's capabilities. Typically, CPUID is only called when the OS / Application first start. Therefore, this value is likely to be 0 most of the time.
Page fault intercepts	Indicates the rate of page fault exceptions intercepted by the hypervisor while executing this guest's code on the virtual processor	Intercepts/Sec	Whenever guest code accesses a page not in the CPU TLB a page fault will occur. This counter is closely correlated with the Large Page TLB Fills measure.
Total intercepts	Indicates the rate of hypervisor intercept messages.	Intercepts/Sec	Whenever a guest VP needs to exit its current mode of running for servicing in the hypervisor, this is called an intercept. Some common causes of intercepts are resolving Guest Physical Address (GPA) to Server Physical Address (SPA) translations, privileged instructions like hlt / cupid / in / out, and the end of the VP's

Measurement	Description	Measurement Unit	Interpretation
			scheduled time slice.
Large page TLB fills	Indicates the rate of virtual TLB fills on large pages.	Fills/Sec	There are two types of TLB entries (and some three). Small TLB which generally means a 4K page and large Page which generally means 2MB. There are fewer Large TLB entries on the order of 8 – 32. A non-zero value for this measures indicates that the root partition is using large pages.
Small page TLB fills	Indicates the rate of virtual TLB fills on 4K pages.	Fills/Sec	There are two types of TLB entries (and some three). Small TLB which generally means a 4K page and large Page which generally means 2MB. There are fewer Large TLB entries on the order of 64 – 1024+.
Cpu utilization of VM	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.
Disk capacity of VM	Indicates the total disk capacity of the VM.	MB	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 Hyper-V host, but very few are actively used. A VM, whether powered on or off, consumes disk space on a host. When the Hyper-V 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

Measurement	Description	Measurement Unit	Interpretation
			the VM that is taking up maximum disk space.
Disk reads and writes by VM	Indicates the rate at which read-write requests were processed by this VM.	Kbytes/Sec	Compare the value of this measure across VMs to know on which VM I/O activity was abnormally high.
Data sent and received by VM	Indicates the rate at which network I/O is processed by this VM.	Mbps	Compare the value of this measure across VMs to know on which VM network I/O activity was abnormally high.
VM health	Indicates the current state of this VM.	Number	If the value reported by this measure is 1, the status is *Ok*. If the value reported by this measure is 3, the status is *Critical*. The Detailed Diagnosis (DD) of this measure shows the VM State, Process ID, and Operational Status.
CPU ready time	Indicates the time duration during which this VM was ready to run (i.e. it had requests to dispatch to the logical processor) but was not able to because of processor contention.	Milliseconds	The values of these measures should typically be low. The more time a VM spends waiting to run, the more lag time there is in responsiveness within the VM.
CPU ready	Indicates the percentage of time during which this VM was ready to run (i.e. it had requests to dispatch to the logical processor) but was not able to because of processor contention.	Percent	

4.7.2 Hyper-V VM Information Test

Hyper-V™ live migration is designed to move running VMs with no impact on VM availability to users. By pre-copying the memory of the migrating VM to the destination physical host, live migration minimizes the amount of transfer time of the VM A live migration is deterministic, meaning that the administrator, or script, that initiates the live migration can control which computer is the destination for the live migration. The guest operating system in the migrating VM is unaware that the migration is happening, so no special configuration for the guest operating system is needed.

Below is a summary of the live migration process:

- All VM memory pages are transferred from the source Hyper-V™ physical host to the destination Hyper-V™ physical host. While this is occurring, any VM modifications to its memory pages are tracked.
- ™Pages that were modified while step 1 was occurring are transferred to the destination physical computer.
- The storage handle for the VM's VHD files are moved to the destination physical computer.
- The destination VM is brought online on the destination Hyper-V™ server.

This test reports the number of guests registered with the server, and promptly alerts administrators to addition/removal of guests from the server.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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 **HOST** listens. By default, this is NULL.
4. **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, 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 2.2.2 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 4.7.2 of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.

7. **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.

8. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.

9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *Microsoft Hyper-V* 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 a Hyper-V 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**.

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:

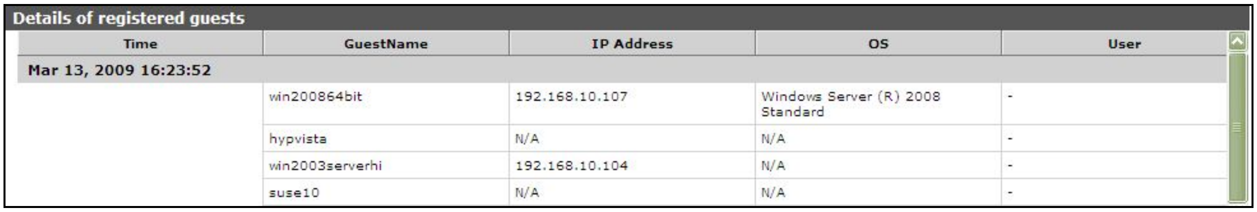
    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Registered VMs	Indicates the total number of virtual machines that have been registered with the server currently.	Number	

Measurement	Description	Measurement Unit	Interpretation
VMs 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).
VMs with users	Indicates the number of powered on guests with users logged in currently.	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 the ‘ Microsoft Hyper-V’ server model.
VMs without users	Indicates the number of powered on guests without any users logged in currently.	Number	Note that this measure will not be available for the ‘Microsoft Hyper-V’ server model.
Added VMs	Indicates the number of guests that were newly added to the server during this measurement period.	Number	The detailed diagnosis of these measures, if enabled, lists the virtual machines that were migrated to or from (as the case may be) the Hyper-V server.
Removed VMs	Indicates the number of guests that were newly removed from the server during this measurement period.	Number	

The detailed diagnosis of the *Registered VMs* measure reports the name of the guests registered with the Hype-V server, the IP address of the guests, the guest OS, and the name of the user currently logged into the guest.

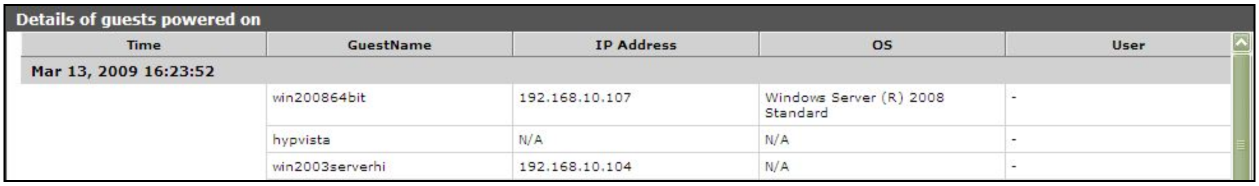


Details of registered guests

Time	GuestName	IP Address	OS	User
Mar 13, 2009 16:23:52				
	win200864bit	192.168.10.107	Windows Server (R) 2008 Standard	-
	hypvista	N/A	N/A	-
	win2003serverhi	192.168.10.104	N/A	-
	suse10	N/A	N/A	-

Figure 4.10: The detailed diagnosis of the Registered guests measure

The detailed diagnosis of the *VMs powered on* measure reports the name of the guests currently powered on, the IP address of the guests, the guest OS, and the name of the user currently logged into the guest.



Details of guests powered on

Time	GuestName	IP Address	OS	User
Mar 13, 2009 16:23:52				
	win200864bit	192.168.10.107	Windows Server (R) 2008 Standard	-
	hypvista	N/A	N/A	-
	win2003serverhi	192.168.10.104	N/A	-

Figure 4.11: The detailed diagnosis of the Guests powered on measure

Note:

The eG agent can extract the name and “outside view” metrics of Linux guests, but can neither discover the IP address nor report “inside view” metrics pertaining to Linux guests. Similarly, the eG agent cannot discover the IP address or obtain the “inside view” of those Windows VMs which do not support **Key/Value Pair Exchange** script

4.7.3 Virtual Machine Management Service Test

The Virtual Machine Management Service (VMMS) is responsible for managing the state of all virtual machines in child partitions. By periodically monitoring the VMMS, you can exercise better control over the operations of the VMs. This test monitors the VMMS and reports the number of VMs in various states.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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.

Measurements reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Deleting	Indicates the number of	Number	

Measurement	Description	Measurement Unit	Interpretation
	virtual machines that are currently being deleted.		
Exporting	Indicates the number of VMs that are currently exporting	Number	
Merging disks	Indicates the number of virtual machines that are merging disks currently.	Number	
Paused	Indicates the number of virtual machines that have been paused currently.	Number	
Running	Indicates the number of virtual machines that are currently running.	Number	
Turned off	Indicates the number of virtual machines that are currently turned off.	Number	

4.7.4 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 the VM Connectivity test, and reports whether the VM is accessible over the network or not.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each VM configured on the Hyper- V 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. **PACKETSIZE -** The size of packets used for the test (in bytes)
5. **PACKETCOUNT –** The number of packets to be transmitted during the test
6. **TIMEOUT -** How long after transmission should a packet be deemed lost (in seconds)
7. **PACKETINTERVAL** - Represents the interval (in milliseconds) between successive packet transmissions during the execution of the network test for a specific target.
8. **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 of VM* 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 reported 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 packets lost during transmission	Percent	Packet loss is often caused by network buffer overflows at a network router or by packet

Measurement	Description	Measurement Unit	Interpretation
	from source to target and back.		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.

4.7.5 Hyper-V Dynamic Memory for VMs

Dynamic Memory is a new Hyper-V feature that helps you to use physical memory more efficiently. With Dynamic Memory, Hyper-V treats memory as a shared resource that can be reallocated automatically among running virtual machines. Dynamic Memory adjusts the amount of memory available to a virtual machine, based on changes in memory demand and values that you specify.

Using this test, you can determine whether or not the Dynamic Memory feature is enabled on a virtual machine, and if so, assess how well that feature works. In the process, you can also ascertain the following:

- Isolate resource-hungry VMs;
- Understand the Dynamic Memory configuration of each VM;
- Figure out whether this configuration needs to be fine-tuned to facilitate more efficient and effective resource-sharing among VMs.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the each VM on the Hyper-V 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.

Measurements reported by the test

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Is dynamic memory enabled?</td><td>Indicates whether the Dynamic Memory feature is enabled or not in this VM.</td><td></td><td>There is no global setting to turn Dynamic Memory on or off at the host level. It must be configured for each virtual machine. By default, a virtual machine is set up with the traditional static amount of memory. You can edit the properties of a virtual machine to enable Dynamic Memory.
This measure reports the value <i>Yes</i> if Dynamic Memory is enabled on a VM, and the value No if it is not.
The table below lists the numeric values that correspond to the <i>Yes/No</i> values reported by the measure:<table><tr><th>State</th><th>Value</th></tr><tr><td>Yes</td><td>1</td></tr><tr><td>No</td><td>0</td></tr></table><b>Note:</b>
By default, this measure</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			reports the values *Yes* or *No* to indicate whether dynamic memory is enabled or not for a measure. The graph of this measure however, represents the dynamic memory status using the numeric equivaluents - 0 or 1 - only.
Added memory	Indicates the amount of physical memory added to this VM.	MB	
Removed memory	Indicates the amount of physical memory removed from this VM.	MB	
Physical memory	Indicates the amount of physical memory allocated to this VM.	MB	
Guest visible physical memory	Indicates the amount of physical memory actually utilized by this VM as seen from within the VM.	MB	
Average pressure	Indicates the average memory pressure on this VM.	Percent	Dynamic Memory determines the amount of memory needed by a virtual machine by calculating something called memory pressure. To perform this calculation, Hyper-V looks at the total committed memory of the guest operating system running in the virtual machine and then calculates pressure as the ratio of how much memory the virtual machine wants to how much it has. A very high value of this

Measurement	Description	Measurement Unit	Interpretation
			measure therefore indicates that the VM is resource-hungry.
Current pressure	Indicates the current memory pressure in this VM.	Percent	
Maximum pressure	Indicates the maximum pressure band in this VM.	Percent	
Minimum pressure	Indicates the minimum pressure band in this VM.	Percent	
Memory add operations	Indicates the number of memory addition operations performed on this VM since the last measurement period.	Number	A consistent rise in the value of this measure could indicate that the memory needs of the VM are growing.
Memory remove operations	Indicates the number of memory removal operations performed on this VM since the last measurement period.	Number	If Dynamic Memory sees pressure reduce within a VM, it is an indication that memory can be returned back to the pool, making it available for reassignment. To remove unneeded memory from a Dynamic Memory- enabled VM, Hyper-V uses a process called ballooning. Using a balloon, Dynamic Memory effectively blocks the memory freed up by a VM. This means that the VM cannot use the memory until the balloon (the block) is shrunk by being re- assigned that memory from the available memory on the host. Once the balloon is in place, the Dynamic Memory works with

Measurement	Description	Measurement Unit	Interpretation
			the parent partition to reassign the physical memory back to the host.
Starting memory	Indicates the total amount of RAM in the virtual system, as seen by this guest operating system. For a virtual system with dynamic memory enabled, this represents the initial memory available at startup.	MB	The value of this measure needs to be high enough to allow the guest operating system to start, but should be as low as possible to allow for optimal memory utilization and potentially higher consolidation ratios.
Maximum memory	Indicates the maximum amount of memory that may be consumed by this VM. For a virtual system with dynamic memory enabled, this represents the maximum memory setting.	MB	The value can be set from as low as the value for Startup RAM to as high as 64 GB. However, a virtual machine can use only as much memory as the maximum amount supported by the guest operating system. For example, if you specify 64 GB for a virtual machine running a guest operating system that supports a maximum of 32 GB, the virtual machine cannot use more than 32 GB.
Memory buffer	Defines the amount of extra memory that should be reserved for this virtual machine at runtime, as a percentage of the total memory that the virtual machine is thought to need.	Percent	Memory buffer is specified as a percentage because the actual amount of memory that represents the buffer changes in response to changes in memory usage while the virtual machine is running. Hyper- V uses performance counters in the

Measurement	Description	Measurement Unit	Interpretation
			virtual machine that identify committed memory to determine the current memory requirements of the virtual machine and then calculates the amount of memory to add as a buffer. The buffer is determined using the following formula: Amount of memory buffer = how much memory the virtual machine actually needs / (memory buffer value / 100). For example, if the memory committed to the guest operating system is 1000 MB and the memory buffer is 20%, Hyper-V will attempt to allocate an additional 20% (200 MB) for a total of 1200 MB of physical memory allocated to the virtual machine.
Memory weight	Defines the memory allocation weighting value for this virtual machine. After all reserves have been met, the remaining memory of the hosting platform will be allocated to virtual systems based on their relative weights (not to exceed the value specified by the Limit property). This property is inherited from CIM_	Number	This provides Hyper-V with a way to determine how to distribute memory among VMs if there is not enough physical memory available on the host to give every VM the amount of memory it requests.

Measurement	Description	Measurement Unit	Interpretation
	ResourceAllocationSettingData.		
Memory demand	Indicates the amount of memory that this VM needs to perform correctly, as per the running workload.	MB	A high value indicates that the VM is processing a memory-intensive workload.

4.7.6 Hyper-V VM Heartbeat Status Test

User access to a VM can be disrupted by many factors. A poor network link or a broken network link can delay/deny users access to a VM. Beside such external network connectivity issues, a user may not be able to reach a VM owing to internal issues as well – these issues can range from a VM lock, a VM crash, or a sudden termination of a VM's operations. This is why, when a user complains of being unable to access a VM, the administrator needs to quickly determine the reason for the inaccessibility of the VM, so that the correct remedial action can be initiated and access to the VM can be swiftly restored.

The **Hyper-V VM Heartbeat Status** test periodically monitors the hearbeat service installed on each VM and reports whether that service and the VM it is operating on are functioning properly or not. The heartbeat service allows the parent partition to detect when a virtual machine has locked up, crashed or otherwise ceased to function. The parent partition sends heartbeat messages to the guest operating system at regular intervals. It is then the job of the Hyper-V Heartbeat Service installed on the guest operating system to send a response to each of these heartbeat messages. When the parent partition fails to receive responses from the child partition, it assumes that the child's Heartbeat Service, and therefore the guest operating system on which it is running, has encountered problems. By closely monitoring the heartbeat service, this test enables administrators to determine whether/not internal issues (eg., a VM lock, a VM crash, etc.) are affecting the accessibility of a VM. If the test reports that the heartbeat service and the VM it is installed on are up and running, the administrator can safely conclude that internal factors are not responsible for the unavailability of that VM; further investigation as to the reason for the VM's unavailability can then be carried out.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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. 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:

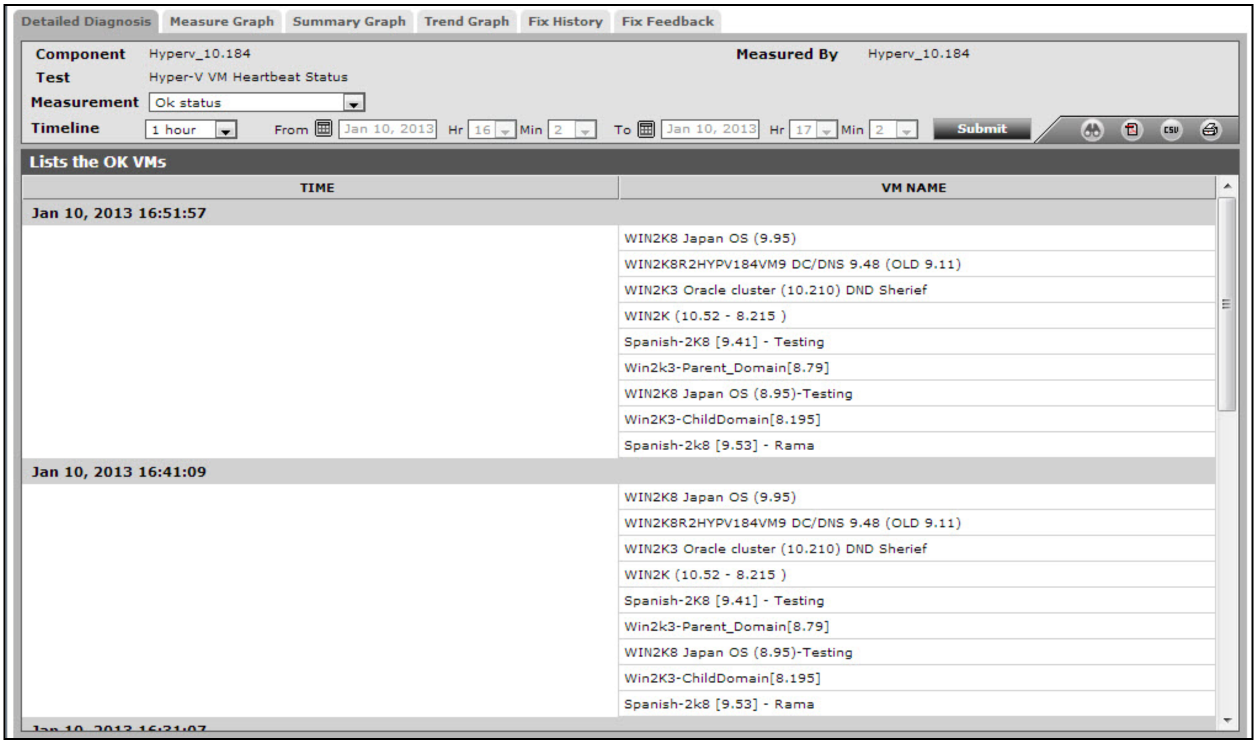
   - The eG manager license should allow the detailed diagnosis capability
   - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
OK status	Indicates the number of VMs on which the heartbeat service is operating normally.	Number	A high value is desired for this measure. A low value indicates that the parent partition is unable to communicate with the heartbeat service on many VMs; this in turn implies that many VMs are currently unreachable. If this is the case, you will have to figure out why those VMs are unavailable and initiate the required corrective action. You can use the detailed diagnosis of this measure to know which VMs are operating normally.
Error status	Indicates the number of VMs that do not support a compatible protocol version.	Number	Ideally, the value of this measure should be low. You can use the detailed diagnosis

Measurement	Description	Measurement Unit	Interpretation
			of this measure to know which VMs have encounrtered errors.
Lost contact status	Indicates the number of VMs on which the heartbeat service has not been installed yet or has not yet been contacted by the parent partition.	Number	Ideally, the value of this measure should be 0. You can use the detailed diagnosis of this measure to know on which VMs the heartbeat service has not been installed or is yet to be contacted.
Lost communication status	Indicates the number of VMs on which the hearbeat service is not responding to the hearbeat messages sent by the parent partition.	Number	Ideally, the value of this measure should be 0. A high value indicates that the heartbeat service on many VMs is not responding to heartbeat messages. This could be owing to a VM lock, a VM crash, or any other activity that can temporarily/permanently suspend VM operations. You can use the detailed diagnosis of this measure to know the VMs with which the parent partition is unable to communicate.
Unknown status	Indicates the number of VMs that have been powered off.	Number	If a VM is powered off, the parent partition will not be able to contact the heartbeat service on that VM at all. This again can cause user accesses to that VM to be denied. You can use the detailed diagnosis of this measure to know the VMs that are in an Unknown state.

The detailed diagnosis of the *OK status* measure reveals the VMs that are currently operating normally.



Component Hyperv_10.184

Test Hyper-V VM Heartbeat Status

Measured By Hyperv_10.184

Measurement Ok status

Timeline 1 hour From Jan 10, 2013 Hr 16 Min 2 To Jan 10, 2013 Hr 17 Min 2

Lists the OK VMs

TIME	VM NAME
Jan 10, 2013 16:51:57	
	WIN2K8 Japan OS (9.95)
	WIN2K8R2HYPV184VM9 DC/DNS 9.48 (OLD 9.11)
	WIN2K3 Oracle cluster (10.210) DND Sherief
	WIN2K (10.52 - 8.215)
	Spanish-2K8 [9.41] - Testing
	Win2k3-Parent_Domain[8.79]
	WIN2K8 Japan OS (8.95)-Testing
	Win2K3-ChildDomain[8.195]
	Spanish-2k8 [9.53] - Rama
Jan 10, 2013 16:41:09	
	WIN2K8 Japan OS (9.95)
	WIN2K8R2HYPV184VM9 DC/DNS 9.48 (OLD 9.11)
	WIN2K3 Oracle cluster (10.210) DND Sherief
	WIN2K (10.52 - 8.215)
	Spanish-2K8 [9.41] - Testing
	Win2k3-Parent_Domain[8.79]
	WIN2K8 Japan OS (8.95)-Testing
	Win2K3-ChildDomain[8.195]
	Spanish-2k8 [9.53] - Rama
Jan 10, 2013 16:31:07	

Figure 4.12: The detailed diagnosis of the OK status measure

4.7.7 Hyper-V VM Replications Test

Hyper-V Replica enables organizations to implement an affordable Business Continuity and Disaster Recovery (BCDR) solution for virtualized workloads. This allows virtual machines running at a primary site to be efficiently replicated to secondary location (Replica site) across a WAN link.

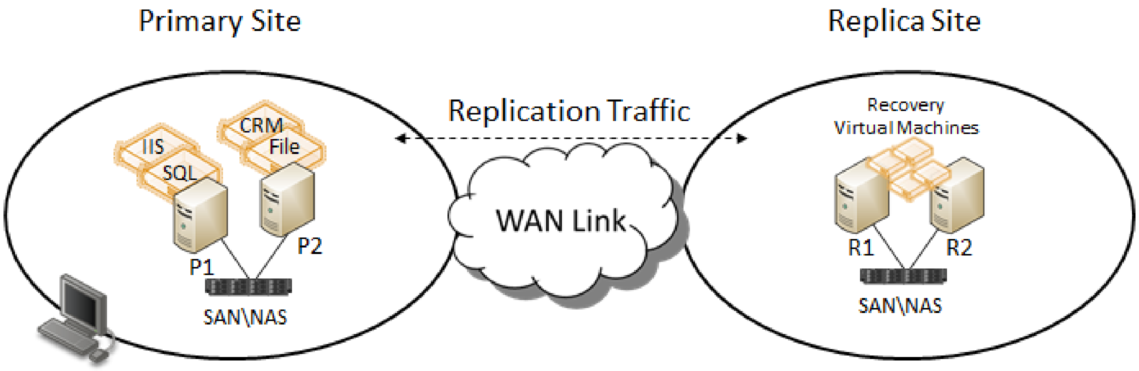


Figure 4.13: How Hyper-V Replica works

Delays/errors in the replication process can cause a severe data non-sync between the primary and secondary sites, resulting in significant loss of data when disaster strikes and recovery is attempted. To protect the data from loss, you need to monitor the replication machinery continuously, and on the slightest sign of a disturbance, alert the relevant administrators and ensure that the anomaly is promptly remediated. The **Hyper-V VM Replication** test does just that. This test monitors the replication activity performed by Hyper-V Replica for each VM on a Hyper-V host, instantly detects latencies or inconsistencies in the process, and proactively warns administrators of the same, so that the necessary corrective/control action can be taken.

Note:

This test will report metrics for Microsoft Hyper-V Server 2012 only.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each VM on a Hyper-V 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.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Average replication latency	Indicates the average time taken to replicate this VM to another host.	Secs	A low value is desired for this measure. Typically, replication is said to be 'Normal' if latency is less than 5 minutes. A high value indicates that too much time is taken for replicating a VM. This could be owing to network connectivity issues, storage issues on the primary or replica or if the primary VM

Measurement	Description	Measurement Unit	Interpretation
			requires resynchronization. This is a cause for concern, as it can cause significant data loss at the time of a fail-over. To identify the VM that is taking the maximum time to replicate, compare the value of this measure across VMs.
Average replication size	Indicates the average size of the replication files related to this VM.	MB	If large sized files are transferred over the network as part of a replication activity, it is bound to consume more bandwidth and even delay the replication process. A low value is hence desired for this measure per VM. To find out which VM's replica contains files of the maximum size, compare the value of this measure across VMs.
Compression efficiency	Indicates the percentage compression efficiency for the files that have been transferred over the network when replicating this VM.	Percent	Higher the value of this measure, greater the compression efficiency. This in turn implies optimal bandwidth usage over the network. A low value hence indicates that too much bandwidth is used when transferring replicated files over the network. Its good practice to configure Hyper- V Replica to compress the data transmitted over the network in the settings for the virtual machine in Hyper- V Manager. You can also use tools outside of Hyper- V to perform compression.
Last replication size	Indicates the size of the files replicated for this VM during the last	MB	

Measurement	Description	Measurement Unit	Interpretation
	replication.		
Network data received	Indicates the data received for this VM over the network since the virtual machine management service was started.	MB	
Network data sent	Indicates the data sent for this VM over the network since the virtual machine management service was started.	MB	
Replication count	Indicates the number of replication cycles that have run for this VM since the virtual machine management service was started.	Number	
Replication latency	Indicates the last replication latency of this VM.	Secs	It is the time taken for the delta to be applied on the recovery since it was snapped. A low value is desired for this measure.
Resynchronized data	Indicates the data sent and received over the network for this VM during the resynchronize operation since the virtual machine management service was started.	MB	A resynchronization essentially compares blocks between the Primary and Replica VHDs and then sends the delta blocks to the Replica. Scenarios where this can happen include, but may not be limited to, a failure occurred on the Primary server when changes were being made to the replication log or, if the Primary is a Failover Cluster, an unplanned cluster failover occurred.

4.7.8 Hyper-V VM Replication Health Status Test

Windows Server 2012 Hyper-V Role introduces a new capability, Hyper-V Replica, as a built-in replication mechanism at a virtual machine (VM) level. Hyper-V Replica can asynchronously replicate a selected VM running at a primary site to a designated replica site across LAN/WAN.

To track the replication status of each VM and promptly capture errors in the replication process, administrators can use the Hyper-V VM Replication Health Status test.

Note:

This test will report metrics for Microsoft Hyper-V Server 2012 only.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each VM on a Hyper-V 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. **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 reported by the test:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Replication state</td><td>Indicates the replication state of this VM.</td><td></td><td>The values that this measure can report and their corresponding numeric values are as follows:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Error</td><td>0</td></tr>
<tr><td>FailOverWaitingCompletion</td><td>1</td></tr>
<tr><td>FailedOver</td><td>2</td></tr>
<tr><td>NotApplicable</td><td>3</td></tr>
<tr><td>ReadyForInitialReplication</td><td>4</td></tr>
<tr><td>Replicating</td><td>5</td></tr>
<tr><td>Resynchronizing</td><td>6</td></tr>
<tr><td>ResynchronizeSuspended</td><td>7</td></tr>
<tr><td>Suspended</td><td>8</td></tr>
<tr><td>SyncedReplicationComplete</td><td>9</td></tr>
<tr><td>WaitingForInitialReplication</td><td>10</td></tr>
<tr><td>WaitingForStartResynchronize</td><td>11</td></tr>
</table>
Note:
By default, this measure reports the **Measure Value**s in the table above to indicate replication state. In the graph of this measure however, the same will be represented using the numeric equivalents only.
The detailed diagnosis of this measure reports the primary server name, the</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>replica server name, and the current replica server name of a VM.</td></tr>
<tr><td>Replication health</td><td>Indicates the replication health of this VM.</td><td></td><td>The values that this measure can report and their corresponding numeric values are as follows:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Normal</td><td>0</td></tr>
<tr><td>Warning</td><td>1</td></tr>
<tr><td>Critical</td><td>2</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s in the table above to indicate replication health. In the graph of this measure however, the same will be represented using the numeric equivalents only.</td></tr>
<tr><td>Replication mode</td><td>Indicates the replication mode of this VM.</td><td></td><td>The values that this measure can report and their corresponding numeric values are as follows:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Normal</td><td>0</td></tr>
<tr><td>Primary</td><td>1</td></tr>
<tr><td>Replica</td><td>2</td></tr>
<tr><td>TestReplica</td><td>3</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s in the table above to indicate replication mode. In the graph of</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			this measure however, the same will be represented using the numeric equivalents only.
Last replication time	Indicates the how much time this VM took to replicate last.	Secs	

4.7.9 Hyper-V VM Checkpoints Test

A checkpoint saves the state of each virtual hard disk that is attached to a virtual machine and all of the hard disk's contents, including application data files. For virtual machines on Hyper- V, a checkpoint also saves the hardware configuration information. By creating checkpoints for a virtual machine, you can restore the virtual machine to a previous state.

A typical use of checkpoints is to create a temporary backup before you update the operating system or an application, or make a configuration change on the virtual machine. A checkpoint allows you to restore the virtual machine to its previous state if the operation fails or adversely affects the virtual machine. For virtual machines on Hyper-V, checkpoints are also useful in a test environment where you want to use multiple hardware configurations on a virtual machine.

You can create multiple checkpoints for a virtual machine. However, checkpoints use hard disk space and, when allowed to proliferate, they can affect the performance of a virtual machine when it is running and during such virtual machine operations as migrating a virtual machine or storing it to the library.

To make sure that checkpoints do not affect VM performance, administrators need to continuously track checkpoint growth per VM, identify ‘heavy-weight’ and obsolete checkpoints that may not be of use any longer, and purge them. The **Hyper-V VM Checkpoints** test helps administrators achieve the same. This test reports the count of large and aged checkpoints per VM, and reveals the names of thesecheckpoints, so that administrators can decide whether/not these checkpoints can be removed to make more storage space available for the VM.

Note:

This test will report metrics for Microsoft Hyper-V Server 2012 only.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each VM on a Hyper-V 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 number at which the configured **HOST** listens.
4. **AGE LIMIT IN DAYS -** By default, the value of this parameter is set to 15 days. This implies that the test will report all those snapshots that are more than 15 days old as *Aged snapshots*. If required, you can change the age limit.
5. **SIZE LIMIT IN MB -** By default, the value of this parameter is set to 10000 MB. This implies that the test will report all those snapshots that have a size more than 10000 MB as *Large snapshots*. If required, you can change this limit.
6. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the admin user parameter of this test in the test configuration page. To know how to use the special page, refer to

Configuring Users for VM Monitoring of this document.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)** , then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

7. **DD FREQUENCY -** 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

8. **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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Number of checkpoints	Indicates the number of checkpoints for this VM.	Number	
Aged checkpoints count	Indicates the number of checkpoints of this VM, the age of which is more than the age limit configured for this test.	Number	Use the detailed diagnosis of this measure to identify the aged checkpoints.
Large checkpoints count	Indicates the number of checkpoints of this VM that are of a size greater	Number	Use the detailed diagnosis of this measure to identify the large-sized checkpoints.

Measurement	Description	Measurement Unit	Interpretation
	than the size limit configured for this test.		

4.8 The Inside View of VMs 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 Hyper-V 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 **Inside View of VMs** layer provide an "internal" view of the workings of each of the guests - these tests execute on an Hyper-V 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 4.14 appears, which provides you with an overview of individual guest performance (see Figure 4.14).

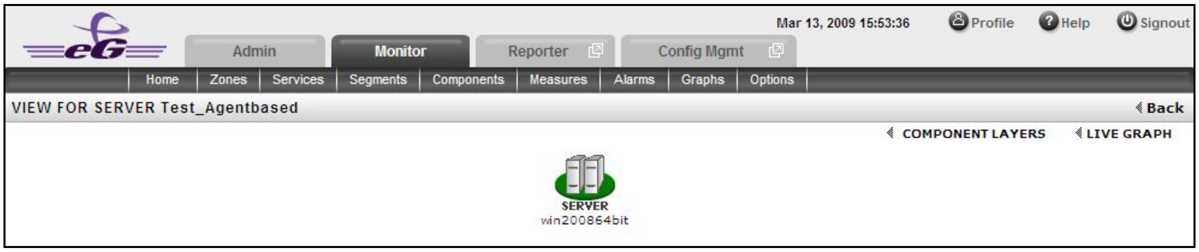
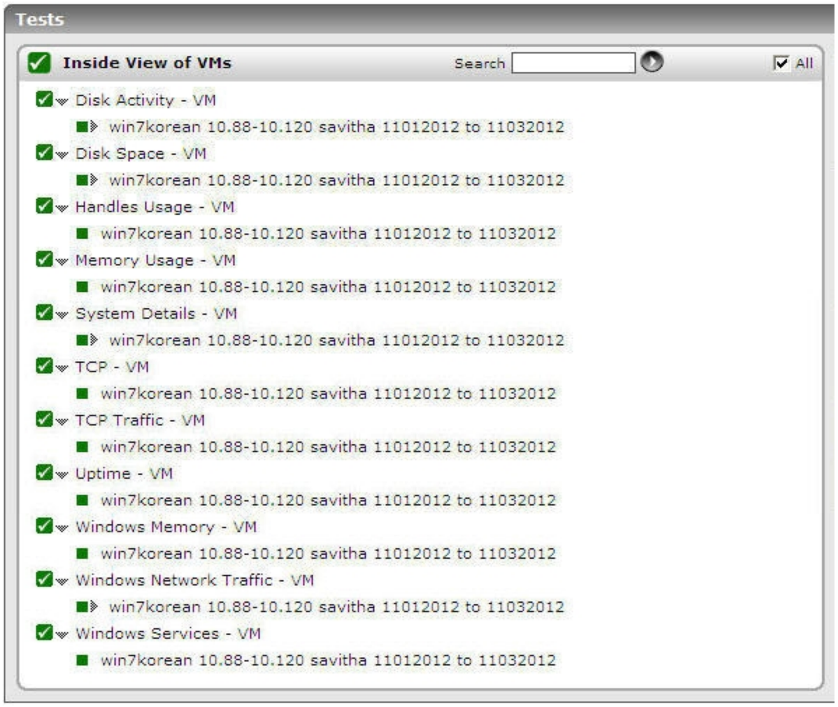


Figure 4.14: A list of guest operating systems on a Hyper-V host and their current state

To return to the layer model of the Hyper-V server and view the tests associated with the **Virtual Servers** layer, click on the **COMPONENT LAYERS** link in Figure 4.14. You can now view the list of tests mapped to the **Inside View of VMs** layer, as depicted by Figure 4.15 below.



Tests

☑ Inside View of VMs Search ☑ All

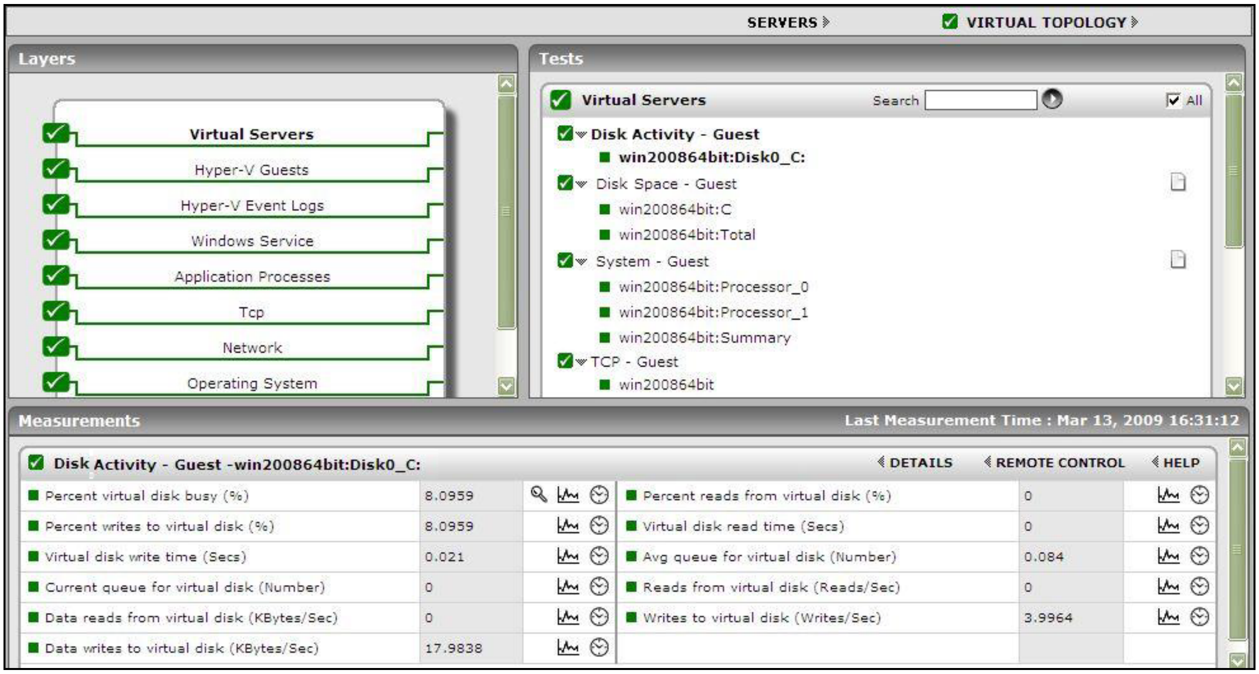
- ☑ Disk Activity - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ Disk Space - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ Handles Usage - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ Memory Usage - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ System Details - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ TCP - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ TCP Traffic - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ Uptime - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ Windows Memory - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ Windows Network Traffic - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012
- ☑ Windows Services - VM
  - win7korean 10.88-10.120 savitha 11012012 to 11032012

Figure 4.15: The tests mapped to the Inside View of VMs layer

If you want to override this default setting - i.e., if you prefer to view the tests mapped to the **Inside View of VMs** 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 **Inside View of VMs** layer is clicked, the list of tests mapped to that layer appears, as depicted by Figure 4.16.



SERVERS ▸ ☑ VIRTUAL TOPOLOGY ▸

Layers

- ☑ Virtual Servers
- ☑ Hyper-V Guests
- ☑ Hyper-V Event Logs
- ☑ Windows Service
- ☑ Application Processes
- ☑ Tcp
- ☑ Network
- ☑ Operating System

Tests

☑ Virtual Servers Search ☑ All

- ☑ Disk Activity - Guest
  - win200864bit:Disk0_C:
- ☑ Disk Space - Guest
  - win200864bit:C
  - win200864bit:Total
- ☑ System - Guest
  - win200864bit:Processor_0
  - win200864bit:Processor_1
  - win200864bit:Summary
- ☑ TCP - Guest
  - win200864bit

Measurements Last Measurement Time : Mar 13, 2009 16:31:12

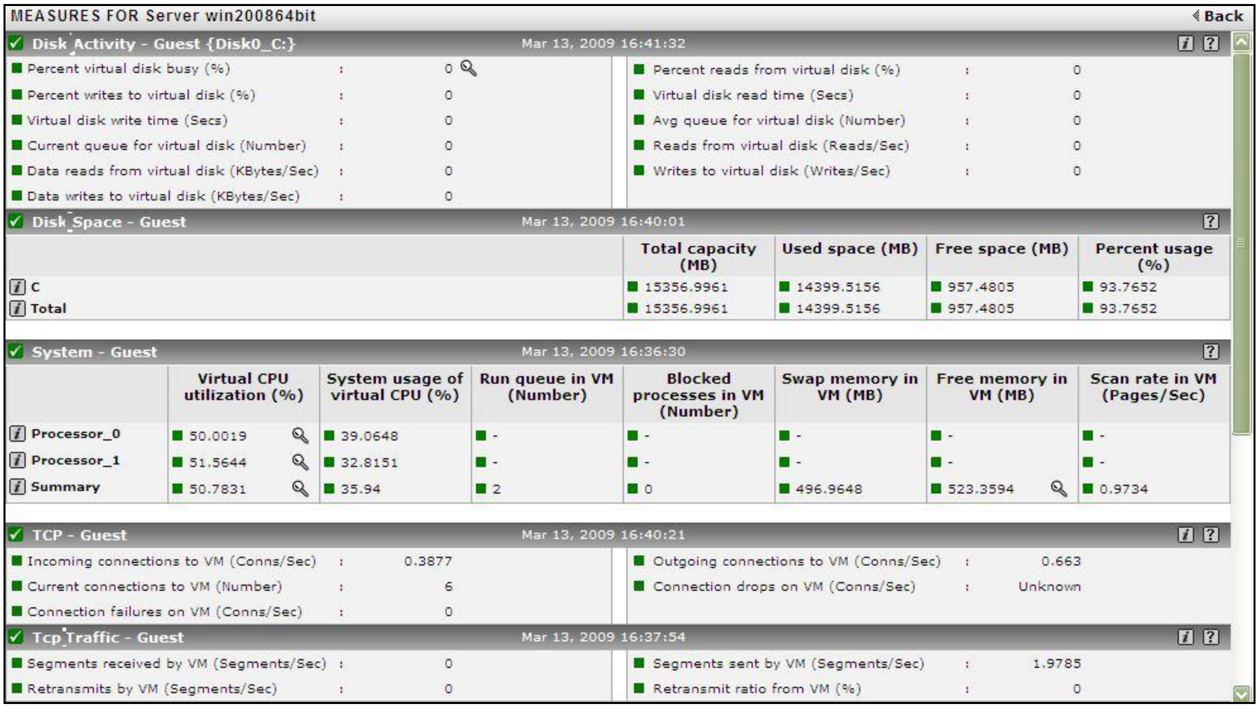
☑ Disk Activity - Guest -win200864bit:Disk0_C: DETAILS REMOTE CONTROL HELP

Percent virtual disk busy (%)	8.0959	Percent reads from virtual disk (%)	0
Percent writes to virtual disk (%)	8.0959	Virtual disk read time (Secs)	0
Virtual disk write time (Secs)	0.021	Avg queue for virtual disk (Number)	0.084
Current queue for virtual disk (Number)	0	Reads from virtual disk (Reads/Sec)	0
Data reads from virtual disk (KBytes/Sec)	0	Writes to virtual disk (Writes/Sec)	3.9964
Data writes to virtual disk (KBytes/Sec)	17.9838		

Figure 4.16: The tests mapped to the Virtual Servers layer

If you now want the **Server view** of Figure 4.14, simply click on the **SERVERS** link above the list of tests in Figure 4.16 (indicated by the arrow).

Clicking on any of the guests in the **Server view** leads you to Figure 4.17 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 4.17.



MEASURES FOR Server win200864bit ◂ Back

✓ Disk_Activity - Guest {Disk0_C:} Mar 13, 2009 16:41:32

Percent virtual disk busy (%)	:	0	Percent reads from virtual disk (%)	:	0
Percent writes to virtual disk (%)	:	0	Virtual disk read time (Secs)	:	0
Virtual disk write time (Secs)	:	0	Avg queue for virtual disk (Number)	:	0
Current queue for virtual disk (Number)	:	0	Reads from virtual disk (Reads/Sec)	:	0
Data reads from virtual disk (KBytes/Sec)	:	0	Writes to virtual disk (Writes/Sec)	:	0
Data writes to virtual disk (KBytes/Sec)	:	0			

✓ Disk_Space - Guest Mar 13, 2009 16:40:01

	Total capacity (MB)	Used space (MB)	Free space (MB)	Percent usage (%)
C	15356.9961	14399.5156	957.4805	93.7652
Total	15356.9961	14399.5156	957.4805	93.7652

✓ System - Guest Mar 13, 2009 16:36:30

	Virtual CPU utilization (%)	System usage of virtual CPU (%)	Run queue in VM (Number)	Blocked processes in VM (Number)	Swap memory in VM (MB)	Free memory in VM (MB)	Scan rate in VM (Pages/Sec)
Processor_0	50.0019	39.0648	-	-	-	-	-
Processor_1	51.5644	32.8151	-	-	-	-	-
Summary	50.7831	35.94	2	0	496.9648	523.3594	0.9734

✓ TCP - Guest Mar 13, 2009 16:40:21

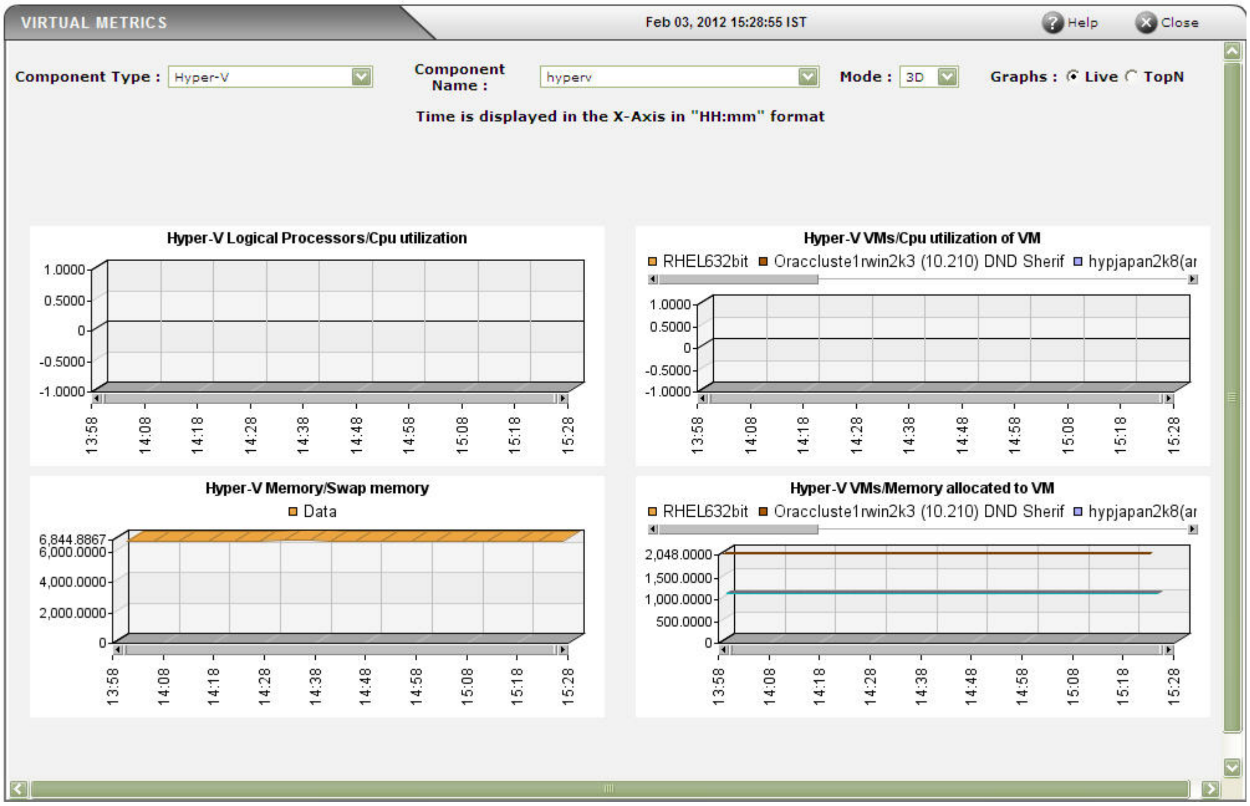
Incoming connections to VM (Conns/Sec)	:	0.3877	Outgoing connections to VM (Conns/Sec)	:	0.663
Current connections to VM (Number)	:	6	Connection drops on VM (Conns/Sec)	:	Unknown
Connection failures on VM (Conns/Sec)	:	0			

✓ Tcp_Traffic - Guest Mar 13, 2009 16:37:54

Segments received by VM (Segments/Sec)	:	0	Segments sent by VM (Segments/Sec)	:	1.9785
Retransmits by VM (Segments/Sec)	:	0	Retransmit ratio from VM (%)	:	0

Figure 4.17: Figure 2.30: Measures pertaining to a chosen guest

To view real-time graphs of pre-configured measures (pertaining to the *Hyper-V* host and the guests operating on it), click on the **LIVE GRAPH** link in Figure 4.14. Alternatively, you can click on the icon that appears in the **Tests** panel (see Figure 4.9) when the **Outside View of VMs layer** is clicked. The graph display that appears subsequently (see Figure 4.18) 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 utilization' measure of the *Hyper-V Logical Processors* test, you will find a graph of the 'Virtual machine cpu utilization' measure of the *Hyper-V Guests* test. This way, you can easily compare and correlate how well the physical CPU resources are being utilized by both the Hyper-V 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 Hyper-V host? or is it the virtual guest? If you access this page from the **LIVE GRAPH** link in Figure 4.14, then, by default, you will view live graphs pertaining to the *Hyper-V* 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 4.18.



Component Type : Hyper-V

Component Name : hyperv

Mode : 3D

Graphs : ● Live ○ TopN

Time is displayed in the X-Axis in "HH:mm" format

Figure 4.18: Figure 2.31: Live graph comparing physical resource usage of a Hyper-V server (on the left) and resource usage levels of the individual VMs (on the right)

As indicated in Figure 4.15, 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 virtual guest hosted on the Hyper-V server. Detailed diagnosis for these tests provide details of individual processes and their utilization levels.

4.8.1 Citrix VDA Status - VM Test

Citrix Virtual Delivery Agent (VDA) is installed on a virtual machine that runs the applications or virtual desktops for the user. The VDA enables connections between the applications/desktops and the users only when the connections are brokered by Citrix. The VDA enables the virtual machines to register with Delivery Controllers and manage the High Definition experience (HDX) connection to a user device. If the VDA failed to register with a delivery controller, it would not be possible for the delivery controller to broker a connection to the target virtual machine. The target virtual machine would therefore become an unusable resource. The VDA issues with respect to registration are logged in the event log of the target virtual server. Some of the most common issues that are logged into the event log are the virtual desktop not added to the correct desktop farm, the virtual desktop firewall not configured properly, DNS configuration failure, Time synchronization failure, WCF failure etc. The eG agent integrates with the XDPing to collect the metrics that details on what exactly was the reason behind the registration issues i.e., what was the service that failed. The **Citrix VDA Status - VM** test helps administrators to figure out which service has failed leading to VDA registration issues!

This test monitors the VDA installed on the target virtual machine and reports whether the services such as user authentication, active directory authentication, DNS lookup, WCF endpoints etc are successful or not. This test also reports the errors and warnings available in the event log when registration failure occurs.

Note:

This test reports metrics only if the connections to the applications and desktops are brokered via Citrix XenDesktop 7.x.

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 the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Hyper-V server

Agent deploying the test : A remote agent

Outputs of the test : One set of results for the virtual machine monitored.

Configurable parameters for the test

1. **TEST PERIOD -** How often should the test be executed. By default, this is set to 15 minutes.
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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring.
   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the

DOMAIN, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.

7. **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.

8. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.

9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *6:1*. 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**.

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
Machine account status	Indicates the current status of the account of		The values that this measure can report and its corresponding numeric

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>the machine on which the VDA was installed.</td><td></td><td>equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Failed</td><td>0</td></tr>
<tr><td>Success</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the account of the machine. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>User authentication status</td><td>Indicates the current status of the User authentication service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Failed</td><td>0</td></tr>
<tr><td>Success</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the user authentication service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>Domain controller time sync status</td><td>Indicates the current status of the Domain controller time sync service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:</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>Numeric Value</th></tr><tr><td>Failed</td><td>0</td></tr><tr><td>Success</td><td>1</td></tr></table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the Domain controller time sync service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>WCF endpoint status</td><td>Indicates the current status of the WCF endpoint service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Failed</td><td>0</td></tr><tr><td>Success</td><td>1</td></tr></table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the WCF endpoint service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>VDA windows service status</td><td>Indicates the current status of the VDA Windows service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Failed</td><td>0</td></tr><tr><td>Success</td><td>1</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><b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the VDA Windows service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>DNS lookup status</td><td>Indicates the current status of the DNS lookup service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Failed</td><td>0</td></tr>
<tr><td>Success</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the DNS lookup service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>Windows firewall status</td><td>Indicates the current status of the Windows firewall service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Failed</td><td>0</td></tr>
<tr><td>Success</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of</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>the Windows firewall service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>Registration status</td><td>Indicates the registration status of the VDA.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>AgentError</td><td>0</td></tr>
<tr><td>Unregistered</td><td>1</td></tr>
<tr><td>Registered</td><td>2</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the registration status of the VDA. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>Has valid license?</td><td>Indicates whether the VDA license is valid or not.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>No</td><td>0</td></tr>
<tr><td>Yes</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate whether the license is valid or not. In the graph of this measure however, the same is represented using the corresponding numeric equivalents</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			only.
Errors in event log in last one hour	Indicates the number of errors detected in the event log for the server during the last 1 hour.	Number	Ideally, the value of this measure should be 0.
Warnings in event log in last one hour	Indicates the number of warning messages that were logged in the event log for the server during the last 1 hour.	Number	Ideally, the value of this measure should be 0.

4.8.2 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 : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for every disk partition on a VM, in the case of a Hyper-V server and

On set of results for every disk partition used by a user who is currently logged into a virtual desktop, in the case of a Hyper-V VDI server

First-level descriptor: VM name or username_on_VM

Second-level descriptor: Disk partition

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the admin user parameter of this test in the test configuration page. To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.
   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE**

VIEW USING flag is set to **eG VM Agent (Windows)** , then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN** , **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.

7. **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.

   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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V host by default.

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 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. **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 reported by the test

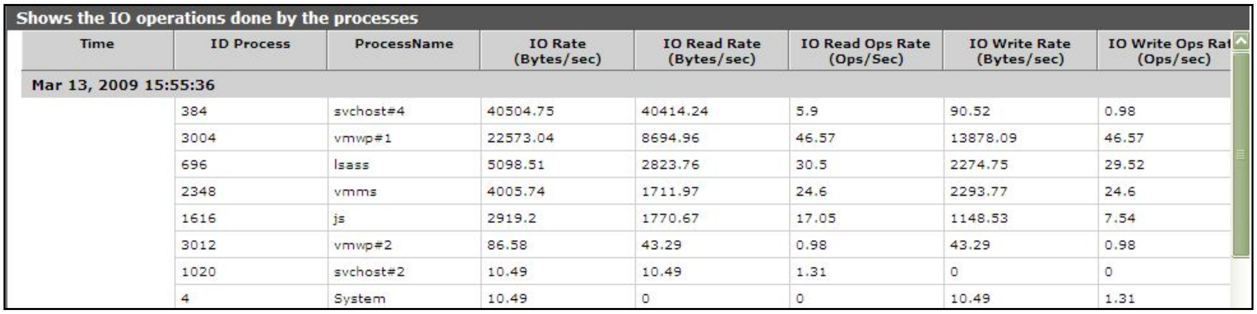
Measurement	Description	Measurement Unit	Interpretation
Percent virtual disk busy	Indicates the percentage of elapsed time during which the disk is busy processing requests (i.e., reads or writes).	Percent	Comparing the percentage of time that the different disks are busy, an administrator can determine 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	Indicates the	Percent	

Measurement	Description	Measurement Unit	Interpretation
virtual disk	percentage of elapsed time that the selected disk drive is busy servicing write requests.		
Virtual disk read time	Indicates the average time in seconds of a read of data from the disk.	Secs	
Virtual disk write time	Indicates the average time in seconds of a write of data from the disk.	Secs	
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

Measurement	Description	Measurement Unit	Interpretation
			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.
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	Secs	The value of this measure is the sum of the values of the Disk service time

Measurement	Description	Measurement Unit	Interpretation
	write operations of this disk.		and Disk queue time measures. A consistent increase in the value of this measure could indicate a latency in I/O processing.
Avg IO read size:	Indicates the average number of bytes transferred from disk during read operations.	KB	Larger I/Os tend to have higher latency (for example, BACKUP/RESTORE operations issue 1 MB transfers by default). **These measures are reported for Windows VMs only.**
Avg IO write size:	Indicates the average number of bytes transferred into disk during write operations.	KB	
Split IO:	Reports the rate at which the operating system divides I/O requests to the disk into multiple requests.	Splits/Sec	A split I/O request might occur if the program requests data in a size that is too large to fit into a single request or if the disk is fragmented. Factors that influence the size of an I/O request can include application design, the file system, or drivers. A high rate of split I/O might not, in itself, represent a problem. However, on single-disk systems, a high rate for this counter tends to indicate disk fragmentation. **This measure is reported for Windows VMs only.**

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.



Shows the IO operations done by the processes

Time	ID Process	ProcessName	IO Rate (Bytes/sec)	IO Read Rate (Bytes/sec)	IO Read Ops Rate (Ops/Sec)	IO Write Rate (Bytes/sec)	IO Write Ops Rat (Ops/sec)
Mar 13, 2009 15:55:36							
	384	svchost#4	40504.75	40414.24	5.9	90.52	0.98
	3004	vmwp#1	22573.04	8694.96	46.57	13878.09	46.57
	696	lsass	5098.51	2823.76	30.5	2274.75	29.52
	2348	vmms	4005.74	1711.97	24.6	2293.77	24.6
	1616	js	2919.2	1770.67	17.05	1148.53	7.54
	3012	vmwp#2	86.58	43.29	0.98	43.29	0.98
	1020	svchost#2	10.49	10.49	1.31	0	0
	4	System	10.49	0	0	10.49	1.31

Figure 4.19: The detailed diagnosis of the Percent virtual disk busy measure

4.8.3 Disk Space - VM Test

This test monitors the space usage of every disk partition on a guest.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for every disk partition on a VM, in the case of a Hyper-V server and

On set of results for every disk partition used by a user who is currently logged into a virtual desktop, in the case of a Hyper-V VDI server

First-level descriptor: VM name or username_on_VM

Second-level descriptor: Disk partition

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the admin user parameter of this test in the test configuration page. To know how to use the special page, refer to 4.7.1 of this document.

   - **If the INSIDE VIEW USING flag is set to ‘eG VM Agent (Windows)’**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)** , then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.

7. **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.

   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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V host by default.

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 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.

Measurements reported 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	
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	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.

Measurement	Description	Measurement Unit	Interpretation
	percentage of disk space used across all disk partitions.		

4.8.4 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 : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : For a Hyper-V server, one set of results for every processor on each VM and

For a Hyper-V VDI server, one set of results for every processor used by the user who is currently logged into each VM

First-level descriptor: VM name or User on VM

Second-level descriptor: Processor name

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **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 reported 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	Number	A value consistently greater than 2 indicates that many processes could be simultaneously contending for the processor.

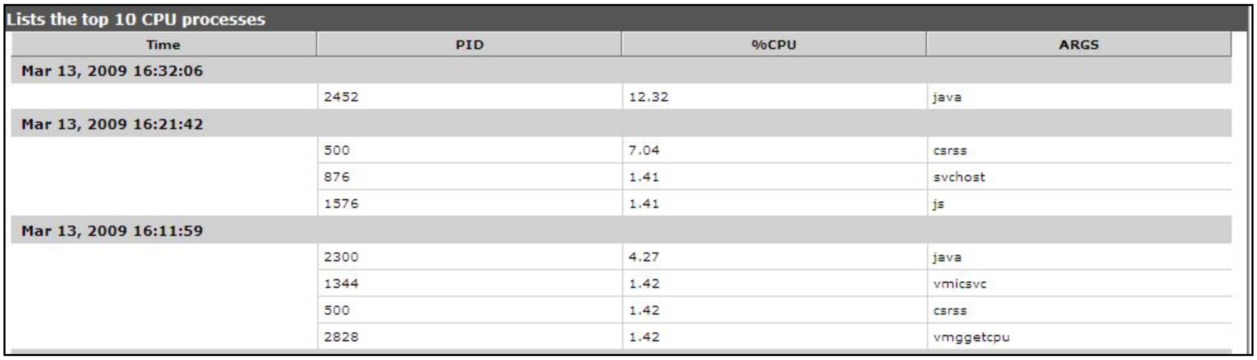
Measurement	Description	Measurement Unit	Interpretation
	the processor cycle. This length does not include the threads that are currently being executed.		
Blocked processes in VM	Indicates the number of processes blocked for I/O, paging, etc.	Number	A high value could indicate an I/O problem on the 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	This measure typically indicates the amount of memory available for use by applications running on the target VM. 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

Measurement	Description	Measurement Unit	Interpretation
			cache memory size as the value of the Free memory in VM measure while monitoring AIX and Linux guest operating systems.
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 4.20). 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.



Lists the top 10 CPU processes

Time	PID	%CPU	ARGS
Mar 13, 2009 16:32:06			
	2452	12.32	java
Mar 13, 2009 16:21:42			
	500	7.04	csrss
	876	1.41	svchost
	1576	1.41	js
Mar 13, 2009 16:11:59			
	2300	4.27	java
	1344	1.42	vmicsvc
	500	1.42	csrss
	2828	1.42	vmggetcpu

Figure 4.20: 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.

4.8.5 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.

This test included in the eG agent monitors the uptime of each VM on a Hyper-V server.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for every guest on the Hyper-V server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **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 reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Has the VM been rebooted?	Indicates whether the VM 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 VM 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 guest 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 guest was rebooted 120 secs back, this metric will report a value of 120 seconds. The accuracy of this metric is dependent on the measurement

Measurement	Description	Measurement Unit	Interpretation
			period – the smaller the measurement period, greater the accuracy.
Total uptime of the VM	Indicates the total time that the VM has been up since its last reboot.	Mins	Administrators may wish to be alerted if a guest has been running without a reboot for a very long period. Setting a threshold for this metric allows administrators to determine such conditions.

Note:

For VMs running Windows 8 (or above), the Uptime - VM test may sometimes report incorrect values. This is because of the 'Fast Startup' feature, which is enabled by default for Windows 8 (and above) operating systems. This feature ensures that the Windows operating system is NOT SHUTDOWN COMPLETELY, when the VM is shutdown. Instead, the operating system saves the image of the Windows kernel and loaded drivers to the file, C:\hiberfil.sys, upon shutdown. When the Windows VM is later started, the operating system simply loads hiberfil.sys into memory to resume operations, instead of performing a clean start. Because of this, the Windows system will not record this event as an actual 'reboot'. As a result, the Uptime - VM test will not be able to correctly report if any reboot happened recently ; neither will it be able to accurately compute the time since the last reboot.

To avoid this, you need to disable the Fast Startup feature on VMs running Windows 8 (and above). The steps to achieve this are outlined below:

1. Login to the target Windows VM.
2. Edit the Windows Registry. Look for the following registry entry:

   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Power

3. Locate the **HiberbootEnabled** key under the entry mentioned above.
4. Change the value of this key to *0* to turn off Fast Startup. By default, its value will be *1*, as Fast Startup is enabled by default.

   Also, note that the Fast Startup feature does not work if the VM is “restarted”; it works only when the VM is shutdown and then started.

4.8.6 Memory Usage - VM Test

This test reports statistics related to the usage of physical memory of the VMs.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : For a Hyper-V server, one set of results will be reported for every VM on the server

For a Hyper-V VDI server, one set of results will be reported for the user who is currently logged into each virtual desktop on the server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN** , **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.
7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **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 reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Total physical memory	Indicates the total physical memory of this VM.	MB	
Used physical memory	Indicates the used physical memory of this VM.	MB	
Free physical memory	Indicates the free physical memory of the VM.	MB	This measure typically indicates the amount of memory available for use by applications running on the target VM. 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.

Measurement	Description	Measurement Unit	Interpretation
Physical memory utilized	Indicates the percent usage of physical 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 physical memory	Indicates the amount of physical memory, immediately available for allocation to a process or for system use.	MB	Not all of the Available physical memoryisFree physical memory. Typically,Available physical memoryismade up of theStandby List, Free List, andZeroed List. When Windows wants to trim a process' working set, the trimmed 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

Measurement	Description	Measurement Unit	Interpretation
			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

Measurement	Description	Measurement Unit	Interpretation
			or for system use. 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

Measurement	Description	Measurement Unit	Interpretation
			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 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.

4.8.7 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 WindowsMemory – Guest test tracks critical metrics corresponding to the System PTEs and the pool areas of kernel memory of a Windows virtual machine.

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 *Hyper-V/Hyper-V VDI* as the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for every Windows VM guest/user on the monitored Hyper-V server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **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 target VM that are utilizing memory excessively - you can change this flag to **YES**.

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 the disk was read to resolve hard fault paging.	Reads/Sec	
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	Pages/Sec	

Measurement	Description	Measurement Unit	Interpretation
	memory that was not in its working set, meaning that the guest had to retrieve it from the page file.		
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 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

Measurement	Description	Measurement Unit	Interpretation
			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.

4.8.8 Windows Network Traffic - VM Test

This is an internal test that monitors the incoming and outgoing traffic through each Windows guest of a Hyper-V server.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : For a Hyper-V server, one set of results will be reported for every network interface supported by each Windows VM on the server.

For a Hyper-V VDI server, one set of results will be reported for every network interface used by the user who is logged into each Windows virtual desktop on the server

First-level descriptor: Windows VM or User on Windows virtual desktop

Second-level descriptor: Network interface

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **IS FULL DUPLEX** - By default, this flag is set to **Yes**, indicating that the incoming and outgoing data traffic is handled in full duplex mode. If the data traffic in your environment is handled in half-duplex mode, set this flag to **No**.

12. **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 reported by the test:

Measurement	Description	Measurement Unit	Interpretation
Incoming traffic	Indicates the rate at which data (including framing characters) is	Mbps	An abnormally high rate of incoming traffic may require additional analysis.

Measurement	Description	Measurement Unit	Interpretation
	received on a network interface.		
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.
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.

Note:

If this t test is not reporting measures for a VM, make sure that you have enabled the SNMP service for the VM.

4.8.9 TCP - VM Test

This test tracks various statistics pertaining to TCP connections to and from each guest on a Hyper-V host. The details of the test are provided below:

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : For a Hyper-V server, one set of results will be reported for every powered-on VM on the server.

For a Hyper-V VDI server, one set of results will be reported for the user who is currently logged into each virtual desktop on the server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an

administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.
7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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.

Measurements reported by the test:

Measurements	Description	Measurement Unit	Interpretation
Incoming	Indicates the	Conns/Sec	A high value can indicate an

Measurements	Description	Measurement Unit	Interpretation
connections to VM	connections per second received by the guest.		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 on VM	Indicates the rate of established TCP connections dropped from the TCP listen queue.	Conns/Sec	This value should be 0 for most of the 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.

4.8.10 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 : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : For a Hyper-V server, one set of results will be reported for every powered-on VM on the server.

For a Hyper-V VDI server, one set of results will be reported for the user who is currently logged into each virtual desktop on the server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.
   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **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.

7. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

8. **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.

9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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.

Measurements reported 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 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

Measurement	Description	Measurement Unit	Interpretation
			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.

4.8.11 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 : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : For a Hyper-V server, one set of results will be reported for every powered-on Windows VM on the server.

For a Hyper-V VDI server, one set of results will be reported for the user who is currently logged into each Windows virtual desktop on the server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.
   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are

identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **HANDLES GROWTH LIMIT** – This defines the upper limit of the handles opened by any process. By default, this parameter is set to 8000.

12. **IGNORE PROCESSES IN DD -** The detailed diagnosis of the *Processes using handles above limit in the VM* measure reveals the top-10 processes in a VM that are using handles above the configured limit, the number of handles used by each process, and the break-up of the handle count by sub-handles (i.e., the count of file handles, disk handles, etc.). For processes that typically open thousands of handles, storing granular, sub-handle-level information pertaining to these handles may impose additional strain on the eG database. In such cases, you can reduce the strain on the eG database by configuring in the **IGNORE PROCESSES IN DD** text box, a comma-separated list of process names/process patterns for which sub-handle-wise breakup need not be collected and stored in the eG database. The default value in this text box is **ccSvcHst.exe**. This implies that, by default, the detailed diagnosis of the *Processes using handles above limit in the VM* measure will only provide the total number of open handles for *ccSvcHst.exe* process, but not the sub-handle-level information. If required, you can choose to exclude the sub-handle-wise breakup from the detailed diagnosis for more processes by including these process names/patterns as part of the **IGNORE PROCESSES IN DD** specification. For instance, your specification can be: **ccSvcHst.exe*,*js.exe*,*java.exe**

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.

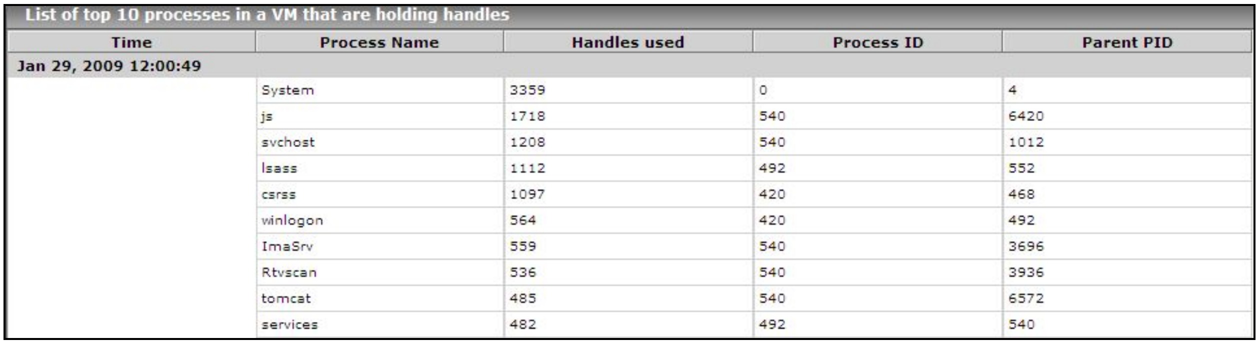
    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - 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 reported 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 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.

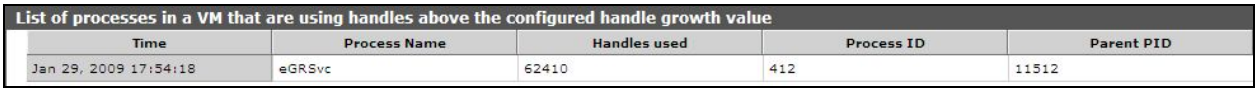


List of top 10 processes in a VM that are holding handles

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 4.21: 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 4.22: The detailed diagnosis of the Processes using handles above limit in VM measure

4.8.12 Windows Services - VM Test

This test tracks the status (whether running or have stopped) of services executing on Windows virtual machines.

This test is disabled by default for Hyper-V VDI server. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick *Hyper-V VDI* as the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : For a Hyper-V server, one set of results will be reported for every powered-on Windows VM on the server.

For a Hyper-V VDI server, one set of results will be reported for the user who is currently logged into each Windows virtual desktop on the server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **IGNORE SERVICES** – 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.

12. **IGNORE PROCESSES IN DD** - The detailed diagnosis of the *Processes using handles above limit in the VM* measure reveals the top-10 processes in a VM that are using handles above the configured limit, the number of handles used by each process, and the break-up of the handle count by sub-handles (i.e., the count of file handles, disk handles, etc.). For processes that typically open thousands of handles, storing granular, sub-handle-level information pertaining to these handles may impose additional strain on the eG database. In such cases, you can reduce the strain on the eG database by configuring in the **IGNORE PROCESSES IN DD** text box, a comma-separated list of process names/process patterns for which sub-handle-wise breakup need not be collected and stored in the eG database. The default value in this text box is **ccSvcHst.exe**. This implies that, by default, the detailed diagnosis of the *Processes using handles above limit in the VM* measure will only provide the total number of open handles for *ccSvcHst.exe* process, but not the sub-handle-level information. If required, you can choose to exclude the sub-handle-wise breakup from the detailed diagnosis for more processes by including these process names/patterns as part of the **IGNORE PROCESSES IN DD** specification. For instance, your specification can be: **ccSvcHst.exe*,*js.exe*,*java.exe**

13. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be

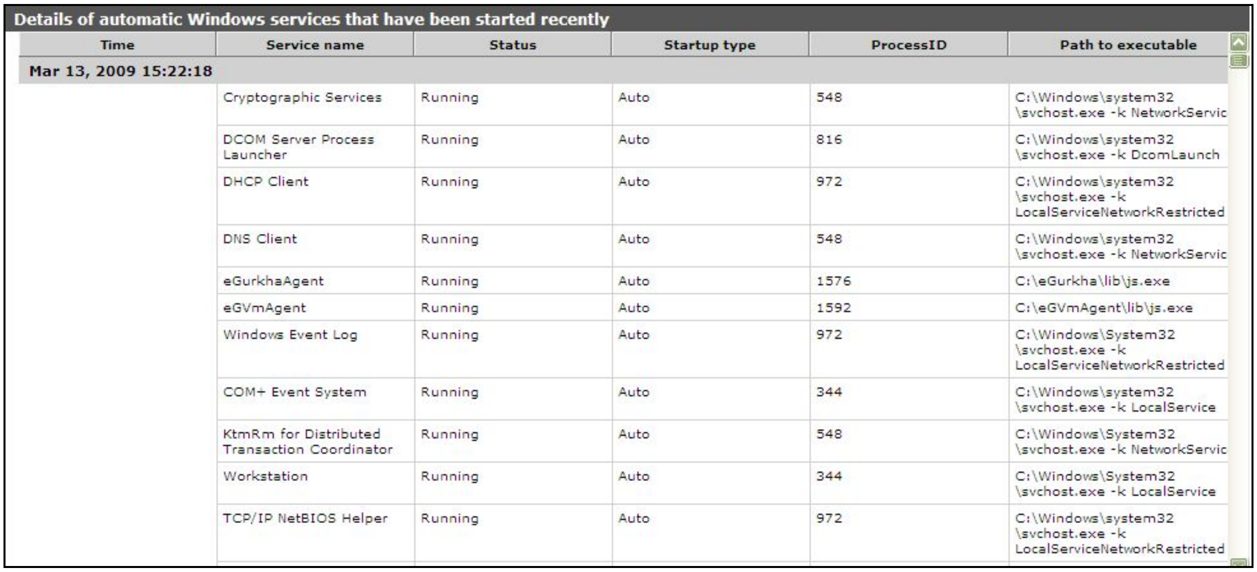
	generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
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 reported 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 automatic) 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	Number	Use the detailed diagnosis of this measure to identify the services that are running.

Measurement	Description	Measurement Unit	Interpretation
	the last measurement period.		
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.

The detailed diagnosis of the *New automatic services started* measure lists the services that were started recently, the startup type, process ID, and the complete path to the executable that controls the service.

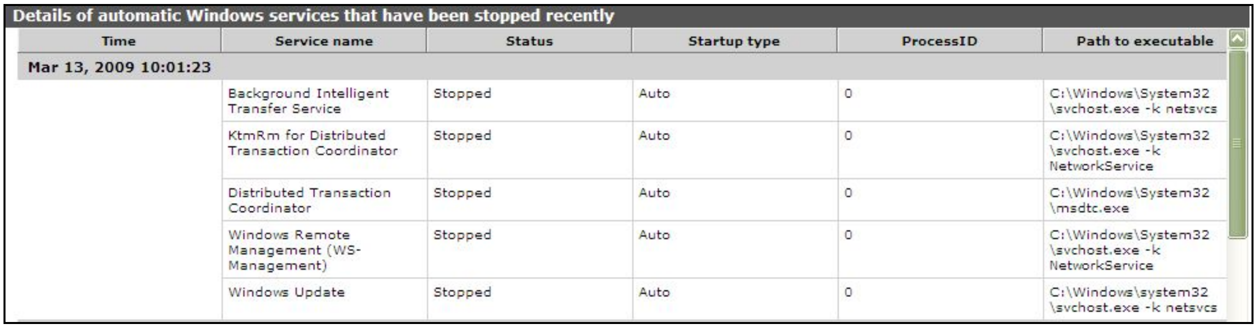


Details of automatic Windows services that have been started recently

Time	Service name	Status	Startup type	ProcessID	Path to executable
Mar 13, 2009 15:22:18					
	Cryptographic Services	Running	Auto	548	C:\Windows\system32 \svchost.exe -k NetworkServic
	DCOM Server Process Launcher	Running	Auto	816	C:\Windows\system32 \svchost.exe -k DcomLaunch
	DHCP Client	Running	Auto	972	C:\Windows\system32 \svchost.exe -k LocalServiceNetworkRestricted
	DNS Client	Running	Auto	548	C:\Windows\system32 \svchost.exe -k NetworkServic
	eGurkhaAgent	Running	Auto	1576	C:\eGurkha\lib\js.exe
	eGVmAgent	Running	Auto	1592	C:\eGVmAgent\lib\js.exe
	Windows Event Log	Running	Auto	972	C:\Windows\System32 \svchost.exe -k LocalServiceNetworkRestricted
	COM+ Event System	Running	Auto	344	C:\Windows\system32 \svchost.exe -k LocalService
	KtmRm for Distributed Transaction Coordinator	Running	Auto	548	C:\Windows\System32 \svchost.exe -k NetworkServic
	Workstation	Running	Auto	344	C:\Windows\System32 \svchost.exe -k LocalService
	TCP/IP NetBIOS Helper	Running	Auto	972	C:\Windows\system32 \svchost.exe -k LocalServiceNetworkRestricted

Figure 4.23: The detailed diagnosis of the New automatic services started measure

The detailed diagnosis of the *New automatic services stopped* measure lists the services that were stopped recently, the startup type, process ID, and the complete path to the executable that controls the service.

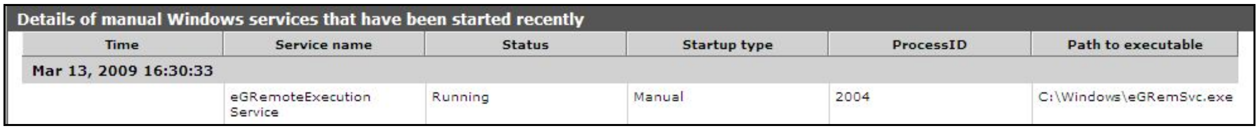


Details of automatic Windows services that have been stopped recently

Time	Service name	Status	Startup type	ProcessID	Path to executable
Mar 13, 2009 10:01:23					
	Background Intelligent Transfer Service	Stopped	Auto	0	C:\Windows\System32 \svchost.exe -k netsvcs
	KtmRm for Distributed Transaction Coordinator	Stopped	Auto	0	C:\Windows\System32 \svchost.exe -k NetworkService
	Distributed Transaction Coordinator	Stopped	Auto	0	C:\Windows\System32 \msdtc.exe
	Windows Remote Management (WS-Management)	Stopped	Auto	0	C:\Windows\System32 \svchost.exe -k NetworkService
	Windows Update	Stopped	Auto	0	C:\Windows\system32 \svchost.exe -k netsvcs

Figure 4.24: The detailed diagnosis of the New automatic services stopped measure

The detailed diagnosis of the *New manual services started* measure lists the services that were started recently, the startup type, process ID, and the complete path to the executable that controls the service.

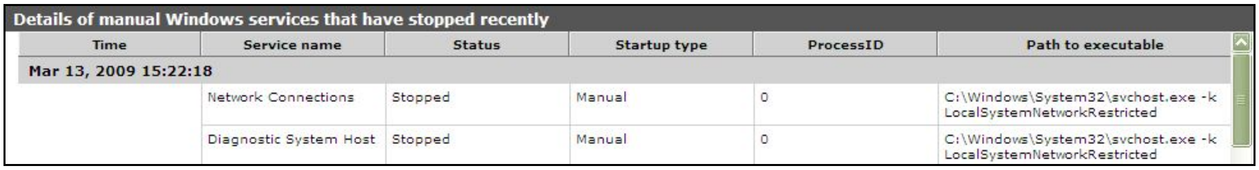


Details of manual Windows services that have been started recently

Time	Service name	Status	Startup type	ProcessID	Path to executable
Mar 13, 2009 16:30:33					
	eGRemoteExecution Service	Running	Manual	2004	C:\Windows\eGRemSvc.exe

Figure 4.25: The detailed diagnosis of the New manual services started measure

The detailed diagnosis of the *New manual services stopped* measure lists the services that were stopped recently, the startup type, process ID, and the complete path to the executable that controls the service.



Details of manual Windows services that have stopped recently

Time	Service name	Status	Startup type	ProcessID	Path to executable
Mar 13, 2009 15:22:18					
	Network Connections	Stopped	Manual	0	C:\Windows\System32\svchost.exe -k LocalSystemNetworkRestricted
	Diagnostic System Host	Stopped	Manual	0	C:\Windows\System32\svchost.exe -k LocalSystemNetworkRestricted

Figure 4.26: The detailed diagnosis of the New manual services stopped measure

4.8.13 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: A Hyper-V server

Agent executing the test: An internal agent

Output of the test: One set of results for each Windows VM on a monitored Hyper-V server

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. **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, 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 Section **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.
7. **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.

8. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.
9. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.
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. **DD FREQUENCY -** Refers to the frequency with which detailed diagnosis measures are to be

generated for this test. For a *Microsoft Hyper-V* 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 a Hyper-V 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**.

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 reported 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 modules stopped suddenly.
Recent service crashes:	Indicates the number of service crash events that occurred on thisVM during the last	Number	An event with the ID 7031 is logged in the Service Control Manager every time a service terminates ungracefully. This

Measurement	Description	Measurement Unit	Interpretation
	measurement period.		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

Measurement	Description	Measurement Unit	Interpretation
			IDs. Use the detailed diagnosis of this measure to know the complete details of the recent service errors.

4.8.14 Page File - VM Test

When the load imposed by applications and services running on a server nears the amount of installed RAM, additional storage is necessary. The page file serves as the temporary store on disk for memory that cannot be accommodated in the physical RAM. Since it is frequently accessed for storing and retrieving data that is needed for virtual memory access by application, the location and sizing of the page files can have a critical impact on server's performance. Ideally, the server operating system and the page file should be available on different drives for optimal performance. Splitting the page file across different drives can improve performance further.

A rule of thumb in sizing the page file is to set the maximum size of the page file to 1.5 times the available RAM. While this works well for systems with smaller physical memory, for other systems, the optimal page file size has to be determined based on experience using the system and studying the typical workload.

This test tracks the usage of each of the page files on a Windows VM. Note that this test is available for VMs running on Windows servers 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 *Hyper-V / Hyper-V VDI* as the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for every page file on a Windows server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **REPORTTOTAL** - Set this flag to **Yes** if you want the test to report total page file usage - i.e., the aggregate usage across multiple page files. In this case therefore, a **Total** descriptor will newly appear for this test in the eG monitoring console.

12. **REPORTTOTALONLY** - If both the **REPORTTOTAL** and **REPORTTOTALONLY** flags are set to **Yes**, then the test will report only the aggregate usage across multiple page files - in other words, the test will report values for the **Total** descriptor only. Likewise, if the **REPORTTOTAL** flag is set to **No**, and the **REPORTTOTALONLY** flag is set to **Yes**, then again, the test will report current usage for the **Total** descriptor only. However, if both the **REPORTTOTAL** and **REPORTTOTALONLY** flags are set to **No**, then the test will report individual usages only. Also, if the **REPORTTOTAL** flag is set to **Yes** and the **REPORTTOTALONLY** flag is set to **No**, then both the individual and Total usages will be reported.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current usage	Indicates the current usage of a page file.	Percent	This metric should be less than 90%. If the page file does not have additional space, additional users/processes cannot be supported and system performance will suffer. To improve

Measurement	Description	Measurement Unit	Interpretation
			performance, consider resizing the page file. Microsoft Windows allows a minimum and maximum size of the page file to be specified. If the system has sufficient disk space, consider setting the page file to start out at the maximum size (by using the same value for the minimum and maximum sizes), so that system resources are not spent growing the page file size when there is a virtual memory shortage.

4.8.15 Windows Security Center Status - VM Test

Windows Security Center (WSC) is a comprehensive reporting tool that helps administrators establish and maintain a protective security layer around Windows VMs to monitor the VM's health state. The Windows Security Center also monitors third party security products such as firewall, antivirus, antimalware and antispyware, installed on the VM. In order for the security products to be compliant with Windows and successfully report status to Action Center, these products should be registered with the security center. The security products communicate any subsequent status changes to the security center using private APIs. The security center, in turn, communicates these updates to Action Center, where they are finally displayed to the end user. With Windows Security Center, administrators can check whether any security product is installed and turned on, and if the definitions of the products are up to date and real-time protection is enabled. By continuously monitoring the Windows Security Center, administrators can instantly find out whether the security products are up-to-date or out dated, and the status of security products in real-time. This is what exactly the **Windows Security Center Status - VM** test does!

This test auto-discovers the security products installed on the Windsows VMs on the target host, and for each security product reports the current definition status and the current protection status. Using these details, administrators are alerted to the systems on which the automatic updates are outdated and virus protection turned off. By closely monitoring the status, administrators can take necessary actions before the end users become vulnerable to virus threats or malicious attacks.

Target of the test: A Hyper-V / Hyper-V VDI server

Agent executing the test: An internal agent

Output of the test: One set of results for every *security product:provider combination* on each Windows VMs on the target server.

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to ‘eG VM Agent (Windows)’**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.
7. **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.

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** 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.

9. **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.

10. **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.

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
Signature status	Indicates the current status of this security product.		The values reported by this measure and its numeric equivalents are mentioned 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>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Unknown</td><td>25</td></tr>
<tr><td>Up to date</td><td>15</td></tr>
<tr><td>Out of date</td><td>10</td></tr>
</table>
Note:
By default, this measure reports the **Measure Value**s listed in the table above to indicate the current state of this security product. The graph of this measure however, represents the status of a server using the numeric equivalents only.
Use the detailed diagnosis of this measure, to know about the name of Windows system on which the product is running, the file paths of product executables and the current status of the product.
</td></tr>
<tr><td>Real-time protection status</td><td>Indicates the real-time protection status of this security product.</td><td></td><td>
The values reported by this measure and its numeric equivalents are mentioned in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Unknown</td><td>25</td></tr>
<tr><td>Snoozed</td><td>20</td></tr>
<tr><td>On</td><td>15</td></tr>
<tr><td>Expired</td><td>10</td></tr>
<tr><td>Off</td><td>0</td></tr>
</table>
</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			Note: By default, this measure reports the **Measure Value**s listed in the table above to indicate the current protection status of this security product. The graph of this measure however, represents the status of a server using the numeric equivalents only.

4.9 Correlation Between Applications in a Hyper-V Virtualized Environment

Using the eG Enterprise administration console, administrators can add applications running on the VMs for monitoring. To monitor these applications, agents can be installed in the 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 *Hyper-V* server an application is running on. This mapping of applications to virtual servers is important for root-cause diagnosis – for example, a problem with the virtual 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 virtual servers. Whether eG Enterprise automatically determines the mapping of applications to virtual 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 virtual 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 a Hyper-V 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 *Microsoft Hyper-V* 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 *Virtual Server*.

The mapping of applications to virtual servers is used by eG Enterprise for correlation – e.g., since the application runs on the virtual server, it is most likely that a problem with the virtual server will impact the performance of the application running on one of the guests. To view this application-virtual server association, simply click on the **VIRTUAL TOPOLOGY** link in the layer model page of the virtual server.

Note:

The **VIRTUAL TOPOLOGY** link will also be available in the layer model page of those server applications that are executing on virtual guests.

Doing so reveals Figure 4.27 depicting the *Hyper-V* server and the server applications executing on it. By clicking on any of the components in Figure 4.27, the user can drill down into specific layers of this component for specific details on the performance of the component.

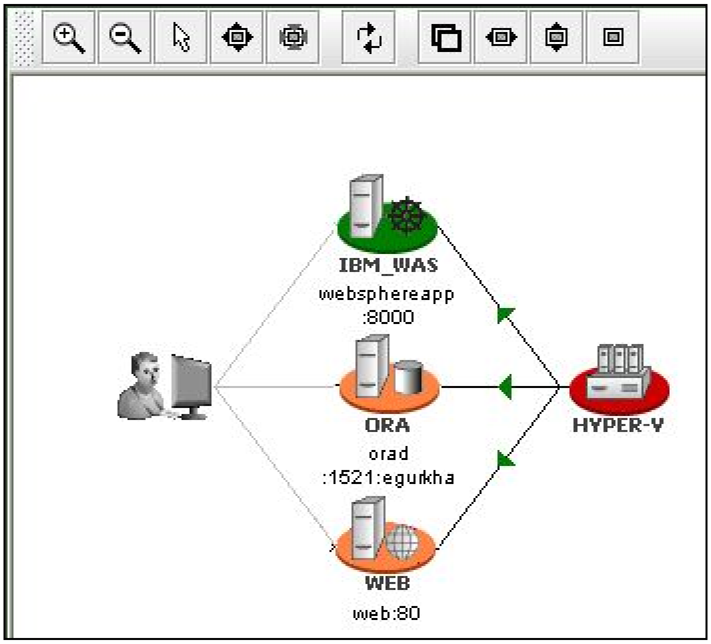


Figure 4.27: Depicts the applications that have been deployed on the guest OS of a virtual server

The arrows in Figure 4.27 depict the dependencies between the virtual server host and the applications running on it. Since the applications are hosted on one of the guests running on the host, they depend on the virtual server host – i.e., any unusual resource usage on the virtual server host impacts the applications running on any of the virtual guests. The dependency information between the virtual server host and the applications hosted on it is used by eG Enterprise for end-to-end correlation.

4.10 Troubleshooting the Failure of the eG Agent to Auto-discover the IP Addresses of VMs

If the eG agent is not able to discover the IP addresses of one/more VMs, then follow the steps given below:

1. Login to the root partition.
2. Go to the command prompt.
3. Switch to the directory, **<EG_INSTALL_DIR>\lib**.
4. Run the following command from that directory:

   cscript eG_HypervGuestInfo.vbs

5. If this script executes successfully, it would return the following information:

- The names of the VMs on the Hyper-V host;
- The current state of each VM (the value *2* if the VM is running, and the value *3* if it is powered-off);
- The Fully Qualified Domain Name (FQDN) of every VM
- The operating system on which each VM is executing

6. Make a note of the FQDN of a VM.
7. Then, issue the following command at the command prompt to identify the IP address of that VM:

   nslookup <FQDN>

8. If this command fails to return the IP address of the VM, it could mean that the IP address is not resolvable in the DNS server.
9. On the other hand, if the **eG_HypervGuestInfo.vbs** script itself fails to return the FQDN and the operating system of the VMs, it could indicate one/more of the following:

   - For VM discovery to occur, the **Integration Services** component should be installed on every target VM. Non-availability of the **Integration Services** component on a VM could cause the script to not report the FQDN and operating system of that VM.
   - The script may also fail if the target VMs are executing on any of the following Windows operating systems:
     - Windows Server 2008 64-bit
     - Windows Server 2008 x86
     - Windows Server 2003 x64 with SP2
     - Windows 2000 Server with SP4
     - Windows 2000 Advanced Server SP4
     - Windows Vista x64 with SP1
     - Windows Vista x86 with SP1
     - Windows XP x86 with SP2/SP3
     - Windows XP x64 with SP2

   This is because the **eG_HypervGuestInfo.vbs** script is not supported on any of the above-mentioned opeating systems.

10. Similarly, if a VM is executing on a Linux operating system, then again the **eG_ HypervGuestInfo.vbs** script will not be able to retrieve the FQDN of that VM.

Chapter 5: The Hyper-V VDI Monitoring Model

In some environments, the virtual guests hosted on Hyper- V 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 Hyper-V 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 Hyper-V server is a virtual guest running on?
- When was a guest moved from a Hyper-V Server? Which server was the guest moved to?
- Why was the guest migrated? What activities on the Hyper-V host caused the migration?

Using the *Hyper-V VDI* model (see Figure 5.1), 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 Hyper-V server and detect anomalies (if any) in its functioning.

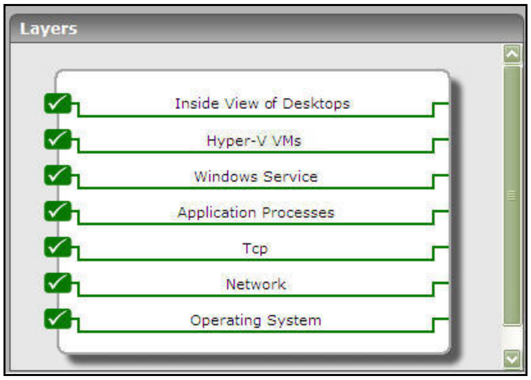


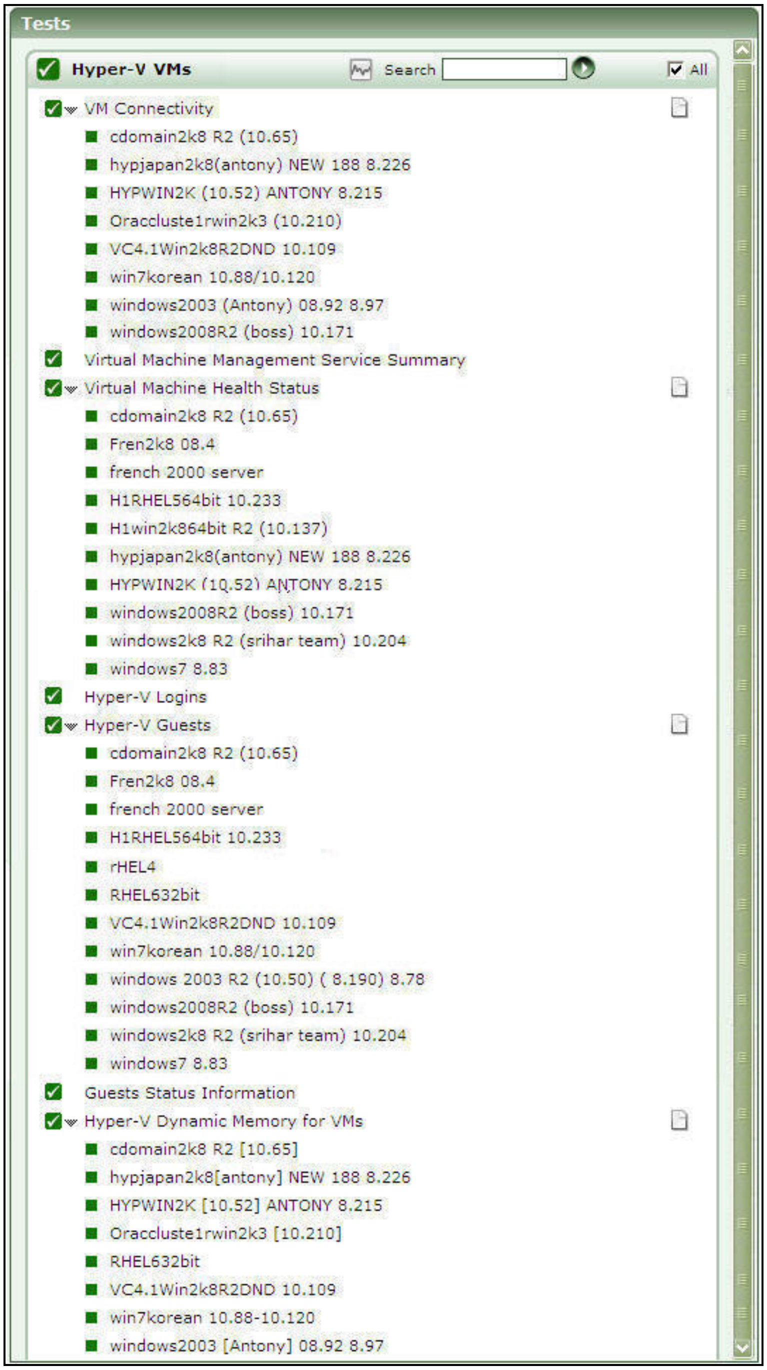
Figure 5.1: The layer model of a Hyper-V VDI server

The layers depicted by Figure 5.1 and the tests associated with the layers are discussed in detail in the sections that follow. Since the last 4 layers of the model have already been dealt with in the previous section, this section will discuss the **Hyper-V VMs** and the **Inside View of Desktops** layer only.

5.1 The Outside View of VMs Layer

The **Outside View of VMs** 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 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



Tests

☑ **Hyper-V VMs** Search ☑ All

- ☑ VM Connectivity
  - 🟩 cdomain2k8 R2 (10.65)
  - 🟩 hypjapan2k8(antony) NEW 188 8.226
  - 🟩 HYPWIN2K (10.52) ANTONY 8.215
  - 🟩 Oraccluste1rwin2k3 (10.210)
  - 🟩 VC4.1Win2k8R2DND 10.109
  - 🟩 win7korean 10.88/10.120
  - 🟩 windows2003 (Antony) 08.92 8.97
  - 🟩 windows2008R2 (boss) 10.171
- ☑ Virtual Machine Management Service Summary
- ☑ Virtual Machine Health Status
  - 🟩 cdomain2k8 R2 (10.65)
  - 🟩 Fren2k8 08.4
  - 🟩 french 2000 server
  - 🟩 H1RHEL564bit 10.233
  - 🟩 H1win2k864bit R2 (10.137)
  - 🟩 hypjapan2k8(antony) NEW 188 8.226
  - 🟩 HYPWIN2K (10.52) ANTONY 8.215
  - 🟩 windows2008R2 (boss) 10.171
  - 🟩 windows2k8 R2 (srihar team) 10.204
  - 🟩 windows7 8.83
- ☑ Hyper-V Logins
- ☑ Hyper-V Guests
  - 🟩 cdomain2k8 R2 (10.65)
  - 🟩 Fren2k8 08.4
  - 🟩 french 2000 server
  - 🟩 H1RHEL564bit 10.233
  - 🟩 rHEL4
  - 🟩 RHEL632bit
  - 🟩 VC4.1Win2k8R2DND 10.109
  - 🟩 win7korean 10.88/10.120
  - 🟩 windows 2003 R2 (10.50) (8.190) 8.78
  - 🟩 windows2008R2 (boss) 10.171
  - 🟩 windows2k8 R2 (srihar team) 10.204
  - 🟩 windows7 8.83
- ☑ Guests Status Information
- ☑ Hyper-V Dynamic Memory for VMs
  - 🟩 cdomain2k8 R2 [10.65]
  - 🟩 hypjapan2k8[antony] NEW 188 8.226
  - 🟩 HYPWIN2K [10.52] ANTONY 8.215
  - 🟩 Oraccluste1rwin2k3 [10.210]
  - 🟩 RHEL632bit
  - 🟩 VC4.1Win2k8R2DND 10.109
  - 🟩 win7korean 10.88-10.120
  - 🟩 windows2003 [Antony] 08.92 8.97

Figure 5.2: The tests associated with the Outside View of VMs layer

The **VM Connectivity** test, and the **Virtual Machine Management Service Summary** test have already been discussed in Chapter 2 of this document. Since the **Hyper-V Guests** test and **Guests Status Information** test reports additional measures for the *Hyper-V VDI* model, and because the

Hyper-V Logins test applies only to this model, the sections to come discuss these 3 tests alone in detail.

5.1.1 Hyper-V VM Details Test

This test monitors the amount of the physical server's resources that each guest on a Hyper-V 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, which disk has the maximum number of user sessions etc.

Target of the test : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results for *every user* or *guest* or *useronguest* to the Hyper-V 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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring of this document.
   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **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.

9. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.

10. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

12. **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.

13. 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
Current sessions:	This measure is relevant only for monitoring of virtual desktops (i.e., for Hyper-V VDI servers). When reporting metrics for specific users, this	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.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>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.</td><td></td><td></td></tr>
<tr><td>Is VM powered on?:</td><td>Whether the virtual machine is Hyper- V server host or not.</td><td></td><td>While the test reports a wide variety of other metrics too for virtual machines that are alive, only the Powered on status is indicated for virtual machines that are currently not available.
If this measure reports the value On, it indicates that the guest is up and running. The value Off could indicate that the guest has been powered-off; it could also indicate that the guest has moved to a different Hyper-V server.
The numeric values that correspond to each of the powered- on states discussed above are listed in the table below:
<table><tr><th>State</th><th>Value</th></tr><tr><td>On</td><td>1</td></tr><tr><td>Off</td><td>0</td></tr></table>
<b>Note:</b>
By default, this measure reports</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			the values On or Off to indicate the status of a VM. The graph of this measure however, represents the status of a VM using the numeric equivalents - 0 or 1.
Virtual CPU allocated to VM:	Indicates the number of processors present in this VM.	Number	All execution in the root and child partitions (where guest VMs run) happens on Virtual Processors (VPs). At a minimum, you will see one VP for each Logical Processor (LP). These account for the root VPs. You will then see one for each VP you have configured to a guest. Therefore, if you have an 8LP system with 1 guest running with 2 VPs, the count here will be 10.
Virtual CPU Utilization of VM:	Indicates the percentage of time spent by the virtual processor assigned to this VM in guest and hypervisor code.	Percent	This measure serves as an effective indicator of how resource-intensive a particular VM is on a specific Hyper-V server.
Virtual machine runtime:	Indicates the percentage of time spent by the virtual processor in guest code.	Percent	Comparing the value of the Virtual machine runtime and Hypervisor runtime measures for every VM will reveal where the virtual processors of the VM have spent more time – in processing guest code or in processing hypervisor code?
Hypervisor runtime:	Indicates the percentage of time the virtual processor spend in hypervisor code.	Percent	
Memory allocated to VM:	Indicates the amount of physical memory	MB	

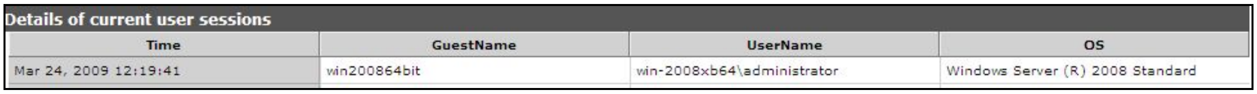
Measurement	Description	Measurement Unit	Interpretation
	allocated to this VM.		
Data transmitted by VM:	Indicates the number of bytes per second sent over the network adapters supported by this VM.	Mbps	
Data received by VM:	Indicates the number of bytes per second sent over the network adapters supported by this VM.	Mbps	
Data dropped by VM:	Indicates the number of bytes dropped on the network adapter.	MB	Ideally, this value should be very low. A high value could be indicative of a network bottleneck.
Disk reads by VM:	Indicates the number of bytes read per second from the disks attached to the IDE controller.	MB/Sec	These measures are good indicators of the activity on the disks attached to the IDE controller.
Disk writes by VM:	Indicates the the number of bytes written per second to the disks attached to the IDE controller.	MB/Sec	
Deposited pages:	Indicates the number of memory pages deposited into the partition.	Number	For each partition, the hypervisor maintains a memory pool of RAM SPA pages. This pool acts just like a checking account. The amount of pages in the pool is called the balance. Pages are deposited or withdrawn from the pool. When a hypercall that requires memory is made by a partition, the hypervisor withdraws the required memory

Measurement	Description	Measurement Unit	Interpretation
			from the total pool balance of that partition. When the balance available in the pool is less, then more memory pages are deposited in the pool. A very high value of this measure therefore, indicates that the balance in the pool maintained for this partition is dwindling. This is a cause for concern.
Hypercalls:	Indicates the rate of hypercalls made by this guest's code on the virtual processor.	Hypercalls/Sec	Hypercalls are one form of enlightenment. Guest OS's use the enlightenments to more efficiently use the system via the hypervisor. TLB flush is an example hypercall. If this value is zero, it is an indication that Integration Components are not installed. New OS's like WS08 can use hypercalls without enlightened drivers. So, hypercalls are only a prerequisite and not a guarantee for not having Integration Components installed.
Control register accesses:	Indicates the rate of control register accesses by this guest on its virtual processors.	Accesses/Sec	Control registers are used to set up address mapping, privilege mode, etc.
HLT instructions:	Indicates the rate of HLT instructions executed by this guest on its virtual processors.	Instructions/Sec	A HLT will cause the hypervisor scheduler to de- schedule the current VP and move to the next VP in the runlist.

Measurement	Description	Measurement Unit	Interpretation
Emulated instruction:	Indicates the rate of emulated instructions while executing guest code on the virtual processor.	Instructions/Sec	
MWAIT instructions:	Indicates the rate of MWAIT instructions executed by this guest on its virtual processors.	Instructions/Sec	The mwait (monitored wait) instruction instructs the processor to enter a wait state in which the processor is instructed to monitor the address range between a and b and wait for an event or a store to that address range.
CPUID instructions:	Indicates the rate of CPUID instructions executed by this guest on its virtual processors.	Instructions/Sec	The CPUID instruction is used to retrieve information on the local CPU's capabilities. Typically, CPUID is only called when the OS / Application first start. Therefore, this value is likely to be 0 most of the time.
Page fault intercepts:	Indicates the rate of page fault exceptions intercepted by the hypervisor while executing this guest's code on the virtual processor	Intercepts/Sec	Whenever guest code accesses a page not in the CPU TLB a page fault will occur. This counter is closely correlated with the Large Page TLB Fills measure.
Total intercepts :	Indicates the rate of hypervisor intercept messages.	Intercepts/Sec	Whenever a guest VP needs to exit its current mode of running for servicing in the hypervisor, this is called an intercept. Some common causes of intercepts are resolving Guest Physical Address (GPA) to Server Physical Address (SPA) translations, privileged

Measurement	Description	Measurement Unit	Interpretation
			instructions like hlt / cupid / in / out, and the end of the VP's scheduled time slice.
Large page TLB fills:	Indicates the rate of virtual TLB fills on large pages.	Fills/Sec	There are two types of TLB entries (and some three). Small TLB which generally means a 4K page and large Page which generally means 2MB. There are fewer Large TLB entries on the order of 8 – 32. A non- zero value for this measures indicates that the root partition is using large pages.
Small page TLB fills:	Indicates the rate of virtual TLB fills on 4K pages.	Fills/Sec	There are two types of TLB entries (and some three). Small TLB which generally means a 4K page and large Page which generally means 2MB. There are fewer Large TLB entries on the order of 64 – 1024+.
Cpu utilization of VM:	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.

The detailed diagnosis of the *Current sessions* measure reveals the guests to which the user is currently logged in.



Details of current user sessions

Time	GuestName	UserName	OS
Mar 24, 2009 12:19:41	win200864bit	win-2008xb64\administrator	Windows Server (R) 2008 Standard

Figure 5.3: The detailed diagnosis of the Current sessions measure

5.1.2 Hyper-V VM Information Test

Hyper-V™ live migration is designed to move running VMs with no impact on VM availability to users. By pre-copying the memory of the migrating VM to the destination physical host, live migration minimizes the amount of transfer time of the VM A live migration is deterministic, meaning that the administrator, or script, that initiates the live migration can control which computer is the destination for the live migration. The guest operating system in the migrating VM is unaware that the migration is happening, so no special configuration for the guest operating system is needed.

Below is a summary of the live migration process:

- All VM memory pages are transferred from the source Hyper-V™ physical host to the destination Hyper-V™ physical host. While this is occurring, any VM modifications to its memory pages are tracked.
- ™Pages that were modified while step 1 was occurring are transferred to the destination physical computer.
- The storage handle for the VM's VHD files are moved to the destination physical computer.
- The destination VM is brought online on the destination Hyper-V™ server.

This test reports the number of guests registered with the server, and promptly alerts administrators to addition/removal of guests from the server.

Target of the test : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.
7. **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.

8. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.
9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *Microsoft Hyper-V* 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 a Hyper-V 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**.

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.

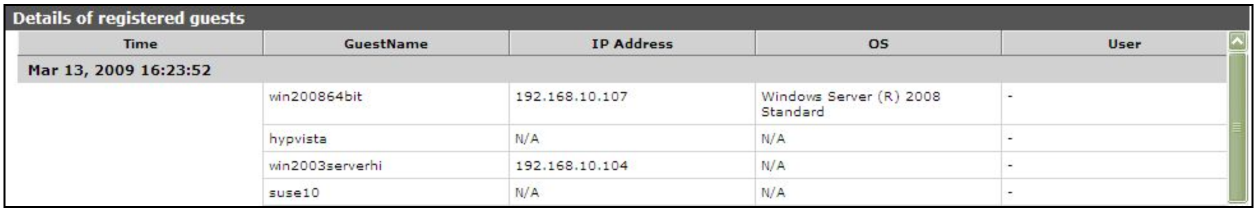
    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

    - 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 reported by the test

Measurement	Description	Measurement Unit	Interpretation
Registered VMs	Indicates the total number of virtual machines that have been registered with the server currently.	Number	
VMs 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).
VMs with users	Indicates the number of powered on guests with users logged in currently.	Number	To know which guests the users have logged into, use the detailed diagnosis capability of this measure (if enabled).
VMs without users	Indicates the number of powered on guests without any users logged in currently.	Number	To know which guests no user has logged into, use the detailed diagnosis capability of this measure (if enabled).
Added VMs	Indicates the number of guests that were newly added to the server during this measurement period.	Number	The detailed diagnosis of these measures, if enabled, lists the virtual machines that were migrated to or from (as the case may be) the Hyper-V server.
Removed VMs	Indicates the number of guests that were newly removed from the server during this measurement period.	Number	

The detailed diagnosis of the *Registered VMs* measure reports the name of the guests registered with the Hype-V server, the IP address of the guests, the guest OS, and the name of the user currently logged into the guest.

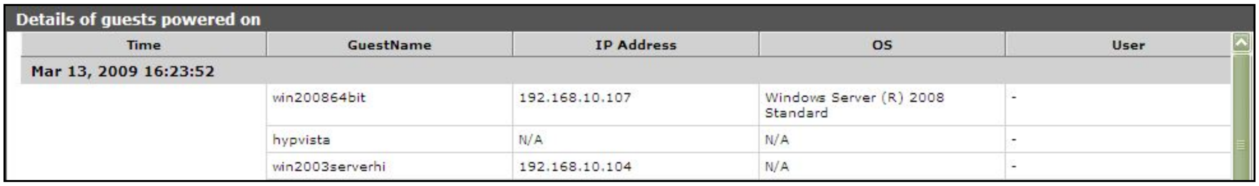


Details of registered guests

Time	GuestName	IP Address	OS	User
Mar 13, 2009 16:23:52				
	win200864bit	192.168.10.107	Windows Server (R) 2008 Standard	-
	hypvista	N/A	N/A	-
	win2003serverhi	192.168.10.104	N/A	-
	suse10	N/A	N/A	-

Figure 5.4: The detailed diagnosis of the Registered guests measure

The detailed diagnosis of the *VMs powered on* measure reports the name of the guests currently powered on, the IP address of the guests, the guest OS, and the name of the user currently logged into the guest.



Details of guests powered on

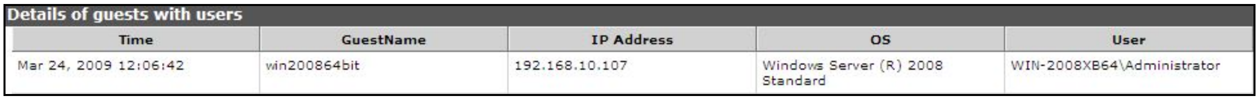
Time	GuestName	IP Address	OS	User
Mar 13, 2009 16:23:52				
	win200864bit	192.168.10.107	Windows Server (R) 2008 Standard	-
	hypvista	N/A	N/A	-
	win2003serverhi	192.168.10.104	N/A	-

Figure 5.5: The detailed diagnosis of the Guests powered on measure

Note:

The eG agent can extract the name and “outside view” metrics of Linux guests, but can neither discover the IP address nor report “inside view” metrics pertaining to Linux guests. Similarly, the eG agent cannot discover the IP address or obtain the “inside view” of those Windows VMs which do not support **Key/Value Pair Exchange** script

The detailed diagnosis of the *VMs with users* measure reveals the name, IP, and OS of the guests to which users are currently logged in, and the names of the users who have logged in.



Details of guests with users

Time	GuestName	IP Address	OS	User
Mar 24, 2009 12:06:42	win200864bit	192.168.10.107	Windows Server (R) 2008 Standard	WIN-2008XB64\Administrator

Figure 5.6: The detailed diagnosis of the VMs with users measure

5.1.3 Hyper-V Logins Test

This test monitors the user logins to guests and reports the total count of logins and logouts.

Target of the test : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results for *every user* or *guest* or *useronguest* to the Hyper-V 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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.
7. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *Microsoft Hyper-V* 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 a Hyper-V 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**.
8. 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
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	If this measure reports a non-zero value, use the detailed diagnosis of the measure to know which user logged into which VM, when.
Percentage of 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.

5.1.4 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 : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results each distinct application that is being accessed by users of 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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the REPORT BY USER flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.
7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **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.

12. **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**.

13. **DD FREQUENCY -** Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *Microsoft Hyper-V* 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 a Hyper-V 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**.

14. 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 reported 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	

Measurement	Description	Measurement Unit	Interpretation
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.
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 physcial CPU (in Mhz) used up by this application.	Mhz	

5.1.5 Application Process Launches - VM Test

When a user complains that it is taking too long to launch applications on virtual desktops, administrators must be able to quickly identify the applications that are being currently accessed by that user, know how much time each application took to launch, and thus pinpoint that application that is the slowest in launching. The **Application Process Launches - VM** test provides these valuable insights to the administrators. This test auto-discovers all the applications that are currently launched on the virtual desktops, and for each discovered application, reports the average and maximum time that application took to launch. This way, the test points administrators to applications that are slow in launching. Detailed diagnostics provided by the test also reveals the users who are currently accessing the applications and the launch time of the application as perceived by each user session;

in the process, the test accurately pinpoints which user was attempting to launch the application when the slowness was observed.

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 the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Hyper-V server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every application on the virtual desktops that is currently launched.

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. **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, 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 **2.2.2** 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*.
5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. 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
Launch count	Represents the number of instances of this application that have been launched currently.	Number	Use the detailed diagnosis of this measure to know which users are currently accessing the application and the time it took for every user to launch the application.
Avg time to launch application	Indicates the average time taken by this application to launch.	Secs	Compare the value of this measure across applications to knsow which application took the longest time to launch. User experience with this application will naturally be poor.
Max time to launch application	Indicates the maximum time taken by this application to launch.	Secs	Compare the value of this measure across applications to know which application registered the highest launch time during the last measurement period. To know which user experienced this delay in launching, use the detailed diagnosis of the Launch count measure.

5.1.6 Hyper-V Memory Usage Test

This test reports how much physical memory has been allocated to the VMs on the Hyper-V host. With the help of this test, administrators can proactively detect a potential memory contention on the Hyper-V host caused due to improper resource allocation to the VMs. The detailed diagnosis of this test also sheds light on that VM that could be over-sized with memory.

Target of the test : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results for the Hyper-V 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.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Available memory for VMs	Indicates the amount of physical memory that is still unused by the VMs.	MB	A high value is desired for this measure.
Memory used by VMs	Indicates the percentage of physical memory allocated to the VMs on the host.	Percent	Ideally, the value of this measure should be low. A value close to 100% is indicative of excessive memory allocation to the VMs. You can use the detailed diagnosis of this measure to identify the VM to which maximum physical memory has been allocated.
Total memory available	Indicates the total physical memory capacity of the Hyper-V host.	MB	
Memory allocated to VMs	Indicates the total amount of memory allocated to VMs.	MB	Ideally, the value of this measure should be low.

5.1.7 Hyper-V Dynamic VHDs Test

A VHD (Virtual Hard Disk) is a dynamically expanding disk on the Hyper-V server, which is initially at a few kilobytes and expands when the VMs need additional storage space. The VHDs can only grow as the VMs add data, however, to the size limit which is designated in the setup wizard. The VHDs enable administrators to create, configure, and boot physical computers without a virtual machine or hypervisor. This functionality simplifies image management because it allows to:

- Standardize the image format and toolsets in the organization.
- Reduce the number of images to catalog and support.
- Enable increased server utilization to conserve energy.

If one/more VMs on the Hyper-V host utilize more dynamic memory resources (from the VHDs) than they are supposed to use, then, other VMs on the host may not have sufficient memory resources resulting in serious memory contention. To avoid memory resource contention among the VMs that use the VHDs, it is necessary to continuously track the dynamic memory allocated to the VMs from time-to-time. The **Hyper-V Dynamic VHDs** test helps administrators in this regard!

This test auto-discovers the VHDs of the target Microsfot Hyper-V server and reports the memory utilization of each VHD. Using this test, administrators can detect the VHD that is hogged continuously for memory resources.

Target of the test : A Microsoft Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for each *VHD* attached to the Microsoft Hyper-V VDI 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 -** Refers to the port at which the specified host listens to.

Measurements reported by the test

Measurement	Description	Measurement Unit	Interpretation
Total size	Indicates the total capacity of this VHD.	MB	
Used size	Indicates the amount of memory that is utilized by the VMs from this VHD.	MB	By comparing the value of this measure across the VHDs, you will be able to identify the VHD that is being utilized at all times.

Measurement	Description	Measurement Unit	Interpretation
			If the value of this measure is close to the Total size measure, then, there may not be adequate memory resources for all the VMs associated with the VHD. This may cause performance bottlenecks in the VMs that are using the VHD.
Free Size	Indicates the amount of memory that is available for use by the VMs in this VHD.	MB	Ideally, the value of this measure should be high.
Percent usage	Indicates the percentage of memory utilized on this VHD by the VMs.	Percent	Ideally, the value of this measure should be low. If the value of this measure is nearing 100%, then, there may be a performance bottleneck in the VMs that are using the VHD due to memory resource contention.

5.2 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 **Inside View of Desktops** layer provide an "internal" view of the workings of each of the guests - these tests execute on an Hyper-V 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 Desktops** layer, does not list the associated tests. Instead, Figure 5.7 appears. This figure displays the current state of all virtual desktops that have been configured on the monitored Hyper-V host.

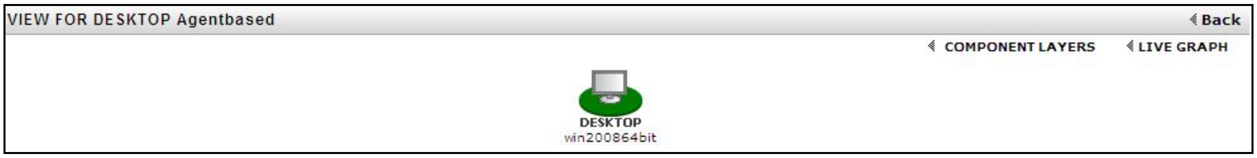
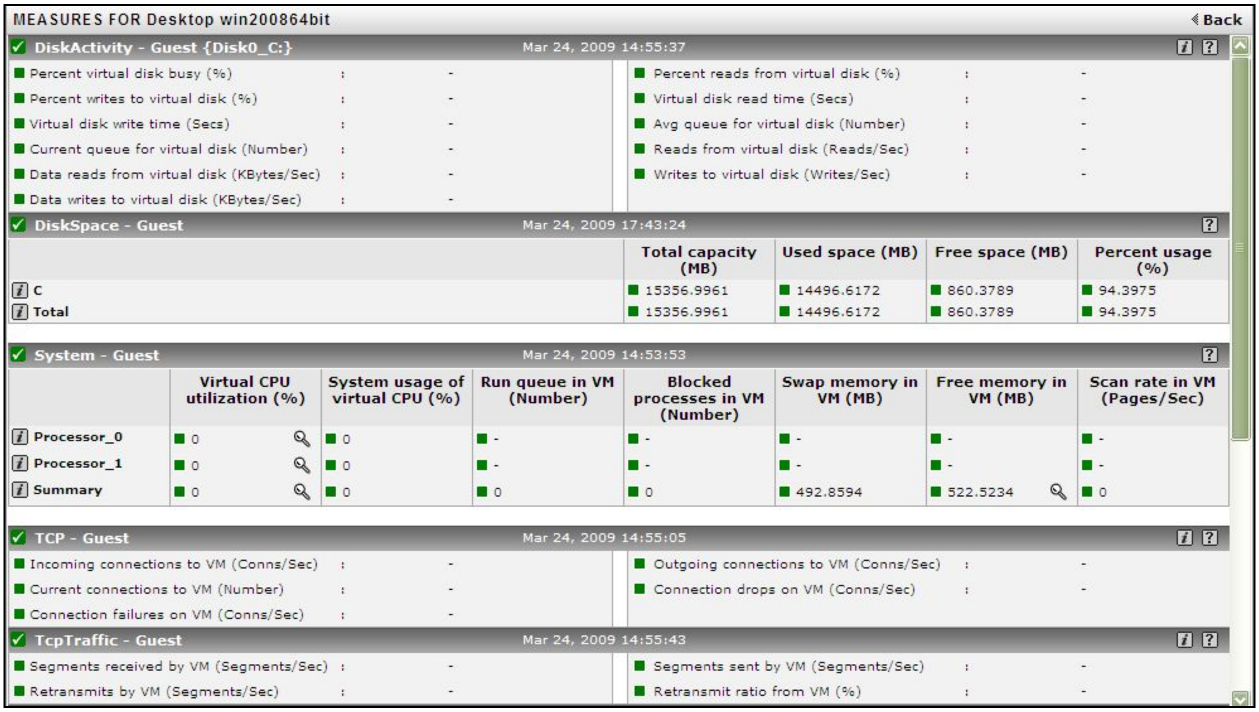


Figure 5.7: The current state of the desktops configured on the Hyper-V host that is monitored

Clicking on any of the guests in the **Desktop view** leads you to Figure 5.8 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 5.8.


MEASURES FOR Desktop win200864bit

Back

✓ DiskActivity - Guest {Disk0_C:} Mar 24, 2009 14:55:37

🟩 Percent virtual disk busy (%)	:	-
🟩 Percent writes to virtual disk (%)	:	-
🟩 Virtual disk write time (Secs)	:	-
🟩 Current queue for virtual disk (Number)	:	-
🟩 Data reads from virtual disk (KBytes/Sec)	:	-
🟩 Data writes to virtual disk (KBytes/Sec)	:	-
🟩 Percent reads from virtual disk (%)	:	-
🟩 Virtual disk read time (Secs)	:	-
🟩 Avg queue for virtual disk (Number)	:	-
🟩 Reads from virtual disk (Reads/Sec)	:	-
🟩 Writes to virtual disk (Writes/Sec)	:	-

✓ DiskSpace - Guest Mar 24, 2009 17:43:24

	Total capacity (MB)	Used space (MB)	Free space (MB)	Percent usage (%)
C	🟩 15356.9961	🟩 14496.6172	🟩 860.3789	🟩 94.3975
Total	🟩 15356.9961	🟩 14496.6172	🟩 860.3789	🟩 94.3975

✓ System - Guest Mar 24, 2009 14:53:53

	Virtual CPU utilization (%)	System usage of virtual CPU (%)	Run queue in VM (Number)	Blocked processes in VM (Number)	Swap memory in VM (MB)	Free memory in VM (MB)	Scan rate in VM (Pages/Sec)
Processor_0	🟩 0	🟩 0	🟩 -	🟩 -	🟩 -	🟩 -	🟩 -
Processor_1	🟩 0	🟩 0	🟩 -	🟩 -	🟩 -	🟩 -	🟩 -
Summary	🟩 0	🟩 0	🟩 0	🟩 0	🟩 492.8594	🟩 522.5234	🟩 0

✓ TCP - Guest Mar 24, 2009 14:55:05

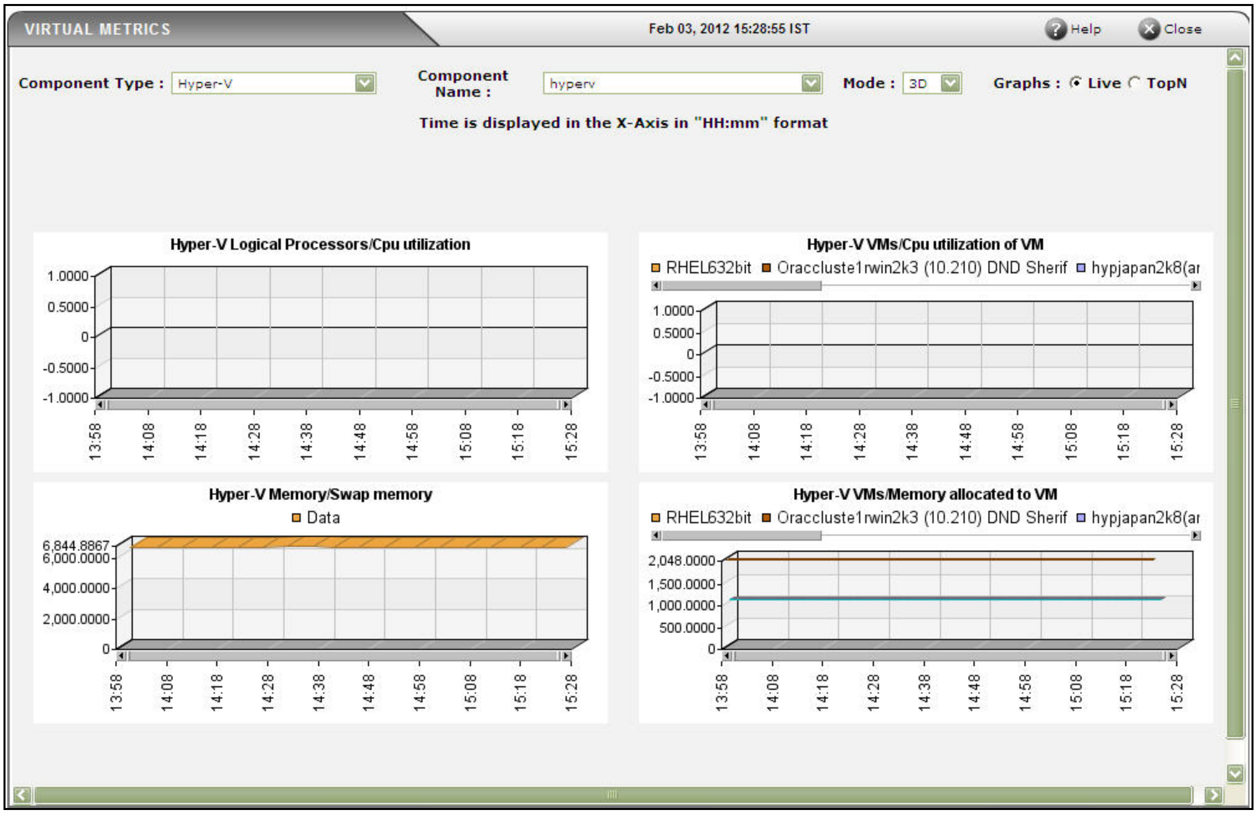
🟩 Incoming connections to VM (Conns/Sec)	:	-
🟩 Current connections to VM (Number)	:	-
🟩 Connection failures on VM (Conns/Sec)	:	-
🟩 Outgoing connections to VM (Conns/Sec)	:	-
🟩 Connection drops on VM (Conns/Sec)	:	-

✓ TcpTraffic - Guest Mar 24, 2009 14:55:43

🟩 Segments received by VM (Segments/Sec)	:	-
🟩 Retransmits by VM (Segments/Sec)	:	-
🟩 Segments sent by VM (Segments/Sec)	:	-
🟩 Retransmit ratio from VM (%)	:	-

Figure 5.8: Measures pertaining to a chosen guest

To view real-time graphs of pre-configured measures (pertaining to the *Microsoft Hyper-V* host and the guests operating on it), click on the **LIVE GRAPH** link. Alternatively, you can click on the icon that appears in the **Tests** panel when the **Outside View of VMs layer** is clicked. The graph display that appears subsequently (see Figure 5.9) 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 utilization' measure of the *Hyper-V Logical Processors* test, you will find a graph of the 'Virtual machine cpu utilization' measure of the *Hyper-V VMs* test. This way, you can easily compare and correlate how well the physical CPU resources are being utilized by both the Hyper-V 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 Hyper-V host? or is it the virtual guest? If you access this page from the **LIVE GRAPH** link, then, by default, you will view live graphs pertaining to the *Hyper-V VDI* server. However, you can select a different virtualized component- type and a different virtualized component using the **type** and **ComponentName** lists (respectively).



VIRTUAL METRICS

Feb 03, 2012 15:28:55 IST

Component Type : Hyper-V

Component Name : hyperv

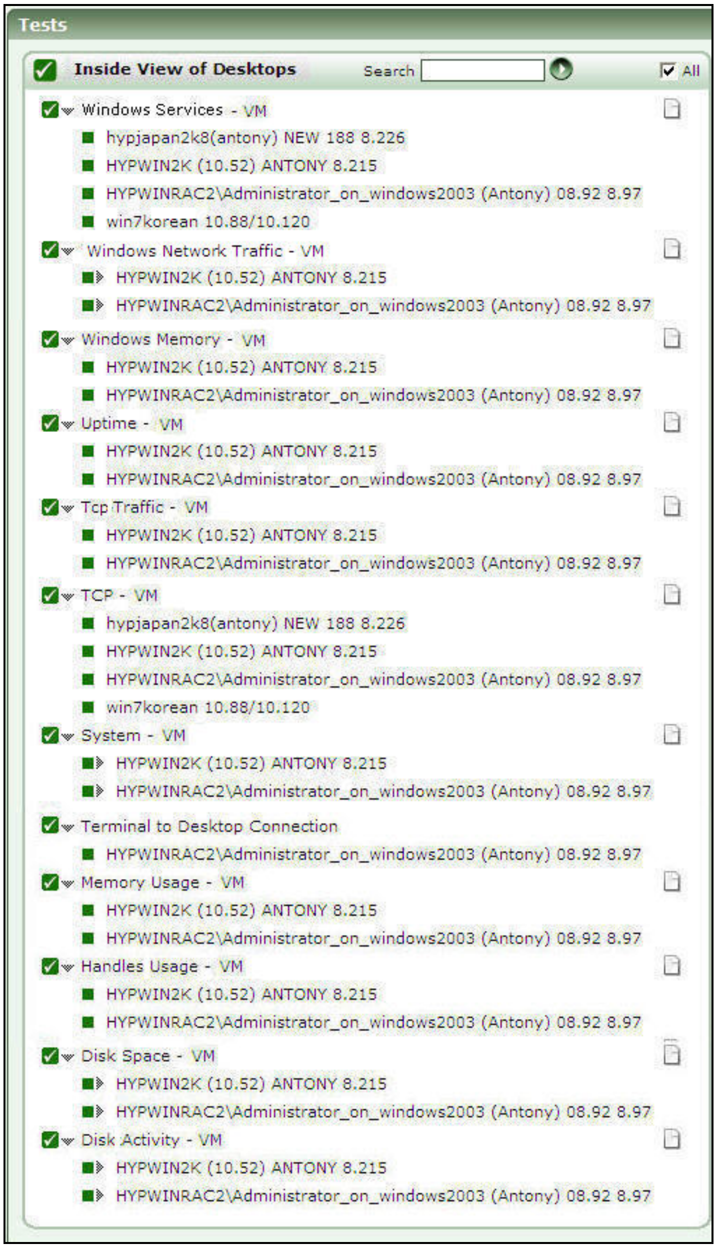
Mode : 3D

Graphs : ● Live ○ TopN

Time is displayed in the X-Axis in "HH:mm" format

Figure 5.9: Live graph comparing physical resource usage of a Hyper-V VDI server (on the left) and resource usage levels of the individual VMs (on the right)

To return to the layer model of the *Microsoft Hyper-V VDI* server and view the tests mapped to the **Inside View of Desktops** layer, click on the **COMPONENT LAYERS** link in Figure 5.7. The tests depicted by Figure 5.10 then appear.



Tests

☑ **Inside View of Desktops** Search ☑ All

- ☑ Windows Services - VM
  - hypjapan2k8(antony) NEW 188 8.226
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
  - win7korean 10.88/10.120
- ☑ Windows Network Traffic - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ Windows Memory - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ Uptime - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ Tcp Traffic - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ TCP - VM
  - hypjapan2k8(antony) NEW 188 8.226
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
  - win7korean 10.88/10.120
- ☑ System - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ Terminal to Desktop Connection
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ Memory Usage - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ Handles Usage - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ Disk Space - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97
- ☑ Disk Activity - VM
  - HYPWIN2K (10.52) ANTONY 8.215
  - HYPWINRAC2\Administrator_on_windows2003 (Antony) 08.92 8.97

Figure 5.10: The tests associated with the Inside View of Desktops layer

Alternatively, you can also configure eG to first display the tests mapped to the **Inside View of Desktops** layer first upon clicking it, and not the **Desktop View**. For this, 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.

5.2.1 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 a Hyper-V host from remote terminals. 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 a Hyper-V host. This test auto-discovers the virtual desktops on the 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 Terminal to Desktop Connection test alerts administrators to such situations.

Note:

This test will work on Windows VMs only.

Target of the test : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for the user who is currently logged into each Windows virtual desktop on the server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **PACKETSIZE -** The size of packets used for the test (in bytes)
12. **PACKETCOUNT -** The number of packets exchanged between the virtual desktop and the user terminal during the test
13. **TIMEOUT -** How long after transmission should a packet be deemed lost (in seconds)
14. **PACKETINTERVAL -** Represents the interval (in milliseconds) between successive packet transmissions during the execution of this test.
15. **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.

Measurements reported 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.
Average delay	Indicates the average delay between transmission of a request by the agent on a virtual desktop and	Secs	Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when

Measurement	Description	Measurement Unit	Interpretation
	receipt of the response back from the user terminal.		connecting to a virtual desktop.
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.

5.2.2 Desktop's HDX Channel Test

As already mentioned, 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 Hyper-V 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 *Microsoft Hyper-V - 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.

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 : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results will be reported 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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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.

Measurements reported 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 output	Indicates the bandwidth used while transmitting sound/audio to this	Kbps	Comparing these values across users will reveal which user is sending/receiving bandwidth-

Measurement	Description	Measurement Unit	Interpretation
	user.		intensive sound/audio files over the ICA channel.
Audio bandwidth input	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 input	Indicates the bandwidth used when this user performs file operations on the mapped drive on the virtual desktop.	Kbps	Comparing the values of these measures across users will reveal which user is performing bandwidth-intensive file operations over the ICA channel.
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	

Measurement	Description	Measurement Unit	Interpretation
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 operatons 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. 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	Indicates the bandwidth used from this user to	Kbps	Comparing the values of these measures across users will reveal

Measurement	Description	Measurement Unit	Interpretation
acceleration bandwidth input	virtual desktop for multimedia traffic.		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.
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	KB/Sec	

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>line speed of all sessions from the desktop to this user.</td><td></td><td></td></tr>
<tr><td>Bandwidth usage</td><td>Indicates the percentage HDX bandwidth consumption of this user.</td><td>Percent</td><td>Compare the value of this measure across users to know which user is consuming the maximum HDX bandwidth.</td></tr>
<tr><td>User's connection quality indicator:</td><td>Indicates the connectivity of this user with the Citrix environment.</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>Poor connection</td><td>1</td></tr>
<tr><td>Weak connection</td><td>2</td></tr>
<tr><td>Strong connection</td><td>3</td></tr>
<tr><td>None</td><td>4</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above. In the graph of this measure however, the value of this measure is represented using their numeric equivalents only.
By default, Citrix recommends a standard computation of user’s connection quality indicator as mentioned in the table below:</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>Connection Quality Indicator</th><th>How is the Connection Quality Indicator calculated?</th></tr>
<tr><td>Weak</td><td>Reported when
• Bandwidth > 1MBPs
• Latency <= 150ms
• ICA RTT <= 180ms</td></tr>
<tr><td>Strong</td><td>Reported when
• Bandwidth > 8 MBPs
• Latency <= 150ms
• ICA RTT <= 180ms</td></tr>
<tr><td>None</td><td>Reported when
• Bandwidth <= 0 MBPs
• Latency < 0
• ICA RTT < 0</td></tr>
<tr><td>Poor</td><td>Reported when any condition other than the above is noticed.</td></tr>
</table>
</td></tr>
</table>

5.2.3 PCoIP Session - VM Test

PCoIP - PC over IP - is a proprietary protocol for remote workstation and desktop resolution. Hyper-V 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 Hyper-V server 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 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.

Note:

This test is relevant only where VMware Horizon View is used to broker connections between the user and the desktops.

That is why, 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 *Microsoft Hyper-V - 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 : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results will be reported 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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.
7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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.

Measurements reported 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

Measurement	Description	Measurement Unit	Interpretation
			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	
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	

Measurement	Description	Measurement Unit	Interpretation
USB data received rate	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 rate	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	

5.2.4 Citrix VDA Status - VM Test

Citrix Virtual Delivery Agent (VDA) is installed on a virtual machine that runs the applications or virtual desktops for the user. The VDA enables connections between the applications/desktops and the users only when the connections are brokered by Citrix. The VDA enables the virtual machines to register with Delivery Controllers and manage the High Definition experience (HDX) connection to a user device. If the VDA failed to register with a delivery controller, it would not be possible for the delivery controller to broker a connection to the target virtual machine. The target virtual machine would therefore become an unusable resource. The VDA issues with respect to registration are logged in the event log of the target virtual server. Some of the most common issues that are logged into the event log are the virtual desktop not added to the correct desktop farm, the virtual desktop firewall not configured properly, DNS configuration failure, Time synchronization failure, WCF failure etc. The eG agent integrates with the XDPing to collect the metrics that details on what exactly was the reason behind the registration issues i.e., what was the service that failed. The

Citrix VDA Status - VM test helps administrators to figure out which service has failed leading to VDA registration issues!

This test monitors the VDA installed on the target virtual machine and reports whether the services such as user authentication, active directory authentication, DNS lookup, WCF endpoints etc are successful or not. This test also reports the errors and warnings available in the event log when registration failure occurs.

Note:

This test reports metrics only if the connections to the applications and desktops are brokered via Citrix XenDesktop 7.x.

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 the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Hyper-V server

Agent deploying the test : A remote agent

Outputs of the test : One set of results for the virtual machine monitored.

Configurable parameters for the test

1. **TEST PERIOD -** How often should the test be executed. By default, this is set to 15 minutes.
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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring.
   - **If the INSIDE VIEW USING flag is set to ‘eG VM Agent (Windows)’**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.

7. **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.

8. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.

9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *6:1*. 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**.

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

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Machine account status</td><td>Indicates the current status of the account of the machine on which the VDA was installed.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Failed</td><td>0</td></tr><tr><td>Success</td><td>1</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>Note:
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the account of the machine. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>User authentication status</td><td>Indicates the current status of the User authentication service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Failed</td><td>0</td></tr><tr><td>Success</td><td>1</td></tr></table>Note:
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the user authentication service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>Domain controller time sync status</td><td>Indicates the current status of the Domain controller time sync service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Failed</td><td>0</td></tr><tr><td>Success</td><td>1</td></tr></table>Note:
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of</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>the Domain controller time sync service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>WCF endpoint status</td><td>Indicates the current status of the WCF endpoint service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Failed</td><td>0</td></tr>
<tr><td>Success</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the WCF endpoint service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>VDA windows service status</td><td>Indicates the current status of the VDA Windows service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Failed</td><td>0</td></tr>
<tr><td>Success</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the VDA Windows service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>DNS lookup status</td><td>Indicates the current</td><td></td><td>The values that this measure can report</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>status of the DNS lookup service.</td><td></td><td>and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Failed</td><td>0</td></tr>
<tr><td>Success</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the DNS lookup service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>Windows firewall status</td><td>Indicates the current status of the Windows firewall service.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Failed</td><td>0</td></tr>
<tr><td>Success</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current status of the Windows firewall service. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.</td></tr>
<tr><td>Registration status</td><td>Indicates the registration status of the VDA.</td><td></td><td>The values that this measure can report and its corresponding numeric equivalents are listed in the table below:</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>Numeric Value</th></tr>
<tr><td>AgentError</td><td>0</td></tr>
<tr><td>Unregistered</td><td>1</td></tr>
<tr><td>Registered</td><td>2</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the registration status of the VDA. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.
</td></tr>
<tr><td>Has valid license?</td><td>Indicates whether the VDA license is valid or not.</td><td></td><td>
The values that this measure can report and its corresponding numeric equivalents are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>No</td><td>0</td></tr>
<tr><td>Yes</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate whether the license is valid or not. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.
</td></tr>
<tr><td>Errors in event log in last one hour</td><td>Indicates the number of errors detected in the event log for the server during the last 1 hour.</td><td>Number</td><td>Ideally, the value of this measure should be 0.</td></tr>
<tr><td>Warnings in event log in last one hour</td><td>Indicates the number of warning messages that were logged in the event log for the server during the last 1 hour.</td><td>Number</td><td>Ideally, the value of this measure should be 0.</td></tr>
</table>

5.2.5 User Profile Management - VM Test

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.

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 - Guest 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.

Note:

This test is relevant only where the Citrix XenDesktop Broker is used to broker connections between the user and the desktops. This is why, 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 the *Microsoft Hyper-V - 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 : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for every user who is connected to 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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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.

Measurements reported 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

Measurement	Description	Measurement Unit	Interpretation
			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 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	

Measurement	Description	Measurement Unit	Interpretation
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 user's profile that are synchronized during logon and categorized by the file size ranging from 1KB to 10KB.	Number	
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	Indicates the number of	Number	All the Processed Logon Files

Measurement	Description	Measurement Unit	Interpretation
Files from 10KB to 100KB	locally copied files for this user's profile that are synchronized during logoff and categorized by the file size ranging from 1KB to 10KB.		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 1MB	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 100KB to 1MB.	Number	
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.

Measurement	Description	Measurement Unit	Interpretation
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	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 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	

5.2.6 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.

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 the *Microsoft Hyper-V - 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.

Note:

This test reports metrics for Windows VMs only.

Target of the test : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results will be reported for every Windows virtual desktop on the monitored Hyper-V server

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. **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, 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 **2.2.2** 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*.
5. **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.

Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.
7. **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.

8. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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.

Measurements reported 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.

5.2.7 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 : A Hyper-V server

Agent executing the test : An internal agent

Output of the test : One set of results for each browser used by the currently logged-in user to a Windows virtual desktop

First-level descriptor: VM name

Second-level descriptor: User name

Third-level descriptor: User name

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **SHOW RECENT WEBSITES DD -** Typically, the detailed diagnosis of the *Recent websites* measure, if enabled, reveals the URL that is open in each browser tab of a virtual desktop. In large VDI environments supporting hundreds of virtual desktops and users, collecting and storing the details of every browser tab that a user opens can increase the strain on the eG database. To avoid this, by default, this test does not collect detailed diagnostics for the *Recent websites* measure. Accordingly, the **SHOW RECENT WEBSITES DD** is set to **No** by

default. You can turn this flag on if you want, by selecting the **Yes** option. If this is done, then this test will collect detailed metrics for the *Recent websites* measure.

7. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

8. **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.

9. **IGNORE VMS INSIDE VIEW -** Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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. **DD FREQUENCY -** Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *Microsoft Hyper-V* 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 a Hyper-V 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**.

13. 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 reported by the test

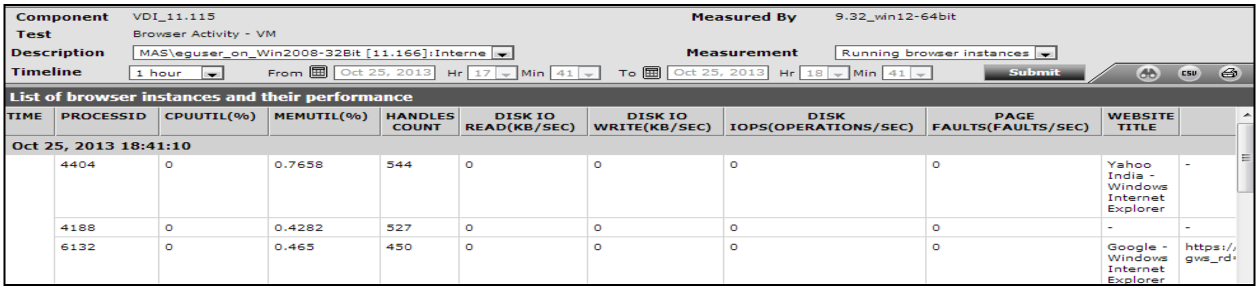
Measurement	Description	Measurement Unit	Interpretation
Running browser instances	Indicates the number of instances of this browser currently running on this virtual desktop.	Number	Use the detailed diagnosis of this measure to know how much resources were utilized by each instance of a browser, so that the

Measurement	Description	Measurement Unit	Interpretation
			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 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.

Measurement	Description	Measurement Unit	Interpretation
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.
Page faults	Indicates the rate at which page faults by the	Faults/Sec	Ideally, the value of this measure should be low. A high value for a

Measurement	Description	Measurement Unit	Interpretation
	threads executing in this browser are occurring on this virtual desktop.		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.



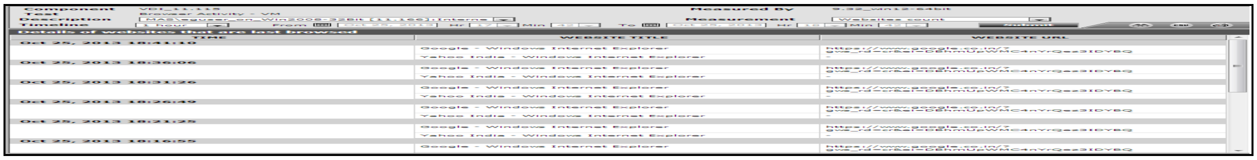
Component VDI_11.115
Test Browser Activity - VM
Description MAS\eguser_on_Win2008-32Bit [11.166]:Interne
Timeline 1 hour From Oct 25, 2013 Hr 17 Min 41 To Oct 25, 2013 Hr 18 Min 41
Measured By 9.32_win12-64bit
Measurement Running browser instances

List of browser instances and their performance

TIME	PROCESSID	CPUUTIL(%)	MEMUTIL(%)	HANDLES COUNT	DISK IO READ(KB/SEC)	DISK IO WRITE(KB/SEC)	DISK IOPS(OPERATIONS/SEC)	PAGE FAULTS(FAULTS/SEC)	WEBSITE TITLE	
Oct 25, 2013 18:41:10										
	4404	0	0.7658	544	0	0	0	0	Yahoo India - Windows Internet Explorer	-
	4188	0	0.4282	527	0	0	0	0	-	-
	6132	0	0.465	450	0	0	0	0	Google - Windows Internet Explorer	https:/ gws_rd=

Figure 5.11: 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.



Component VDI_11.115
Test Browser Activity - VM
Description MAS\eguser_on_Win2008-32Bit [11.166]:Interne
Timeline 1 hour From Oct 25, 2013 Hr 17 Min 42 To Oct 25, 2013 Hr 18 Min 42
Measured By 9.32_win12-64bit
Measurement Websites count

Details of websites that are last browsed

TIME	WEBSITE TITLE	WEBSITE URL
Oct 25, 2013 18:41:10		
	Google - Windows Internet Explorer	https://www.google.co.in/?gws_rd=cr&ei=DBhmUpWMC4nYrQez3IDYBQ
	Yahoo India - Windows Internet Explorer	-
Oct 25, 2013 18:36:06		
	Google - Windows Internet Explorer	https://www.google.co.in/?gws_rd=cr&ei=DBhmUpWMC4nYrQez3IDYBQ
	Yahoo India - Windows Internet Explorer	-
Oct 25, 2013 18:31:26		
	Google - Windows Internet Explorer	https://www.google.co.in/?gws_rd=cr&ei=DBhmUpWMC4nYrQez3IDYBQ
	Yahoo India - Windows Internet Explorer	-
Oct 25, 2013 18:26:49		
	Google - Windows Internet Explorer	https://www.google.co.in/?gws_rd=cr&ei=DBhmUpWMC4nYrQez3IDYBQ
	Yahoo India - Windows Internet Explorer	-
Oct 25, 2013 18:21:25		
	Google - Windows Internet Explorer	https://www.google.co.in/?gws_rd=cr&ei=DBhmUpWMC4nYrQez3IDYBQ
	Yahoo India - Windows Internet Explorer	-
Oct 25, 2013 18:16:55		
	Google - Windows Internet Explorer	https://www.google.co.in/?gws_rd=cr&ei=DBhmUpWMC4nYrQez3IDYBQ

Figure 5.12: The detailed diagnosis of the Recent web sites measure

5.2.8 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 XenServer, 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 : A Hyper-V server hosting virtual desktops

Agent executing the test : An internal agent

Output of the test : One set of results for the user who is currently connected to each virtual desktop on the monitored server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to ‘eG VM Agent (Windows)’**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.

9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *Microsoft Hyper-V* 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 a Hyper-V 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**.

13. 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 reported by the test

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Personal vDisk service status</td><td>Indicates whether Citrix Personal vDisk service is running or not on this VM.</td><td></td><td>The values that this measure can report and their corresponding numeric values have been discussed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Stopped</td><td>0</td></tr>
<tr><td>Running</td><td>1</td></tr>
<tr><td>Not installed</td><td>2</td></tr>
</table>
<b>Note:</b>
By default, this test reports the <b>Measures Value</b>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.</td></tr>
<tr><td>Recompose status</td><td>Indicates the status of the initially provisioned disk or the updated image.</td><td>Number</td><td>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:
• 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</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			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

Measurement	Description	Measurement Unit	Interpretation
			users.
Space allocated for user applications:	Indicates the amount of space allocation for storing user applications on the personal vDisk attached to this VM.	MB	These measures indicate how much space has been allocated to this .vhd file and how much of the allocated space has been utilized by user applications contained in this file. 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:.

Measurement	Description	Measurement Unit	Interpretation
			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 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	ndicates 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 of10 GB is recommended.
Space utilized	Indicates the percentage of space in	Percent	A consistent increase in the value of this measure is a cause for concern,

Measurement	Description	Measurement Unit	Interpretation
	the personal vDisk attached to this VM that is currently used.		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 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.

5.2.9 Virtual Desktop Session Start-up Details Test

Figure 5.13 depicts a typical user logon process to a virtual desktop via XenDesktop broker.

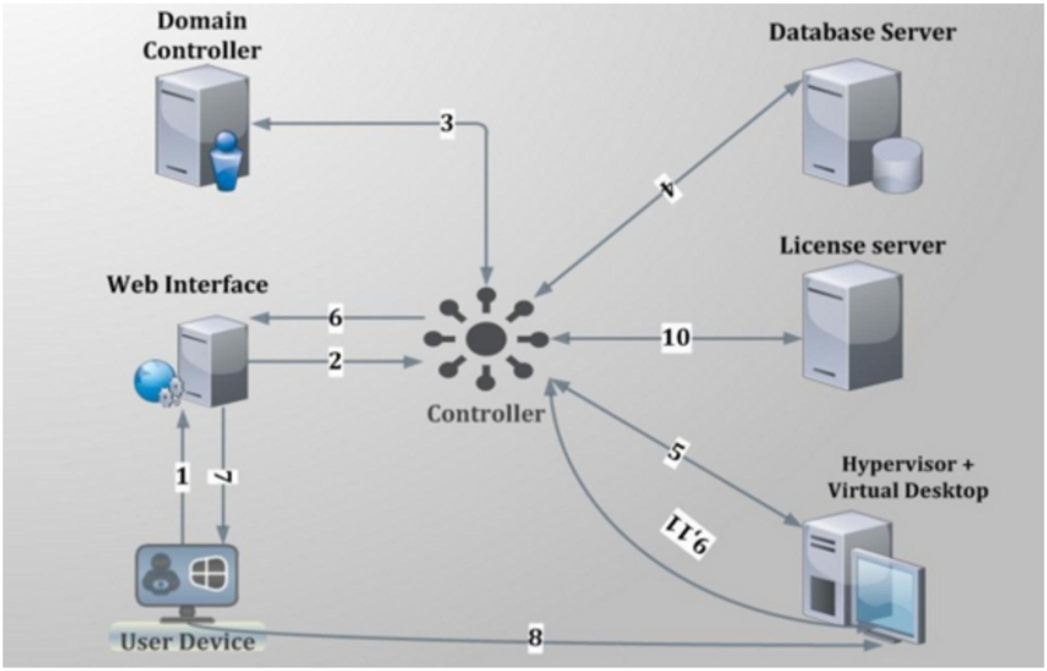


Figure 5.13: Citrix user logon process

The process depicted by Figure 5.13 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 connection-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 : A Hyper-V server hosting virtual desktops

Agent executing the test : An internal agent

Output of the test : One set of results for the user who is currently connected to each virtual desktop on the monitored server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.
   - **If the INSIDE VIEW USING flag is set to ‘eG VM Agent (Windows)’**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **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 *Hyper-V* server. For a *Hyper-V VDI* server on the other hand, the default status of this flag is **No**. This is because, in such environments, the virtual desktops will be in the powered-off state most of the time.

9. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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. **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.

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. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. For a *Microsoft Hyper-V* 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 a Hyper-V 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**.

13. 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.

Measurement	Description	Measurement Unit	Interpretation
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.
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.
Logon script	Indicates the time taken	Secs	If a user complains of slowness, then, you

Measurement	Description	Measurement Unit	Interpretation
execution duration:	for the login script to execute for this user.		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.
Client side session start-up processing 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 Server side session start-up processing duration 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 Client side session start-up processing 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 duration, Configuration file download duration, User credential obtention by client duration, ICA file download duration, Launch page web server duration, Name resolution duration, Name resolution web server duration, Session lookup duration, Session creation at client duration, *Ticket response web server duration*, *Reconnect enumeration duration*, and *Reconnect enumeration web server duration* to know what client-side issue is causing the Client side session start-up processing duration to be high.
Backup URL count:	**This measure is relevant when the Citrix Receiver is the session launch**	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

Measurement	Description	Measurement Unit	Interpretation
	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.		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 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 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	Secs	If the Client side session start-up processing 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 duration, User credential obtention by client duration, ICA file download duration, Launch page web

Measurement	Description	Measurement Unit	Interpretation
	the Web Interface service.		server duration, Name resolution duration, Name resolution web server duration, Session lookup duration, Session creation at client duration, *Ticket response web server duration*, *Reconnect enumeration duration*, and *Reconnect enumeration 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.
Configuration file download 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 Client side session start-up processing 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 duration, User credential obtention by client duration, ICA file download duration, Launch page web server duration, Name resolution duration, Name resolution web server duration, Session lookup duration, Session creation at client duration, *Ticket response web server duration*, *Reconnect enumeration duration*, and *Reconnect enumeration 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.
User credential obtention by client duration:	**This measure is relevant when the Citrix Receiver is the session launch mechanism.** It measures the time required by this user's		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

Measurement	Description	Measurement Unit	Interpretation
	session to obtain the user credentials.		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 time it takes for this user's client to download the ICA file from the web server.	Secs	The overall process here is: 1. The user clicks on application icon. 2. The user's browser requests the Web Interface launch page. 3. 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). 4. The Web Interface generates ICA file data. 5. The Web Interface sends the ICA file data back to the user's browser. 6. 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,

Measurement	Description	Measurement Unit	Interpretation
			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 mechanism.** It measures the time needed by this user's session to process the launch page (launch. aspx) on the Web Interface server.	Secs	If the value of this measure is high, it indicates at a bottleneck 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 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

Measurement	Description	Measurement Unit	Interpretation
			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 mechanism.** It is the time it takes the XML service to resolve the name of this virtual desktop to its IP address.	Secs	When this metric is high, there could be an issue with the Web Interface server or the Citrix Receiver, the XML Service, 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 lookup 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 at client duration:	Indicates the new session creation time.	Secs	In the event of slowness, if the Client side session start-up processing duration of a user session is found to be higher than the Session start-up server duration, you may

Measurement	Description	Measurement Unit	Interpretation
			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.
Reconnect enumeration 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.

Measurement	Description	Measurement Unit	Interpretation
Server side session start-up processing 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, User credential obtention by server duration, Program neighbourhood credentials obtention server duration, Pass-through credentials duration, Credential authentication duration, Profile load server duration, Session creation processing duration, Endpoint resources mapping duration, Endpoint printers mapping 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 server duration'.
User credential obtention by 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.
Pass-through	Indicates the time spent	Secs	This only applies to a Security Support

Measurement	Description	Measurement Unit	Interpretation
credentials duration:	by this virtual desktop performing network operations to obtain credentials for this user.		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 User credential obtention by 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 Server side session start-up processing duration duration.
Credential authentication 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, User credential obtention by server duration, Program neighbourhood credentials obtention server duration, Pass-through credentials duration, Profile load server duration, Session creation processing duration, Endpoint resources mapping duration, Endpoint printers mapping 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

Measurement	Description	Measurement Unit	Interpretation
			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 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.
Session creation processing 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.
Endpoint resources mapping 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.
Endpoint printers mapping 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

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>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 created - especially if users have a lot of local printers.</td></tr>
<tr><td>Profile provider</td><td>Indicates the provider who handles this user's profile.</td><td></td><td>The values reported by this measure and their corresponding numeric equivalents are described in the table below:
<table>
<tr><th>Measure Values</th><th>Numeric Values</th></tr>
<tr><td>Citrix Profile management</td><td>0</td></tr>
<tr><td>Microsoft Roaming profile</td><td>1</td></tr>
<tr><td>Others</td><td>2</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the above-mentioned <b>Measure Value</b>s while indicating the provider who handles this user's profile. However, in the graph of this measure, the values will be represented using the corresponding numeric equivalents i.e., 0 to 2.</td></tr>
<tr><td>Profile type:</td><td>Indicates the type of this user's profile.</td><td></td><td>The values reported by this measure and their corresponding numeric equivalents are described in the table below:</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 Values</th><th>Numeric Values</th></tr><tr><td>Managed profile</td><td>0</td></tr><tr><td>Temporary profile</td><td>1</td></tr><tr><td>Mandatory profile</td><td>2</td></tr><tr><td>Roaming profile</td><td>3</td></tr><tr><td>Unknown type</td><td>4</td></tr></table>
<b>Note:</b>
By default, this measure reports the above-mentioned <b>Measure Value</b>s while indicating the profile type of this users. However, in the graph of this measure, the values will be represented using the corresponding numeric equivalents i.e., 0 to 4.</td></tr>
<tr><td>Group Policy processing status:</td><td>Indicates the current status of the Group policy that is applied for this user.</td><td></td><td>The values reported by this measure and their corresponding numeric equivalents are described in the table below:
<table><tr><th>Measure Values</th><th>Numeric Values</th></tr><tr><td>Success</td><td>1</td></tr><tr><td>Warning</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr></table>
<b>Note:</b>
By default, this measure reports the above-mentioned <b>Measure Value</b>s while indicating the current status of the Group policy. However, in the graph of this measure, the values will be represented using the corresponding numeric equivalents i.e., 1 to 3.</td></tr>
<tr><td>User account discovery:</td><td>Indicates the amount of time taken by the LDAP</td><td>Secs</td><td>Compare the value of this measure across users to know which user’s logon process</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
	call for this user to connect and bind to Active Directory during the last measurement period.		spent maximum time in retrieving account information. To know which domain controller and DNS is being used, use the detailed diagnosis of this measure.
LDAP bind time to active directory:	Indicates the amount of time taken by the LDAP call for this user to connect and bind to Active Directory during the last measurement period.	Secs	Compare the value of this measure across users to know which user's logon process spent maximum time in connecting to Active Directory. Besides impacting authentication time, high LDAP bind time may also affect group policy processing.
DC discovery time	Indicates the time taken to discover the domain controller to be used for processing group policies for this user during the last measurement period.	Secs	Compare the value of this measure across users to know which user's logon process spent maximum time in domain controller discovery.
Total Group Policy Object file access time:	Indicates the amount of time the logon process took to access group policy object files for this user during the last measurement period.	Secs	Compare the value of this measure across users to know which user's logon process spent maximum time in accessing the group policy object file. To know which files were accessed and the time taken to access each file, use the detailed diagnosis of this measure. With the help of the detailed diagnostics, you can accurately isolate the object file that took the longest to access, and thus delayed the logon process.
Total Client-side extensions applied:	Indicates the total number of client side extensions used for processing group policies for this user during the last measurement period.	Number	

Measurement	Description	Measurement Unit	Interpretation
Client-side extensions with success state:	Indicates the number of client side extensions that were successfully used for processing group policies for this user during the last measurement period.	Number	Use the detailed diagnosis of this measure to know which were the successful client side extensions for a user, and which group policy was processed by each extension.
Client-side extensions with warning state:	Indicates the number of warnings received when client side extensions were used for processing group policies for this user during the last measurement period.	Number	Use the detailed diagnosis of this measure to know which were the client side extensions that resulted in the generation of warning events at the time of processing. You will also know which group policies were processed by each extension.
Client-side extensions with error state:	Indicates the number of errors registered when client side extensions were used for processing group policies for this user during the last measurement period.	Number	Ideally, the value of this measure should be zero. A sudden/gradual increase in the value of this measure is a cause of concern. If a non-zero value is reported for this measure, then use the detailed diagnosis of this measure to know which client side extensions resulted in processing errors. You will also know which group policies were processed by each such extension. Moreover, the error code will also be displayed as part of detailed diagnostics, so that you can figure out what type of error occurred when processing the client side extensions.
Total Client-side extension processed time:	Indicates the amount of time that client side extensions took for processing group policies for this user during the last measurement period.	Secs	Compare the value of this measure across users to know which user's logon process spent maximum time in client side extension processing. If this measure reports an unusually high value for any user, then, you may want to check the value of the LDAP bind time to

Measurement	Description	Measurement Unit	Interpretation
			active directory measure for that user to figure out if a delay in connecting to AD is affecting group policy processing. This is because, group policies are built on top of AD, and hence rely on the directory service's infrastructure for their operation. As a consequence, DNS and AD issues may affect Group Policies severely. One could say that if an AD issue does not interfere with authentication, at the very least it will hamper group policy processing. You can also use the detailed diagnosis of this measure to know which client side extension was used to process which group policy for a particular user. Detailed diagnostics also reveal the processing time for each client side extension. This way, you can quickly identify the client side extension that took too long to be processed and thus delayed the user logon.
Estimated network bandwidth between VM and Domain Controller:	Indicates the estimated network bandwidth between the VM and domain controller for this user during the last measurement period.	Kbps	
Is link between VM and Domain Controller slow?:	Indicates whether/not the network connection between the VM and domain controller is currently slow for this user.		Several components of Group Policy rely on a fast network connection. If a fast connection is unavailable between a VM and the DOC, group policy processing can be delayed. This is why, if the *Group Policy processing duration* measure reports an abnormally high value, you may want to check the value of the *Is link between VM and domain controller slow?* measure to determine whether the network connection between the VM and domain

<table>
<thead>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
</thead>
<tbody>
<tr><td></td><td></td><td></td><td>
<p>controller is slow.</p>
<p>If the network connection between the VM and domain controller is slow for a user, then this measure will report the value <i>Yes</i>. If it is fast, then this measure will report the value <i>No (connection is fast)</i>.</p>
<p>The numeric values that correspond to the above-mentioned measure values are as follows:</p>
<table>
<thead><tr><th>Measure Value</th><th>Numeric Value</th></tr></thead>
<tbody>
<tr><td>Yes</td><td>1</td></tr>
<tr><td>No (connection is fast)</td><td>2</td></tr>
</tbody>
</table>
<p><b>Note:</b></p>
<ul>
<li>By default, this test reports the <b>Measure Value</b>s listed in the table above to indicate the quality of the network link between the VM and the domain controller. In the graph of this measure however, the same is indicated using the numeric equivalents only.</li>
<li>To determine whether the network link is slow or fast, the Group Policy service compares the result of the estimated bandwidth to the slow link threshold (configured by Group Policy). A value below the threshold results in the Group Policy service flagging the network connection as a slow link. This measure reports the status of this flag only. To know the slow link threshold that the</li>
</ul>
</td></tr>
</tbody>
</table>

Measurement	Description	Measurement Unit	Interpretation
			Group Policy has configured for this link, use the detailed diagnosis of this measure.
Is the user's profile size large?:	Indicates whether the profile size of this user exceeds the default profile quota size of 100MB.	Boolean	If this measure shows 0, it indicates that the current profile size has not exceeded the quota size. The value 1 indicates that the current profile size has exceeded the quota size.
Current profile size:	Indicates the current profile size of this user.	MB	
Number of files in user’s profile:	Indicates the number of files available in this user profile.	Number	
Large files in user’s profile:	The number of files in this user profile, which exceed the default file size limit of 100 MB.	Number	The detailed diagnosis of this measure, if enabled, lists all the files that have exceeded the default file size limit of 100 MB.
Group Policy applied on:	Indicates whether the group policy for this user is applied during foreground processing or background processing.		Foreground and background processing are key concepts in Group Policy. Foreground processing only occurs when the machine starts up or when the user logs on. Some policy areas (also called Client Side Extensions (CSEs)) can only run during foreground processing. Examples of these include Folder Redirection, Software Installation and Group Policy Preferences Drive Mapping. In contrast, background processing is that thing that occurs every 90 or so minutes on Windows workstations, where GP refreshes itself periodically. Background processing happens in the background, while the user is working and they generally never notice it. While background processing does not impact performance, foreground processing can extend start and login times.

<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 values that this measure can report and their corresponding numeric values are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Background</td><td>1</td></tr>
<tr><td>Foreground</td><td>2</td></tr>
</table>
<b>Note:</b>
By default, this test reports the <b>Measure Value</b>s listed in the table above to indicate when the group policy of a user was applied. In the graph of this measure however, the same is indicated using the numeric equivalents only.</td></tr>
<tr><td>Group Policy processing mode:</td><td>Indicates whether the group policies of this user are processed in the synchronous or asynchronous mode.</td><td></td><td>Foreground processing can operate under two different modes - synchronously or asynchronously. Asynchronous GP processing does not prevent the user from using their desktop while GP processing completes. For example, when the computer is starting up, GP asynchronous processing starts to occur for the computer, and in the meantime, the user is presented the Windows logon prompt. Likewise, for asynchronous user processing, the user logs on and is presented with their desktop while GP finishes processing. The user is not delayed getting either their logon prompt or their desktop during asynchronous GP processing. When foreground processing is synchronous, the user is not presented with the logon prompt until computer GP processing has completed after a system boot. Likewise the user will not see their desktop at logon until user GP processing completes. This can have the effect of</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			making the user feel like the system is running slow. In short, synchronous processing can impact startup time, where asynchronous does not. Foreground processing will run synchronously for two reasons: • The administrator forces synchronous processing through a policy setting. This can be done by enabling the Computer ConfigurationPoliciesAdministrative TemplatesSystemLogonAlways wait for the network at computer startup and logon policy setting. Enabling this setting will make all foreground processing synchronous. This is commonly used for troubleshooting problems with Group Policy processing, but does not always get turned back off again. • A particular CSE requires synchronous foreground processing. There are four CSEs provided by Microsoft that currently require synchronous foreground processing: Software Installation, Folder Redirection, Microsoft Disk Quota and GP Preferences Drive Mapping. If any of these are enabled within one or more GPOs, they will trigger the next foreground processing cycle to run synchronously when they are changed. It is therefore best to avoid synchronous CSEs and to not force synchronous policy. If usage of synchronous CSEs is necessary, minimize changes to these

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>policy settings.
The values that this measure can report and their corresponding numeric values are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Synchronous</td><td>1</td></tr>
<tr><td>Asynchronous</td><td>2</td></tr>
</table>
<b>Note:</b>
By default, this test reports the <b>Measure Value</b>s listed in the table above to indicate when the group policy of a user was applied. In the graph of this measure however, the same is indicated using the numeric equivalents only.</td></tr>
</table>

5.2.10 Virtual Desktop Sessions Details Test

A user logged into a virtual desktop does not imply active usage of 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. Idle users themselves are unproductive resources. Besides, since these 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 : A Microsoft Hyper-V server

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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **IDLE TIME** - Specify the time duration (in minutes) of inactivity beyond which a session is considered to be “idle” by this test. By default, this parameter is set to 30 (minutes). This implies that by default, the test counts all sessions that have been inactive for over 30 minutes as idle sessions.

Measurements reported 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.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Time since last activity:</td><td>Indicates the time that has elapsed since this user performed an action on this desktop.</td><td>Mins</td><td>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.</td></tr>
<tr><td>Is session idle in long time?</td><td>Indicates whether/not the session has been idle beyond the time duration specified against the <b>IDLE TIME</b> parameter.</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>No</td><td>0</td></tr><tr><td>Yes</td><td>1</td></tr></table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above. In the graph of this measure however, the value of this measure is represented using their numeric equivalents only.</td></tr>
<tr><td>Total idle time in session:</td><td>Indicates the total time for which this user was idle during the session.</td><td>Mins</td><td>If the value of this measure is the same as the value of the <i>Total time in session</i> 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 <i>Total time in session</i> measure for a user, it implies that the user has been idle for a long time.
If the value of this measure is much</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			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.

5.2.11 Page File - VM Test

When the load imposed by applications and services running on a server nears the amount of installed RAM, additional storage is necessary. The page file serves as the temporary store on disk for memory that cannot be accommodated in the physical RAM. Since it is frequently accessed for storing and retrieving data that is needed for virtual memory access by application, the location and sizing of the page files can have a critical impact on server's performance. Ideally, the server operating system and the page file should be available on different drives for optimal performance. Splitting the page file across different drives can improve performance further.

A rule of thumb in sizing the page file is to set the maximum size of the page file to 1.5 times the available RAM. While this works well for systems with smaller physical memory, for other systems, the optimal page file size has to be determined based on experience using the system and studying the typical workload.

This test tracks the usage of each of the page files on a Windows VM. Note that this test is available for VMs running on Windows servers 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 *Hyper-V / Hyper-V VDI* as the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Hyper-V / Hyper-V VDI server

Agent executing the test : An internal agent

Output of the test : One set of results for every page file on a Windows server

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. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.

- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **REPORTTOTAL** - Set this flag to **Yes** if you want the test to report total page file usage - i.e., the aggregate usage across multiple page files. In this case therefore, a **Total** descriptor will newly appear for this test in the eG monitoring console.
12. **REPORTTOTALONLY** - If both the **REPORTTOTAL** and **REPORTTOTALONLY** flags are set to **Yes**, then the test will report only the aggregate usage across multiple page files - in other words, the test will report values for the **Total** descriptor only. Likewise, if the **REPORTTOTAL** flag is set to **No**, and the **REPORTTOTALONLY** flag is set to **Yes**, then again, the test will report current usage for the **Total** descriptor only. However, if both the **REPORTTOTAL** and **REPORTTOTALONLY** flags are set to **No**, then the test will report individual usages only. Also, if the **REPORTTOTAL** flag is set to **Yes** and the **REPORTTOTALONLY** flag is set to **No**, then both the individual and Total usages will be reported.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current usage	Indicates the current usage of a page file.	Percent	This metric should be less than 90%. If the page file does not have additional space, additional users/processes cannot be supported and system performance will suffer. To improve

Measurement	Description	Measurement Unit	Interpretation
			performance, consider resizing the page file. Microsoft Windows allows a minimum and maximum size of the page file to be specified. If the system has sufficient disk space, consider setting the page file to start out at the maximum size (by using the same value for the minimum and maximum sizes), so that system resources are not spent growing the page file size when there is a virtual memory shortage.

5.2.12 User Logon - VM Test

The process of a user logging into a virtual server is fairly complex. First, the domain controller is discovered and the login credentials are authenticated. Then, the corresponding user profile is identified and loaded. Next, group policies are applied and logon scripts are processed to setup the user environment. In the meantime, additional processing may take place for a user – say, applying system profiles, creating new printers for the user, and so on. A slowdown in any of these steps can significantly delay the logon process for a user. Since logons on Windows happen sequentially, this may adversely impact the logins for other users who may be trying to access the virtual server at the same time. Hence, if a user complains that he/she is unable to access an application/desktop published on virtual server, administrators must be able to rapidly isolate exactly where the logon process is stalling and for which user. The typical process for monitoring and troubleshooting the login process on Windows is to use the user environment debugging mechanism. To enable this on Windows and to set the logging level associated with the userenv.log file, perform the following steps:

- Start a registry editor (e.g., regedit.exe).
- Navigate to the **HKEY_ LOCAL_ MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon** registry subkey.
- From the Edit menu, select New, DWORD Value.
- Enter the name UserEnvDebugLevel, then press Enter.

- Double-click the new value, set it to 65538 (decimal) - which corresponds to the debugger output.

Once these changes are enabled, details about the Windows login process are logged into the file *%systemroot%\debug\usermode\userenv.log* . The log file is written to the *%Systemroot%\Debug\UserMode\Userenv.log* file. If the Userenv.log file is larger than 300 KB, the file is renamed *Userenv.bak*, and a new *Userenv.log* file is created. This action occurs when a user logs on locally or by using Terminal Services, and the Winlogon process starts. However, because the size check only occurs when a user logs on, the Userenv.log file may grow beyond the 300 KB limit. The 300 KB limit cannot be modified.

The **User Logon - VM** test periodically checks the userenv log file on Windows to monitor the user login and profile loading process and accurately identify where the process is bottlenecked. On Windows 2008 (or above), this test takes the help of the Windows event logs to capture anomalies in the user login and profile loading process and report where the process is bottlenecked - in the authentication process? during profile loading? during GPO processing and if so, which GPO?

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 the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Microsoft Hyper-V server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every user to the Microsoft Hyper-V 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** – Refers to the port used by the host.
4. **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, 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 **2.2.2** 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*.

5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 encircled '+' icon that appears alongside the . To know how to use the special page, refer to Configuring Users for VM Monitoring of this document.

   - **If the INSIDE VIEW USING flag is set to ‘eG VM Agent (Windows)’**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.

7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. **REPORT FOR EACH USER –** By default, this flag is set to **Yes**. This implies that, by default, the test will report metrics for each user to the virtual machine. If you set this flag to **No**, then metrics will be reported for VMs.

12. **REPORT BY DOMAIN NAME –** By default, this flag is set to **No**. This means that, by default, the test will report metrics for each username only. You can set this flag to **Yes**, to ensure that the test reports metrics for each domainname\username.

13. **REPORT UNKNOWN –** By default, this flag is set to **No**. Accordingly, the test, by default, disregards user sessions that have remained active on the server for a duration lesser than the **TEST PERIOD**. If you want the test to report metrics for such users as well, then set this flag to **Yes**. In this case, the test will additionally support an Unknown descriptor – the metrics reported by this descriptor will be aggregated across all such user sessions that have been active on the server only for a limited duration.

14. **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**.

15. **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
Logon duration:	Indicates the average time taken by this user for logging in during the last measurement period.	Msecs	If this value is abnormally high for any user, then, you can compare the User account discovery time, LDAP bind time to Active Directory, Client side extension processed time, DC discovery time, Total group policy object file access time, Avg system policy processing time and User profile load time measures to know exactly where that user's login process experienced a bottleneck - is it when loading the profile? is it when processing system policies? is it when processing group policies? is it when interacting with AD for authenticating the user login?
User account discovery:	Indicates the amount of time taken by the system call to get account information for this user during the last measurement period.	Msecs	Compare the value of this measure across users to know which user's logon process spent maximum time in retrieving account information.
LDAP bind time to Active Directory:	Indicates the amount of time taken by the LDAP call for this user to connect and bind to Active Directory during the last measurement period.	MSecs	Compare the value of this measure across users to know which user's logon process spent maximum time in connecting to Active Directory. Besides impacting authentication time, high LDAP bind time may also affect group policy processing.
Client side extension processed time:	Indicates the amount of time that client side extensions took for	MSecs	Compare the value of this measure across users to know which user's logon process spent maximum time

Measurement	Description	Measurement Unit	Interpretation
	processing group policies for this user during the last measurement period.		in group policy processing. If this measure reports an unusually high value for any user, then, you may want to check the value of the LDAP bind time to Active Directory measure for that user to figure out if a delay in connecting to AD is affecting group policy processing. This is because, group policies are built on top of AD, and hence rely on the directory service's infrastructure for their operation. As a consequence, DNS and AD issues may affect Group Policies severely. One could say that if an AD issue does not interfere with authentication, at the very least it will hamper group policy processing. You can also use the detailed diagnosis of this measure to know which client side extension was used to process which group policy for a particular user.
DC discovery time:	Indicates the time taken to discover the domain controller to be used for processing group policies for this user during the last measurement period.	MSecs	Compare the value of this measure across users to know which user's logon process spent maximum time in domain controller discovery.
Total group policy object file accessed time:	Indicates the amount of time the logon process took to access group	MSecs	Compare the value of this measure across users to know which user's logon process spent maximum time

Measurement	Description	Measurement Unit	Interpretation
	policy object files for this user during the last measurement period.		in accessing the group policy object file.
User profile load time:	Indicates the amount of time it took to load this user's profile successfully in the last measurement period.	MSecs	Compare the value of this measure across users to know which user's profile took the longest time to load. One of the common reasons for long profile load times is large profile size. In such circumstances, you can use the User Profile test to determine the current size of this user's profile. If the profile size is found to be large, you can conclude that it is indeed the size of the profile which is affecting the profile load time. Another reason would be the absence of a profile. If the user does not already have a profile a new one is created. This slows down the initial logon quite a bit compared to subsequent logons. The main reason is that Active Setup runs the IE/Mail/Theme initialization routines. Moreover, this measure reports the average time taken for loading a user's profile across all the sessions of that user. To know the profile load time per user session, use the detailed diagnosis of this measure. This will accurately pinpoint the session in which the profile took the longest to load.

Measurement	Description	Measurement Unit	Interpretation
Group policy starts:	Indicates the number of group policy applications started for this user in the last measurement period.	Number	Logon performance improves when fewer Group Policies are applied. Merge GPOs when possible instead of having multiple GPOs.
Group policy completes:	Indicates the number of group policy applications completed for this user in the last measurement period.	Number	
Client side extensions applied:	Indicates the number of client side extensions used for processing group policies for this user during the last measurement period.	Number	
Max group policy time:	Indicates the maximum time taken for applying group policies for this user in the last measurement period.	Msecs	
Profile load starts:	Indicates the number of profile loads started for this user in the last measurement period.	Number	Use the detailed diagnosis of this measure to know the details of the user sessions in which profile loads were started.
Profile load successes:	Indicates the number of successful profile loads for this user in the last measurement period.	Number	
Profile loading failures:	Indicates the number of profile load failures for this user in the last measurement period.	Number	An unusual increase in number of profile loading failures is a cause for concern. The userenv.log/event logs file will have details of what profile loads failed and why.

Measurement	Description	Measurement Unit	Interpretation
Profile load failures percent:	Indicates the percentage of profile loads that failed for this user in the last measurement period.	Percent	A low value is desired for this measure. Compare the value of this measure across users to know which user's profile failed to load most often.
Avg user profile load time:	Indicates the average time it took to load this user's profile successfully in the last measurement period.	Msecs	Ideally, profile load time should be low for any user. A high value or a consistent rise in this value is a cause for concern, as it indicates a delay in profile loading. This in turn will have a negative impact on user experience. One of the common reasons for long profile load times is large profile size. Compare the value of this measure across users to identify that user whose profile took the longest to load. Then, use the User Profile test to determine the current size of this user's profile. If the profile size is found to be large, you can conclude that it is indeed the size of the profile which is affecting the profile load time.
Max profile load time:	Indicates the maximum time it took to load a profile during the last measurement period.	Msecs	
Profile unload starts:	Indicates the number of profile unloads started for this user during the last measurement period.	Number	Use the detailed diagnosis of this measure to know when a user's session was initiated and how long each session remained active on the Hyper-V server. From this, you can infer how many sessions were

Measurement	Description	Measurement Unit	Interpretation
			active for a user on the server and the duration of each session, and thus identify long-running sessions for the user.
Profile unload successes:	Indicates the number of successful profile unloads for this user during the last measurement period.	Number	
Profile unload failures:	Indicates the number of unsuccessful profile unloads during the last measurement period.	Number	
Profile unload failures percent:	Indicates the profile unload failures as a percentage of the total profile unloads.	Percent	
Avg user profile unload time:	Indicates the average time for unloading a profile during the last measurement period.	Msecs	
Max profile unload time:	Indicates the maximum time for unloading a profile during the last measurement period.	Msecs	
System policy starts:	Indicates the number of system policy processes that were started for this user in the last measurement period.	Number	
System policy completes:	Indicates the number of system policy completions for this user	Number	Compare the total number of starts to completions. if there is a significant discrepancy, this

Measurement	Description	Measurement Unit	Interpretation
	in the last measurement period.		denotes a bottleneck in system policy application. Check the userenv.log file for more details.
Avg system policy processing time:	Indicates the average time taken for applying system policies in the last measurement period for this user.	Msecs	If the system policy times are long, check the detailed diagnosis to view if the policy handling is taking time for all users. Analyze the userenv.log to determine the reason for any slowdown.
Max system policy time:	Indicates the maximum time for applying system policies for this user in the last measurement period.	Msecs	

5.2.13 Outlook Add-ins - VM Test

Outlook add-ins are integrations built by third parties into Microsoft Outlook using the new web technologies based platform. Microsoft Outlook add-ins have three key aspects:

- The same add-in and business logic works across desktop Microsoft Outlook for Windows and Mac, web (Office 365 and Outlook.com), and mobile.
- Outlook add-ins consist of a manifest, which describes how the add-in integrates into Outlook (for example, a button or a task pane), and JavaScript/HTML code, which makes up the UI and business logic of the add-in.
- Outlook add-ins can be acquired from the Office store or side-loaded by end-users or administrators.

The Outlook add-ins may be useful in connecting the business and social networks of the users. These add-ins when integrated with Microsoft Outlook simplifies the job of the users as they can stay up to date on the status and activities of their contacts by merely overlooking the Microsoft Outlook! When a user complains that it is taking too long to launch the add-ins of the Microsoft Outlook published on virtual desktops, administrators must be able to quickly identify the add-ins that were loaded while the Microsoft Outlook is opened by the user, know how much time each add-in took to load, and thus pinpoint the add-in that is the slowest in loading. The **Outlook Add-ins - VM** test provides these valuable insights to the administrators. This test auto-discovers all the add-ins integrated with the Microsoft Outlook published on the virtual desktops hosted by the virtual server, and for each discovered add-in, reports the number of times the add-in was loaded and the average and maximum time that add-in took to load. This way, the test points administrators to add-ins that are slow in loading.

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 the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Microsoft Hyper-V server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every outlook add-in integrated with the Microsoft Outlook published on the virtual desktops on the Microsoft Hyper-V 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 **HOST** listens. By default, this is NULL.
4. **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, 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 **2.2.2** 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*.
5. **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. Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 PASSWORD**s 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Hyper-V VDI* model, 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*.
7. **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.

8. **IGNORE VMS INSIDE VIEW** - Administrators of some high security Hyper-V 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 a Hyper-V 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.

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 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. 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
Number of times loaded in last measure period	Indicates the number of times this outlook add-in was loaded during the last measurement period.	Number	The detailed diagnosis of this measure lists the time and duration for which the outlook add-in was loaded. Compare the value of this measure across the add-ins to figure out the most/least popular add-in.
Average load time	Indicates the average time taken by this outlook add-in to load.	Secs	
Maximum load time	Indicates the maximum time taken by this outlook add-in to load.	Secs	Compare the value of this measure across the add-ins to figure out the add-in that is the slowest to load.

5.2.14 Windows Security Center Status - VM Test

Windows Security Center (WSC) is a comprehensive reporting tool that helps administrators establish and maintain a protective security layer around Windows VMs to monitor the VM's health state. The Windows Security Center also monitors third party security products such as firewall, antivirus, antimalware and antispyware, installed on the VM. In order for the security products to be compliant with Windows and successfully report status to Action Center, these products should be registered with the security center. The security products communicate any subsequent status changes to the security center using private APIs. The security center, in turn, communicates these updates to Action Center, where they are finally displayed to the end user. With Windows Security Center, administrators can check whether any security product is installed and turned on, and if the definitions of the products are up to date and real-time protection is enabled. By continuously monitoring the Windows Security Center, administrators can instantly find out whether the security products are up-to-date or out dated, and the status of security products in real-time. This is what exactly the **Windows Security Center Status - VM** test does!

This test auto-discovers the security products installed on the Windsows VMs on the target host, and for each security product reports the current definition status and the current protection status. Using these details, administrators are alerted to the systems on which the automatic updates are outdated and virus protection turned off. By closely monitoring the status, administrators can take necessary actions before the end users become vulnerable to virus threats or malicious attacks.

Target of the test: A Hyper-V / Hyper-V VDI server

Agent executing the test: An internal agent

Output of the test: One set of results for every *security product:provider combination* on each Windows VMs on the target server.

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. **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, 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 **2.2.2** 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*.
5. **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.

Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.
7. **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.

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** 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.

9. **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.

10. **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.

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

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Signature status</td><td>Indicates the current status of this security product.</td><td></td><td>The values reported by this measure and its numeric equivalents are mentioned in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Unknown</td><td>25</td></tr>
<tr><td>Up to date</td><td>15</td></tr>
<tr><td>Out of date</td><td>10</td></tr>
</table>

<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current state of this security product. The graph of this measure however, represents the status of a server using the numeric equivalents only.
Use the detailed diagnosis of this measure, to know about the name of Windows system on which the product is running, the file paths of product executables and the current status of the product.</td></tr>
<tr><td>Real-time protection status</td><td>Indicates the real-time protection status of this security product.</td><td></td><td>The values reported by this measure and its numeric equivalents are mentioned in the table below:</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>Numeric Value</th></tr>
<tr><td>Unknown</td><td>25</td></tr>
<tr><td>Snoozed</td><td>20</td></tr>
<tr><td>On</td><td>15</td></tr>
<tr><td>Expired</td><td>10</td></tr>
<tr><td>Off</td><td>0</td></tr>
</table>
<b>Note:</b>
By default, this measure reports the <b>Measure Value</b>s listed in the table above to indicate the current protection status of this security product. The graph of this measure however, represents the status of a server using the numeric equivalents only.
</td></tr>
</table>

5.2.15 Windows Update Details - VM Test

Microsoft regularly releases various Windows updates to enhance and protect the Windows operating system. These updates are also applicable for the Windows virtual desktops on the VMs. The Windows updates fix newly discovered security holes and bugs, add malware definitions to Windows Defender and Security Essentials utilities, strengthen Office security and add new features/enhancements to the Windows operating system. By installing these updates regularly, you can keep the operating system highly secure, reliable and stable, and can maintain the performance of the operating system at peak. If the operating system is not updated regularly, the critical bugs and security errors may increase vulnerabilities. These vulnerabilities can be exploited by the malware or hackers, thus exposing the operating system to malicious attacks and degrading the operating system's performance. To avoid such eventualities, you should regularly check whether the Windows operating system is up-to-date or not. This check can be easily done using the **Windows Update Details - VM** test.

This test continuously monitors the Windows operating system and reports the current status of the Windows updates for the operating system. Besides, this test indicates whether any update is pending for the operating system and whether the Windows system is rebooted or not. In the process, this test also reports the total number of updates to be installed for the virtual desktop and the number of Windows updates of different types at regular intervals.

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 *Microsoft Windows* as the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test: A Hyper-V VDI server

Agent executing the test: An internal agent

Output of the test: One set of results for every Windows virtual desktop on the target server.

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. **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, 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 **2.2.2** 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*.
5. **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.

Accordingly, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)** by default. To obtain a remote connection, the test must be configured with the privileges of an administrative user to the domain within which the guests reside. 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 VMs 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 Configuring Users for VM Monitoring of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'**: On the other hand, if the **INSIDE VIEW USING** flag is set to **eG VM Agent (Windows)**, then it implies that the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

6. **REPORT BY USER** – For the *Microsoft Hyper-V* monitoring model, the **REPORT BY USER** flag is set to **NO** by default, indicating that by default, the guest operating systems on the Hyper-V server are identified using the hostname specified in the operating system. On the other hand, for the *Microsoft Hyper-V VDI* model, 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*.
7. **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.

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** 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.

9. **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.

10. **DD FOR TOTAL UPDATES** – In large VDI environments where hundreds of Windows virtual desktops have been provisioned, the frequent collection of detailed diagnosis information related to the update details of the virtual desktops may increase the processing overheads of the eG agent, and may even choke the eG database. To avoid this, by default, the **DD FOR TOTAL UPDATES** flag is set to **No** indicating that this test will not report the detailed diagnostics for the *Total Updates Available* measure. However, you can set this flag to **Yes** if you want to collect the detailed diagnostics of the *Total Updates Available* measure.

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 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

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Are pending updates available?</td><td>Indicates whether/not the updates are pending.</td><td></td><td>The values that this measure can report and the numeric values they indicate have been listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>No</td><td>0</td></tr>
<tr><td>Yes</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure can report the <b>Measure Value</b>s mentioned above while indicating whether/not the updates are available. However, the graph of this measure is indicated using the numeric equivalents.</td></tr>
<tr><td>Is a system reboot pending?</td><td>Indicates whether the Windows virtual desktop is rebooted or not.</td><td></td><td>The values that this measure can report and the numeric values they indicate have been listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>No</td><td>0</td></tr>
<tr><td>Yes</td><td>1</td></tr>
</table>
<b>Note:</b>
By default, this measure can report the <b>Measure Value</b>s mentioned above</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>while indicating whether the system is rebooted or not. However, the graph of this measure is indicated using the numeric equivalents.</td></tr>
<tr><td>Windows update service status</td><td>Indicates the current status of the Windows update service.</td><td></td><td>The values that this measure can report and the numeric values they indicate have been listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Unknown</td><td>0</td></tr>
<tr><td>Running</td><td>1</td></tr>
<tr><td>Start pending</td><td>2</td></tr>
<tr><td>Continue pending</td><td>3</td></tr>
<tr><td>Pause pending</td><td>4</td></tr>
<tr><td>Stop pending</td><td>5</td></tr>
<tr><td>Paused</td><td>6</td></tr>
<tr><td>Stopped</td><td>7</td></tr>
</table>
<b>Note:</b>
By default, this measure can report the <b>Measure Value</b>s mentioned above while indicating the current status of Windows update service. However, the graph of this measure is indicated using the numeric equivalents.</td></tr>
<tr><td>Total updates available</td><td>Indicates the total number of Windows updates available for the virtual desktop.</td><td>Number</td><td>The detailed diagnosis of this measure, if enabled, lists the Windows updates available for the system and the categories of the available updates.</td></tr>
<tr><td>Critical updates available</td><td>Indicates the number of critical updates available for the virtual desktop.</td><td>Number</td><td>A critical update is a widely and frequently released update that deals with the specific, non-security related, critical bugs. If these bugs are not fixed quickly, they can cause serious</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			performance degradation, interoperability malfunction or disturb application compatibility.
Important updates available	Indicates the number of important updates available for the virtual desktop.	Number	The important updates help fixing the vulnerabilities using which malware/hackers can exploit the system resources or steal data. This in tun may leave the confidentiality and integrity of the system defenseless and make the user data unavailable.
Moderate updates available	Indicates the number of moderate security updates available for the virtual desktop.	Number	The moderate updates fix a vulnerability whose exploitation can be mitigated to a significant degree by default configuration, auditing, or difficulty of exploitation.
Low updates available	Indicates the number of low security updates available for the virtual desktop.	Number	These updates fix the vulnerability whose exploitation is extremely difficult.
Optional updates available	Indicates the number of optional updates available for the virtual desktop.	Number	An optional update includes Feature Pack and standard Updates, and does not have a severity rating.

About eG Innovations

eG Innovations provides intelligent performance management solutions that automate and dramatically accelerate the discovery, diagnosis, and resolution of IT performance issues in on-premises, cloud and hybrid environments. Where traditional monitoring tools often fail to provide insight into the performance drivers of business services and user experience, eG Innovations provides total performance visibility across every layer and every tier of the IT infrastructure that supports the business service chain. From desktops to applications, from servers to network and storage, from virtualization to cloud, eG Innovations helps companies proactively discover, instantly diagnose, and rapidly resolve even the most challenging performance and user experience issues.

eG Innovations is dedicated to helping businesses across the globe transform IT service delivery into a competitive advantage and a center for productivity, growth and profit. Many of the world's largest businesses use eG Enterprise to enhance IT service performance, increase operational efficiency, ensure IT effectiveness and deliver on the ROI promise of transformational IT investments across physical, virtual and cloud environments.

To learn more visit www.eginnovations.com.

Contact Us

For support queries, email support@eginnovations.com.

To contact eG Innovations sales team, email sales@eginnovations.com.