Monitoring Citrix XenApp Servers

eG Innovations Product Documentation

www.eginnovations.com

Total Performance Visibility

Table of Contents

Table of Figures

Chapter 1: Introduction

Citrix based environments are growing in popularity as cost-effective, efficient modes of accessing a variety of heterogeneous applications on-demand. By hosting applications on Citrix farms and making them accessible over a distributed network, IT administrators can allow users in different locations effectively access and share hardware resources and software licenses. While such thin-client environments offer economies of scale, there are significant challenges in maintaining and operating these environments. In order to be an effective alternative for desktop applications, Citrix environments must deliver the same quality of service that users have come to expect from their desktop applications.

Typically, Citrix server farms include multiple tiers of software. A front-end web interface (Nfuse or StoreFront) server is used to support web-based accesses to the server farm. Active directory servers handle user authentication and rights association, while user profiles are loaded from profile servers. The authenticated requests are passed to the Citrix XenApp servers that host a number of applications. In turn, the applications may use backend databases, printers, etc., for different functionalities. Owing to the multi-tier nature of Citrix environments, a slow-down in one tier (e.g., the authentication server) can cause a slow-down of the entire service. When a slow-down occurs, an administrator of the Citrix farm has to quickly determine what the source of the problem could be - i.e., Is it the network? Or the web interface server? Or the Active Directory server? Or the profile server? Or the Citrix XenApp server? Or the backend database? Accurate, fast diagnosis of problems helps reduce downtime and improve customer satisfaction.

The eG Enterprise offers 100% web-based monitoring of Citrix XenApp server farms. The eG Enterprise includes extensive, pre-defined, customized models of the different applications in the Citrix farm including Citrix XenApp, Citrix ZDCs, Nfuse server, the Citrix StoreFront server, the access gateways, the netscaler LB, the Secure Ticketing Authority, the Windows domain controllers, infrastructure servers like DNS, LDAP, Active Directory, and other network devices.

This document discusses the monitoring models offered by eG Enterprise for the Citrix XenApp servers.

The foundation of the Citrix Access Suite, Citrix XenApp server, is the world's most widely deployed server for centrally managing heterogeneous applications and delivering their functionality as a service to workers, wherever they may be.

Using a specialized *Citrix XenApp 4/5/6.x* monitoring model, eG Enterprise provides monitoring support to Citrix XenApp Servers 4.0/4.5/5/6/6.5. To monitor *Citrix XenApp servers v7 (and above)*, eG Enterprise offers a dedicated Citrix XenApp monitoring model.

Note:

All versions of Citrix XenApp can be monitored only in an *agent-based* manner.

Chapter 2: Administering the eG Manager to work with a Citrix XenApp Server 4 / 5 / 6.x

To do the above, do the following:

1. Log into the eG administrative interface.
2. eG Enterprise is capable of automatically discovering the Citrix XenApp server. If a Citrix XenApp server is already discovered, then directly proceed towards managing it using the **COMPONENTS - MANAGE/UNMANAGE** page (Infrastructure -> Components -> Manage/Unmanage). However, if it is yet to be discovered, then run discovery (Infrastructure -> Components -> Discovery) to get it discovered or add the Citrix XenApp server manually using the **COMPONENTS** page (Infrastructure -> Components -> Add/Modify). Remember that components manually added are managed automatically. Discovered components, however, are managed using the **COMPONENTS - MANAGE/UNMANAGE** page. Figure 2.1 and Chapter 2 clearly illustrate the process of managing a Citrix XenApp server.

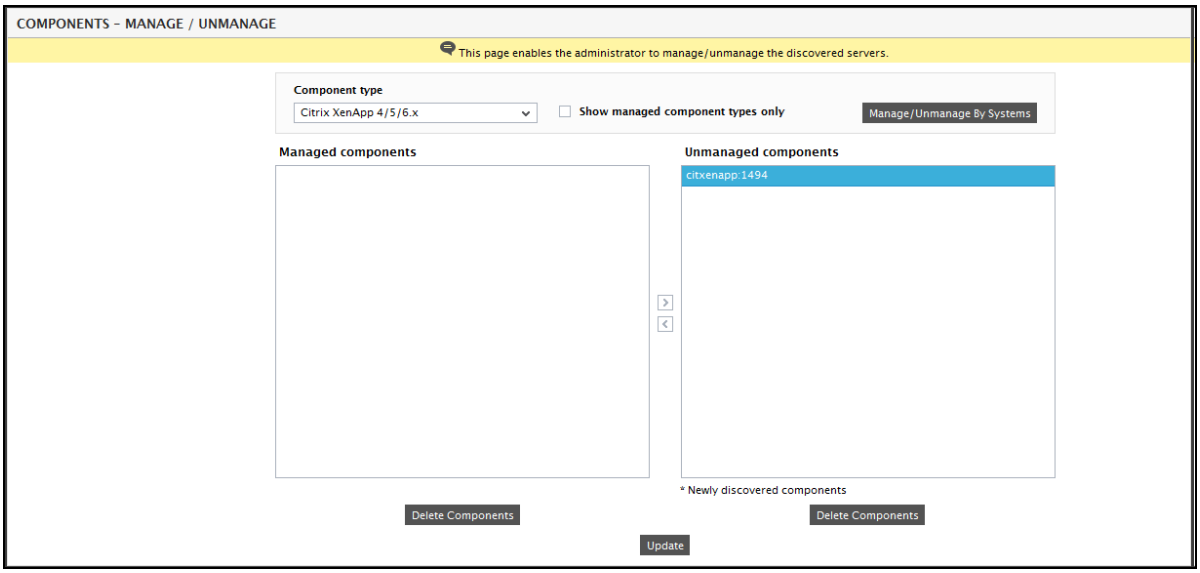


Figure 2.1: Selecting the Citrix XenApp server 4 / 5 / 6.x to be managed

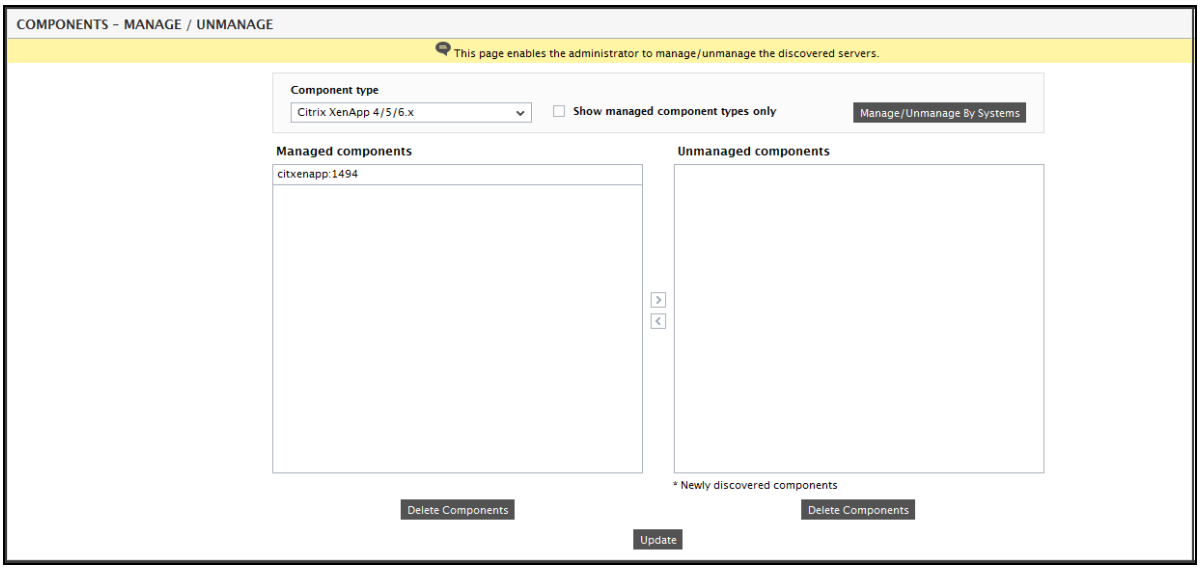


Figure 2.2: Managing the Citrix XenApp 4 / 5 / 6.x server

3. The tests mapped for the Citrix XenApp 4/5/6.x server will be configured automatically.
4. Finally, sign out of the eG administrative interface.

Chapter 3: Monitoring Citrix XenApp Servers 4/5/6.x

In this section, we will be discussing the monitoring capabilities of the *Citrix XenApp 4/5/6.x* monitoring model alone. This model reveals the following:

XenApp Server Monitoring	• Are the Citrix servers available to service user requests? • Are there sporadic disconnects from the Citrix server? • At what times do peak usage of the servers happen and is the server capacity adequate? • Is the user load being balanced across all the servers? • Is the data store available? • What are the access rates to the data store, the dynamic store, and the local host cache? • How much IMA traffic is happening between servers?
User Monitoring	• What is the average response time that critical users are seeing when connecting to a XenApp server? • How many users are logged in to each server in the Citrix farm? • What is the resource usage (CPU and memory) for each user? • Are specific user profiles too large?
Operating System Monitoring	• What is the average CPU and memory usage on all the servers in the farm? • Is any unusual memory scanning/paging activity happening on the systems? • Are the critical XenApp server and IMA processes up? What is their resource consumption?
Published Applications Monitoring	• What are the published applications on a XenApp server? • Who is using each application? • What is the resource usage for each published application?
License Monitoring	• How many product and connection licenses are available in the farm and what is their usage?

	• Are there enough licenses available for each of the published applications?
Infrastructure Services Monitoring	• Is the web interface server forwarding requests to the XenApp server? • Are the backend databases working? • What is the resource usage of the databases? • Are users able to login to the server farm? How long is the login process taking? • What is the usage of the Microsoft Windows Domain Controller?

Figure 3.1: Layer model of a Citrix XenApp server 4/5/6.x

The sections to come elobarate on the tests executing on the **Operating System layer** and each of the top 6 layers of the monitoring model depicted by Figure 3.1, and the measures they report.

3.1 The Operating System Layer

The tests mapped to this layer report the health of the Windows operating system on which the XenApp server operates.

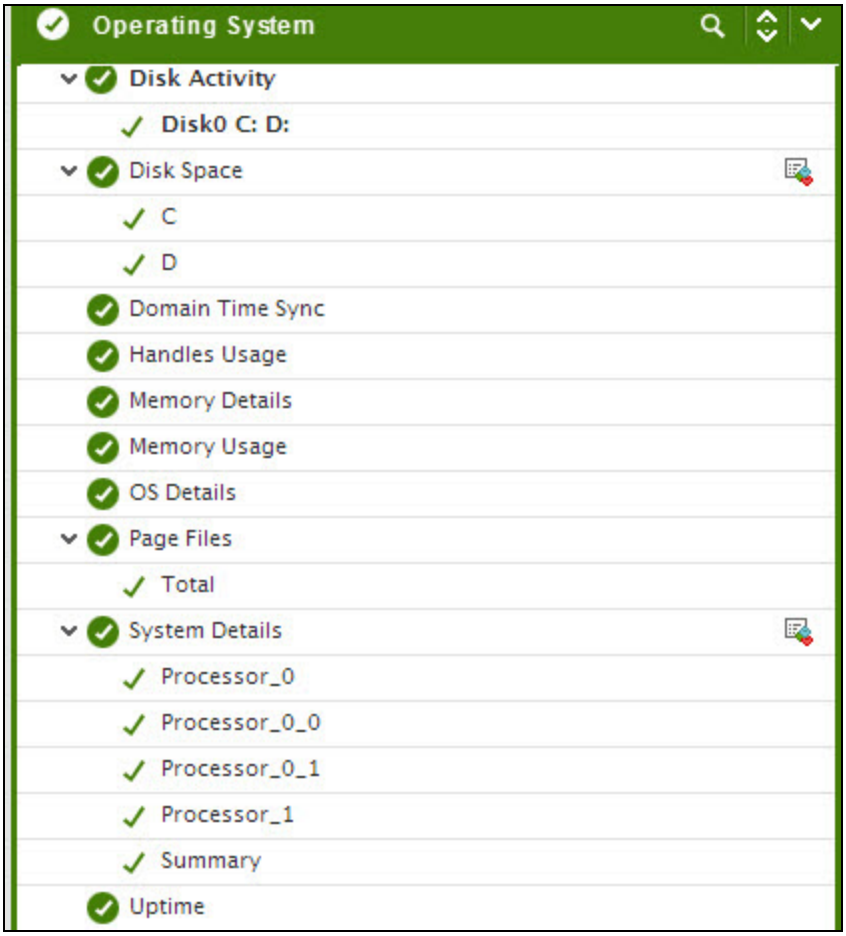


Figure 3.2: The tests mapped to the Operating System layer

All the tests mapped to this layer, except the **PVS Write Cache** test, have already been discussed in the *Monitoring Unix and Windows Servers* document. The sub-section that follows therefore will talk about the **PVS Write Cache** test only.

3.1.1 PVS Write Cache Test

Provisioning Services (PVS) is a service utilized to stream an operating system image from a file, known as a vDisk, to a physical or virtual computer. The recipient of the stream can be a disk less computer with the vDisk acting as its hard disk drive. One of the primary benefits of PVS is the ability

to utilize a single vDisk to stream to multiple computers. This type of vDisk is known as a Standard vDisk and offers increased consistency, security, and centralized management.

Standard vDisks are Read-Only. All modifications, such as application installations, are written to a temporary file known as the Write Cache. When read requests for the newly written files come in, they are read from the write cache.

The Write Cache file can be configured to reside in the following locations:

- Cache on device hard drive: Write cache can exist as a file in NTFS format, located on the target-device's hard drive. This write cache option frees up the Provisioning Server since it does not have to process write requests and does not have the finite limitation of RAM.
- Cache in device RAM: Write cache can exist as a temporary file in the target device's RAM. This provides the fastest method of disk access since memory access is always faster than disk access. This measure will report metrics only if the cache resides in the device RAM.
- Cache in device RAM with overflow on hard disk (only available for Windows 7 and Server 2008 R2 (NT 6.1) and later): In this case, when RAM is zero, the target device write cache is only written to the local disk. When RAM is not zero, the target device write cache is written to RAM first. When RAM is full, the least recently used block of data is written to the local differencing disk to accommodate newer data on RAM. The amount of RAM specified is the non-paged kernel memory that the target device will consume.
- Cache on server: Write cache can exist as a temporary file on a Provisioning Server. In this configuration, all writes are handled by the Provisioning Server, which can increase disk IO and network traffic.
- Cache on server persistent: This cache option allows for the saving of changes between reboots. Using this option, after rebooting, a target device is able to retrieve changes made from previous sessions that differ from the read only vDisk image. If a vDisk is set to Cache on server persistent, each target device that accesses the vDisk automatically has a device-specific, writable disk file created. Any changes made to the vDisk image are written to that file, which is not automatically deleted upon shutdown. This saves target device specific changes that are made to the vDisk image.

For virtual XenApp servers, administrators typically use the server's hard drive for storing the write cache. Storing the write cache on the target side is beneficial as it keeps the write "close" to the target and minimizes the load on the Provisioning Servers, but it requires more resources. If the write-cache does not have enough disk space resources to grow, then many modifications to the vDisk will be lost. This is why, it is imperative that the write-cache is sized right, its usage is tracked continuously, and the lack of adequate disk space for the write cache brought to the attention of administrators rapidly. This is what the **PVS Write Cache** test does! This test monitors the size and

usage of the write cache and proactively alerts administrators when the write-cache runs out of space.

Note:

This test will report metrics only if the write-cache resides in one of the following locations:

- Cache on device hard drive
- Cache on server
- Cache on server persistent

Target of the test : A Provisioned Citrix XenApp server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the provisioned Citrix XenApp server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - Host name of the server for which the test is to be configured
3. **PORT** - Enter the port to which the specified **HOST** listens
4. **PVS WRITE CACHE LOCATION** – Specify the location of the write cache file to be monitored. By default, this will be: *d:\vdiskdiff.vhdx*.
5. **PVS WRITE CACHE MAX SIZE** – Specify the maximum size upto which the write cache file can grow. By default, this is set to 10 GB.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Write cache size:	Indicates the current size of the write cache.	GB	
Write cache utilization:	Indicates the percent usage of the write cache.	Percent	The value of this measure is computed using the following formula: (PVS Write Cache Max Size – Write cache size) / Write cache size * 100

Measurement	Description	Measurement Unit	Interpretation
			If the value of this measure is close to 100%, it indicates that the write cache may soon run out of space. Under such circumstances, you have the following options: • You can increase the maximum size to which write cache can grow, or; • Redirect some items out of the write cache and into a persistent drive. Before increasing the maximum write cache size, you will have to take the following into account: • Basically the write cache will store all writes which would have gone to the hard disk. So if a user tends to copy large files locally to his/her desktop the write cache will grow at the same pace as the files are transferred. If there is any application which caches files or portions of a central DB locally for better performance, then the write cache will grow again. • But there are some items which will always hit the write cache and these are split into two areas again. On one hand there is the user space, which contains items such as the user profile or internet/application related temp files. The user related write cache

Measurement	Description	Measurement Unit	Interpretation
			disk space needs to be multiplied by the amount of users logged on to a particular system. • On the other hand we have the system space, which contains items such as logs or system temp / cache files, but we will also find files which are modified by the OS or any service for whatever reason. The system related write cache disk space is typically larger for server operating systems than for workstations. If you choose to redirect, then one/more of the following items can be redirected: • Windows Pagefile. In fact the PVS Target Device driver detects if a local drive is available and redirects the pagefile automatically. • Windows Event Log. While the eventlog is typically quite small (maybe 100MB or so) many customers redirect it for supportability and traceability reasons. • Citrix related logs. Same as Windows Event Log. • Anti Virus pattern. In case the virus scanner allows redirecting the pattern file, doing so saves some write cache space but it also saves some network traffic as it is not required to

Measurement	Description	Measurement Unit	Interpretation
			load the pattern from scratch after every reboot.

3.1.2 Grid GPUs Test

GPU-accelerated computing is the use of a graphics processing unit (GPU) together with a CPU to accelerate scientific, analytics, engineering, consumer, and enterprise applications. GPU-accelerated computing enhances application performance by offloading compute-intensive portions of the application to the GPU, while the remainder of the code still runs on the CPU.

Imagine if you could access to your GPU-accelerated applications anywhere on any device, even those requiring intensive graphics power. NVIDIA GRID makes this possible. With NVIDIA GRID, a virtualized GPU designed specifically for virtualized server environments, data center managers can bring true PC graphics-rich experiences to users.

The NVIDIA GRID GPUs will be hosted in enterprise data centers and allow users to run virtual desktops or virtual applications on multiple devices connected to the internet and across multiple operating systems, including PCs, notebooks, tablets and even smartphones. Users can utilize their online-connected devices to enjoy the GPU power remotely.

Virtual application delivery with XenApp and NVIDIA GRID™ offloads graphics processing from the CPU to the GPU, allowing the data center manager to deliver to all user types for the first time.

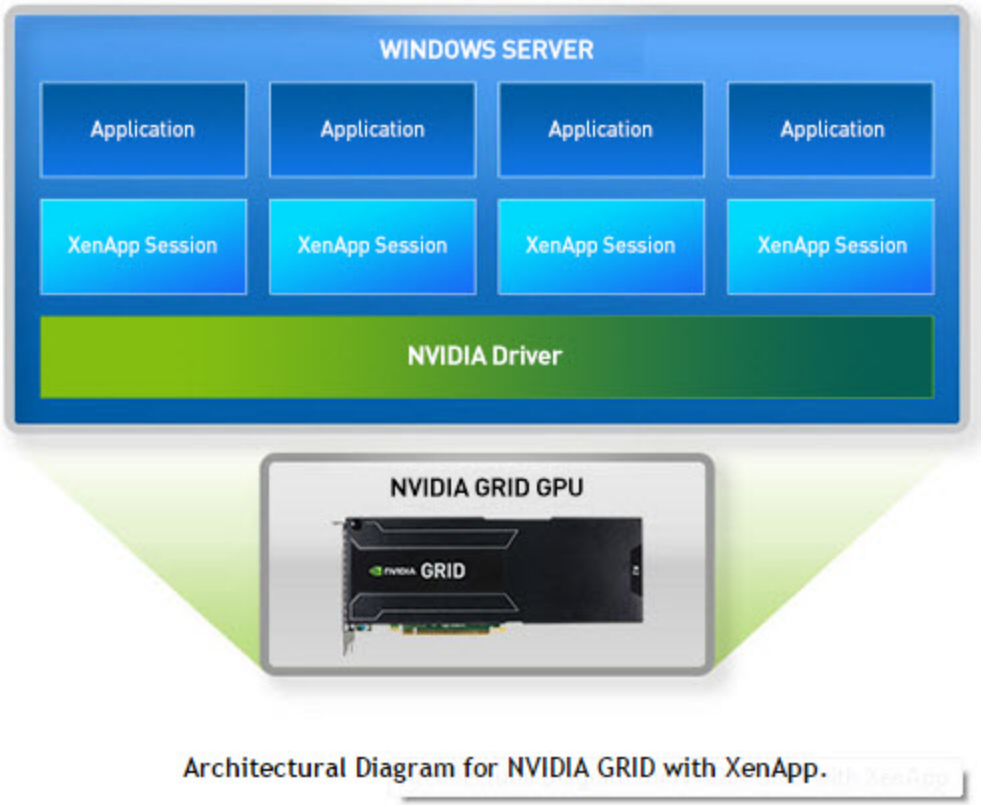


Figure 3.3: An Architectural diagram for NVIDIA GRID with XenApp

In GPU-enabled Citrix XenApp environments, if users to virtual applications complain of slowness when accessing graphic applications, administrators must be able to instantly figure out what is causing the slowness – is it because adequate GPU resources are not available to the host? Or is it because of excessive utilization of GPU memory and processing resources by a few virtual applications on the host? Accurate answers to these questions can help administrators determine whether/not:

- The host is sized with sufficient GPU resources;
- The GPUs are configured with enough graphics memory;

Measures to right-size the host and fine-tune its GPU configuration can be initiated based on the results of this analysis. This is exactly what the **Grid GPUs** test helps administrators achieve!

Using this test, administrators can identify the physical GPUs on the NVIDIA GRID card used by the host. For each physical GPU, administrators can determine how actively memory on that GPU is utilized, thus revealing the GPU on which memory is used consistently. In addition, the test also indicates how busy each GPU is, and in the process pinpoints those physical GPUs that are being

over-utilized by the virtual applications on the host. The adequacy of the physical GPU resources is thus revealed. Moreover, the power consumption and temperature of each GPU of the host is also monitored and its current temperature and power usage can be ascertained; administrators are thus alerted to abnormal power usage of the GPU and unexpected fluctuations in its temperature. The power limit set and the clock frequencies configured are also revealed, so that administrators can figure out whether the GPU is rightly configured for optimal processing or is any fine-tuning required.

Target of the test : A Citrix XenApp server / Microsoft RDS

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each GRID physical GPU assigned to the host being monitored

Configuration 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 Citrix server
4. **NVIDIA PATH** – Specify the full path to the install directory of the NVIDIA. By default, the NVIDIA will be installed in the *C:/Progra~1/NVIDIA~1/NVSMI* directory. If the NVIDIA indeed resides in its default location, set the **NVIDIA PATH** to *none*. On the other hand, if the NVIDIA has been installed in a different location, provide the full path to that location against **NVIDIA PATH**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
GPU memory utilization:	Indicates the proportion of time over the past sample period during which global (device) memory was being read or written on this GPU.	Percent	A value close to 100% is a cause for concern as it indicates that graphics memory on a GPU is almost always in use. In a XenApp environment, this could be because one/more sessions to XenApp are consistently accessing graphic-intensive applications.

Measurement	Description	Measurement Unit	Interpretation
			In a Shared vGPU environment on the other hand, memory may be consumed all the time, if one/more VMs/virtual desktops on the host utilize the graphics memory excessively and constantly. If you find that only a single VM/virtual desktop has been consistently hogging the graphic memory resources, you may want to switch to the Dedicated GPU mode, so that excessive memory usage by that VM/virtual desktop has no impact on the performance of other VMs/virtual desktops on that host. If the value of this measure is high almost all the time for most of the GPUs, it could mean that the host is not sized with adequate graphics memory.
Allocated frame buffer memory:	Indicates the amount of frame buffer memory on-board this GPU that has been allocated to the host.	MiB	Frame buffer memory refers to the memory used to hold pixel properties such as color, alpha, depth, stencil, mask, etc. Properties like the screen resolution, color level, and refresh speed of the frame buffer can impact graphics performance. Also, if Error-correcting code (ECC) is enabled on a host, the available frame buffer memory may be decreased by several percent. This is because, ECC uses up memory to

Measurement	Description	Measurement Unit	Interpretation
			detect and correct the most common kinds of internal data corruption. Moreover, the driver may also reserve a small amount of memory for internal use, even without active work on the GPU; this too may impact frame buffer memory. For optimal graphics performance therefore, adequate frame buffer memory should be allocated to the host.
Unallocated frame buffer memory:	Indicates the amount of frame buffer memory on-board this GPU that has not been allocated to the host.	MiB	
GPU compute utilization:	Indicates the proportion of time over the past sample period during which one or more kernels was executing on this GPU.	Percent	A value close to 100% indicates that the GPU is busy processing graphic requests almost all the time. In a XenApp environment, this could be because one/more sessions to XenApp are consistently accessing graphic-intensive applications. In a Shared vGPU environment on the other hand, a GPU may be in use almost all the time, if one/more VMs/virtual desktops on the host are running highly graphic- intensive applications. If you find that only a single VM/virtual desktop has been consistently hogging the GPU resources, you may want to switch to the Dedicated GPU mode, so that

Measurement	Description	Measurement Unit	Interpretation
			excessive GPU usage by that VM/virtual desktop has no impact on the performance of other VMs/virtual desktops on that host. If all GPUs are found to be busy most of the time, you may want to consider augmenting the GPU resources of the host. Compare the value of this measure across physical GPUs to know which GPU is being used more than the rest.
Power consumption:	Indicates the current power usage of this GPU.	Watts	A very high value is indicative of excessive power usage by the GPU. In such cases, you may want to enable Power management so that the GPU limits power draw under load to fit within a predefined power envelope by manipulating the current performance state.
Core GPU temperature:	Indicates the current temperature of this GPU.	Celsius	Ideally, the value of this measure should be low. A very high value is indicative of abnormal GPU temperature.
Total framebuffer memory:	Indicates the total size of frame buffer memory of this GPU.	MiB	Frame buffer memory refers to the memory used to hold pixel properties such as color, alpha, depth, stencil, mask, etc.
Total BAR1 memory:	Indicates the total size of the BAR1 memory of this GPU.	MiB	BAR1 is used to map the frame buffer (device memory) so that it can be directly accessed by the CPU or by 3rd party devices (peer-to-peer

Measurement	Description	Measurement Unit	Interpretation
			on the PCIe bus).
Allocated BAR1 memory:	Indicates the amount of BAR1 memory on this GPU that is allocated to the host.	MiB	For better user experience with graphic applications, enough BAR1 memory should be available to the host.
Unallocated BAR1 memory:	Indicates the total size of BAR1 memory of this GPU that is still not allocated to the host.	MiB	
Power management:	Indicates whether/not power management is enabled for this GPU.		Many NVIDIA graphics cards support multiple performance levels so that the server can save power when full graphics performance is not required. The default Power Management Mode of the graphics card is Adaptive. In this mode, the graphics card monitors GPU usage and seamlessly switches between modes based on the performance demands of the application. This allows the GPU to always use the minimum amount of power required to run a given application. This mode is recommended by NVIDIA for best overall balance of power and performance. If the power management mode is set to Adaptive, the value of this measure will be Supported. Alternatively, you can set the Power Management Mode to Maximum Performance. This mode allows

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>users to maintain the card at its maximum performance level when 3D applications are running regardless of GPU usage. If the power management mode of a GPU is Maximum Performance, then the value of this measure will be Maximum.

The numeric values that correspond to these measure values are discussed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Supported</td><td>1</td></tr><tr><td>Maximum</td><td>0</td></tr></table>
Note:

By default, this measure will report the Measure Values listed in the table above to indicate the power management status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Power limit:</td><td>Indicates the power limit configured for this GPU.</td><td>Watts</td><td>This measure will report a value only if the value of the ‘Power management’ measure is ‘Supported’.

The power limit setting controls how much voltage a GPU can use when under load. Its not advisable to set the power limit at its maximum – i.e., the value of this measure should not be the same as the value of the Max</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			power limit measure - as it can cause the GPU to behave strangely under duress.
Default power limit:	Indicates the default power management algorithm's power ceiling for this GPU.	Watts	This measure will report a value only if the value of the 'Power management' measure is 'Supported'.
Enforced power limit:	Indicates the power management algorithm's power ceiling for this GPU.	Watts	This measure will report a value only if the value of the 'Power management' measure is 'Supported'. The total board power draw is manipulated by the power management algorithm such that it stays under the value reported by this measure.
Min power limit:	The minimum value that the power limit of this GPU can be set to.	Watts	This measure will report a value only if the value of the 'Power management' measure is 'Supported'.
Max power limit:	The maximum value that the power limit of this GPU can be set to.	Watts	This measure will report a value only if the value of the 'Power management' measure is 'Supported'. If the value of this measure is the same as that of the Power limit measure, then the GPU may behave strangely.
Graphics clock:	Indicates the current frequency of the graphics clock of this GPU.	MHz	GPU has many more cores than your average CPU but these cores are much simpler and much smaller so that many more actually fit on a

Measurement	Description	Measurement Unit	Interpretation
			small piece of silicon. These smaller, simpler cores go by different names depending upon the tasks they perform. Stream processors are the cores that perform a single thread at a slow rate. But since GPUs contain numerous stream processors, they make overall computation high. The streaming multiprocessor clock refers to how fast the stream processors run. The Graphics clock is the speed at which the GPU operates. The memory clock is how fast the memory on the card runs. By correlating the frequencies of these clocks (i.e., the value of these measures) with the memory usage, power usage, and overall performance of the GPU, you can figure out if overclocking is required or not. Overclocking is the process of forcing a GPU core/memory to run faster than its manufactured frequency. Overclocking can have both positive and negative effects on GPU performance. For instance, memory overclocking helps on cards with low memory bandwidth, and with games with a lot of post-processing/textures/filters like AA that are VRAM intensive. On the other hand, speeding up the operation frequency of a shader/streaming processor/memory, without properly analyzing its need and its effects, may increase its thermal output in a linear fashion. At the same time,

Measurement	Description	Measurement Unit	Interpretation
Streaming multiprocessor clock:	Indicates the current frequency of the streaming multiprocessor clock of this GPU.	MHz	
Memory clock:	Indicates the current frequency of the memory clock of this GPU.	MHz	
Fan speed:	Indicates the percent of maximum speed that this GPU's fan is currently intended to run at.	Percent	The value of this measure could range from 0 to 100%. An abnormally high value for this measure could indicate a problem condition – eg., a sudden surge in the temperature of the GPU that could cause the fan to spin faster. Note that the reported speed is only the intended fan speed. If the fan is physically blocked and unable to spin, this output will not match the actual fan speed. Many parts do not report fan speeds because they rely on cooling via fans in the surrounding enclosure. By default the fan speed is increased or decreased automatically in response to changes in temperature.
Compute processes:	Indicates the number of processes having compute context on this GPU.	Number	Use the detailed diagnosis of this measure to know which processes are currently using the GPU. The process details provided as part of the detailed diagnosis include, the PID of the process, the process

Measurement	Description	Measurement Unit	Interpretation
			name, and the GPU memory used by the process. Note that the GPU memory usage of the processes will not be available in the detailed diagnosis, if the Windows platform on which XenApp operates is running in the WDDM mode. In this mode, the Windows KMD manages all the memory, and not the NVIDIA driver. Therefore, the NVIDIA SMI commands that the test uses to collect metrics will not be able to capture the GPU memory usage of the processes.
Volatile single bit errors:	Indicates the number of volatile single bit errors in this GPU.	Number	Volatile error counters track the number of errors detected since the last driver load. Single bit ECC errors are automatically corrected by the hardware and do not result in data corruption. Ideally, the value of this measure should be 0.
Volatile double bit errors:	Indicates the total number of volatile double bit errors in this GPU.	Number	Volatile error counters track the number of errors detected since the last driver load. Double bit errors are detected but not corrected. Ideally, the value of this measure should be 0.
Aggregate single bit errors:	Indicates the total number of aggregate single bit errors in this GPU.	Number	Aggregate error counts persist indefinitely and thus act as a lifetime counter. Single bit ECC errors are automatically corrected by the hardware and do not result in data

Measurement	Description	Measurement Unit	Interpretation
			corruption. Ideally, the value of this measure should be 0.
Aggregate double bit errors:	Indicates the total number of aggregate double bit errors in this GPU.	Number	Aggregate error counts persist indefinitely and thus act as a lifetime counter. Double bit errors are detected but not corrected. Ideally, the value of this measure should be 0.

3.2 The Application Processes Layer

Using the tests mapped to this layer, you can do the following:

- Capture key application and system error events that have occurred on the server;
- Verify whether the processes critical to the functioning of the Citrix server are currently operational or not, and also monitor the CPU/memory usage of these processes;
- Periodically check the availability of the Citrix server's TCP port, the responsiveness of the port to client requests, and also the availability of ICA connection to the port.

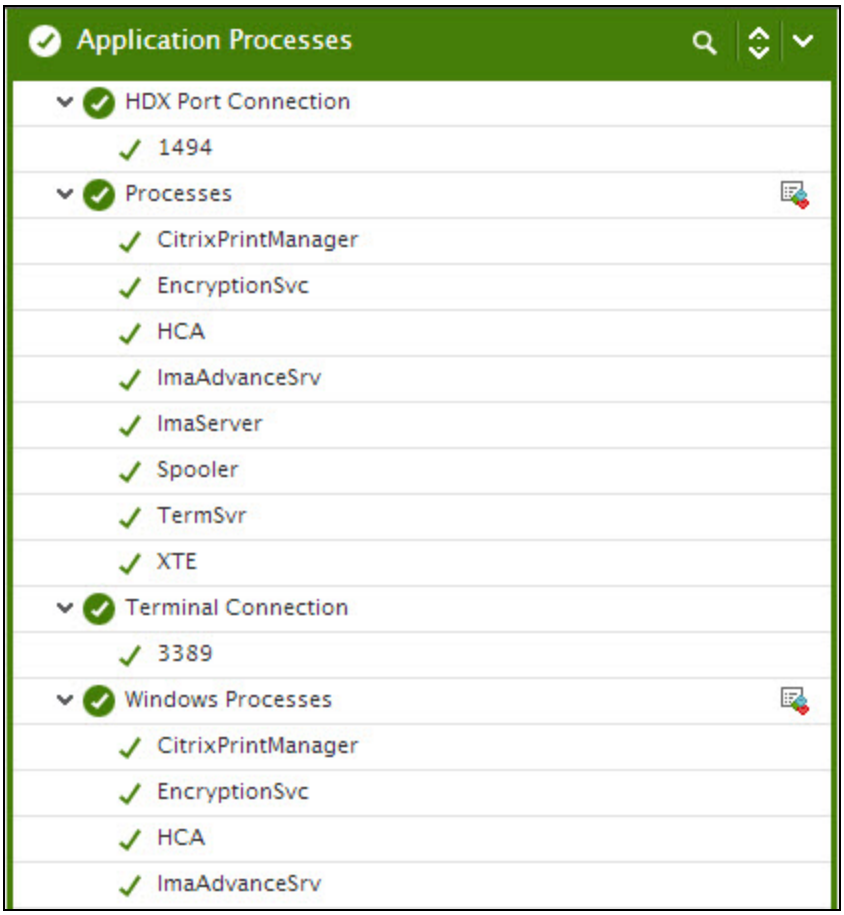


Figure 3.4: The tests mapped to the Application Processes layer

The section that follows will discuss the **IcaConnection** test alone, as all other tests mapped to this layer have already been discussed in the *Monitoring Unix and Windows Servers* document.

3.2.1 HDX Port Connection Test

This test primarily checks whether the critical TCP ports on the Citrix server are up/down, and reports the responsiveness of each configured port to client requests. For a Citrix server however, these checks might not be adequate at all times; you could have a case where the Citrix server port is up but the server is still not responding. When a connection is made to the Citrix server, it will typically send a message "ICA" to the client. This check connects to the port and then validates the response from the citrix server to see if the ICA stream is being received by the client. Hence, this test additionally reports the ICA connection availability.

Target of the test : A Citrix server

Agent deploying the testn : An external agent

Outputs of the test : One set of results for every configured port name or port number

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - Host name of the server for which the test is to be configured
3. **PORT** - Enter the port to which the specified **HOST** listens
4. **TARGETPORTS** – Specify either a comma-separated list of port numbers that are to be tested (eg., 1494,1495,1496), or a comma-separated list of *port name:port number* pairs that are to be tested (eg., ica:1494,smtp:25,mssql:1433). In the latter case, the port name will be displayed in the monitor interface. Alternatively, this parameter can take a comma-separated list of *port name:IP address:port number* pairs that are to be tested, so as to enable the test to try and connect to Tcp ports on multiple IP addresses. For example, *mysql:192.168.0.102:1433,egwebsite:209.15.165.127:80*.
5. **TIMEOUT** - Here, specify the maximum duration (in seconds) for which the test will wait for a response from the server. The default **TIMEOUT** period is 60 seconds.
6. **ISPASSIVE** – If the value chosen is **YES**, then the server under consideration is a passive server in a cluster. No alerts will be generated if the server is not running. Measures will be reported as "Not applicable' by the agent if the server is not up.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
TCP connection availability:	Whether the TCP connection is available or not.	Percent	An availability problem can be caused by different factors – e.g., the server process may not be up, a network problem may exist, or there could be a configuration problem with the DNS server.
Response time:	Time taken (in seconds) by the server to respond to a request.	Secs	An increase in response time can be caused by several factors such as a server bottleneck, a configuration problem with the DNS server, a network problem, etc.

Measurement	Description	Measurement Unit	Interpretation
ICA connection availability:	Indicates whether ICA connection is available or not.	Percent	While the value 100 for this measure indicates that the ICA stream is being received by the client, the value 0 indicates that it is not.

3.3 The Windows Services Layer

The test mapped to this layer indicates whether the Windows services critical to the functioning of the Citrix server are currently available or not.

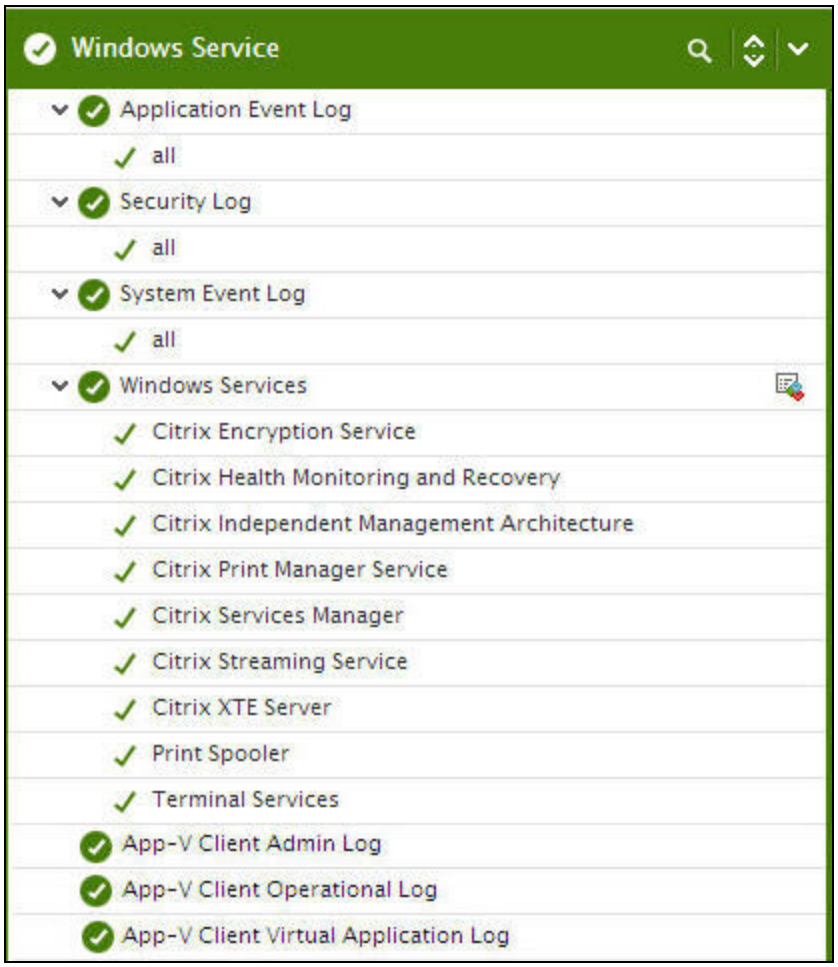


Figure 3.5: The test mapped to the Windows Services layer

Since most of the tests mapped to this layer have already been dealt with in the *Monitoring Unix and Windows Servers* document, let us now discuss the tests that are exclusive for this server.

3.3.1 App-V Client Admin Log Test

This test reports the statistical information about the admin events generated by the target system.

Note:

This test will report metrics only when the App-V Client is installed on the Citrix XenApp Server.

Target of the test : An App-V Client on the target Citrix XenApp Server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the App-V Client that is to be 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** - Specify the port at which the specified host listens to. By default, this is *8080*.
4. **LOGTYPE** - Refers to the type of event logs to be monitored. The default value is *Microsoft-AppV-Client/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*.
- Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use *all* to indicate that all the event sources have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded.
- 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 filterparameter contains the value: all. 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 ⊕ icon appears near the **FILTER** list box, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the ⊕ icon 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. **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**.

10. **DDFORWARNING** -To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **No**.

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 **DDFREQUENCY**.

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

    The option to selectively enabled/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
Information messages:	Indicates the number of App- V Client admin 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 by one or more applications. Please check the App- V Client admin logs in the Event Log Viewer for more details.
Warnings:	Indicates the number of App- V Clientadmin warnings that were generated when the test was last executed.	Number	A high value of this measure indicates application problems that may not have an immediate impact, but may cause future problems in one or more applications. Please check the App- V Client admin logs in the Event Log Viewer for more details.
Error messages:	Indicates the number of App- V Client admin error events that were generated during the last measurement period.	Number	A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems. An increasing trend or high value indicates the existence of problems like loss of functionality or data in one or more applications. Please check the App- V Client admin logs in the Event Log Viewer for more details.
Critical messages:	Indicates the number of App- V Client admin	Number	A very low value (zero) indicates that the system is in a healthy state

Measurement	Description	Measurement Unit	Interpretation
	critical error events that were generated when the test was last executed.		and all applications are running smoothly without any potential problems. An increasing trend or high value indicates the existence of fatal/irrepairable problems in one or more applications. Please check the App- V Client admin logs in the Event Log Viewer for more details.
Verbose messages:	Indicates the number of App- V Client admin verbose events that were generated when the test was last executed.	Number	The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the App- V Client admin logs in the Event Log Viewer for more details.

3.3.2 App-V Client Operational Log Test

This test reports the statistical information about the operation events generated by the target system.

Note:

This test will report metrics only when the App-V Client is installed on the Citrix XenApp Server.

Target of the test : An App-V Client on the target Citrix XenApp Server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the App-V Client that is to be 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** - Specify the port at which the specified **HOST** listens to. By default, this is 8080.
4. **LOGTYPE** - Refers to the type of event logs to be monitored. The default value is *Microsoft-AppV-Client/Operational*.
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*.
   - Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use *all* to indicate that all the event sources have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded.
   - 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. 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 icon appears near the **FILTER** list box, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the icon 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. **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**.
10. **DDFORWARNING** - To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **NO**.

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

    The option to selectively enabled/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
Information messages:	Indicates the number of App-V Client operational 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 by one or more applications.
Warnings:	Indicates the number of App-V Client operational warnings that were generated when the test was last executed.	Number	A high value of this measure indicates application problems that may not have an immediate impact, but may cause future problems in one or more applications.
Error messages:	Indicates the number of App-V Client operational error events that were generated during the last measurement	Number	A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems.

Measurement	Description	Measurement Unit	Interpretation
	period.		An increasing trend or high value indicates the existence of problems like loss of functionality or data in one or more applications.
Critical messages:	Indicates the number of App-V Client operational critical error events that were generated when the test was last executed.	Number	A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems. An increasing trend or high value indicates the existence of fatal/irrepairable problems in one or more applications.
Verbose messages:	Indicates the number of App-V Client operational verbose events that were generated when the test was last executed.	Number	The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period.

3.3.3 App-V Client Virtual Application Log Test

This test reports the statistical information about the virtual application events generated by the target system.

Note:

This test will report metrics only when the App-V Client is installed on the Citrix XenApp Server.

Target of the test : An App-V Client on the target Citrix XenApp Server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the App-V Client that is to be 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** – Specify the port at which the specified HOST listens to. By default, this is 8080.
4. **LOGTYPE** – Refers to the type of event logs to be monitored. The default value is *Microsoft-AppV-Client/Virtual Applications*.
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*.
   - Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use *all* to indicate that all the event sources have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded.
   - 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*. 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 icon appears near the **FILTER** list box, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the icon 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. **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**.

10. **DDFORWARNING** – To ensure that the test does not generate and store detailed measures for warning events, set the **DDFORWARNING** flag to **No**.
11. **DD FREQUENCY**- Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
12. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enabled/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
Information messages:	Indicates the number of App- V Client virtual application informational events that were generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed by one or more applications. Please check the App- V Client Virtual Application logs in the Event Log Viewer for more details.
Warnings:	Indicates the number of App- V Client virtual application warnings	Number	A high value of this measure indicates application problems that may not have an immediate impact,

Measurement	Description	Measurement Unit	Interpretation
	that were generated when the test was last executed.		but may cause future problems in one or more applications. Please check the App-V Client Virtual Application logs in the Event Log Viewer for more details.
Error messages:	Indicates the number of App-V Client virtual application error events that were generated during the last measurement period.	Number	A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems. An increasing trend or high value indicates the existence of problems like loss of functionality or data in one or more applications. Please check the App-V Client Virtual Application logs in the Event Log Viewer for more details. Please check the App-V Client Virtual Application logs in the Event Log Viewer for more details.
Critical messages:	Indicates the number of App-V Client virtual applications critical error events that were generated when the test was last executed.	Number	A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems. An increasing trend or high value indicates the existence of fatal/irrepairable problems in one or more applications. Please check the App-V Client Virtual Application logs in the Event

Measurement	Description	Measurement Unit	Interpretation
			Log Viewer for more details.
Verbose messages:	Indicates the number of App- V Client virtual application verbose events that were generated when the test was last executed.	Number	The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the App- V Client Virtual Application logs in the Event Log Viewer for more details.

3.3.4 WinSock Errors Test

In computing, the Windows Sockets API (WSA), which was later shortened to Winsock, is a technical specificatio that defines how Windows network software should access network services, especially TCP/IP. It defines a standard interface between a Windows TCP/IP client application (such as an FTP client or a web browser) and the underlying TCP/IP protocol stack.

The **WinSock Errors** test scans the Windows event logs for winsock-related errors and reports the count of such errors.

Target of the test : A Citrix XenApp server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Client that is to be 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** – Specify the port at which the specified **HOST** listens to. By default, this is 8080.
4. **LOGTYPE** – Refers to the type of event logs to be monitored. The default value is *Microsoft-Windows-Winsock-AFD/Operational*.
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*.
   - Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use *all* to indicate that all the event sources have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded.
   - 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*. 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 icon appears near the **FILTER** list box, once the **YES** option is chosen against **POLICY BASED FILTER**. Clicking on the icon 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. **DD FREQUENCY**- Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
10. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enabled/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
Send errors:	Indicates the number of send errors captured by the event log during the last measurement period.	Number	The send function and WSAsend functions send data on a connected socket. The value of this measure will be incremented when errors are returned on failed send and WSAsend requests. Ideally, the value of this measure should be 0. In case of a non-zero value, use the detailed diagnosis of this measure to know what send errors occurred. Typically, event IDs 1003, 1005, 1007, 1011, 1013, and 3007 are classified as send errors.
Receive errors:	Indicates the number of receive errors captured by the event log during the last measurement period.	Number	The recv, WSARecv, and WSARecvEx functions receive data from a connected socket or a bound connectionless socket. If the recv, WSARecv, and WSARecvEx requests fail and return errors, such errors are captured by the event log. The value of this measure represents the count of these errors. Ideally, the value of this measure should be 0. In case of a non-zero value, use the detailed diagnosis of this measure to know what receive errors occurred. Typically, event IDs 1004, 1006, 1009, 1012, 1015 are classified as receive errors.
Connect errors	Indicates the number of connect errors captured	Number	The connect, ConnectEx, WSAConnect, WSAConnectByList,

Measurement	Description	Measurement Unit	Interpretation
	by the event log during the last measurement period.		or WSAConnectByName functions typically establish a connection to a specified socket. If calls to these functions fail owing to errors, such error events are captured by the event logs. The value of this measure denotes the count of such errors. Ideally, the value of this measure should be 0. In case of a non-zero value, use the detailed diagnosis of this measure to know what connect errors occurred. Typically, event IDs 1017, 1018, 1020, 1021, 3006 are classified as connect errors.
Accept errors:	Indicates the number of accept errors that occurred during the last measurement period.	Number	The accept, AcceptEx, and WSAAccept functions permit an incoming connection attempt on a socket. If calls to any of these functions fail, then the errors causing the failures are captured by the event logs. The value of this measure denotes the count of such errors. Ideally, the value of this measure should be 0. In case of a non-zero value, use the detailed diagnosis of this measure to know what accept errors occurred. Typically, event IDs 1023, 1024, 1026, 1027 are classified as accept errors.
Bind errors:	Indicates the number of bind errors that occurred during the last	Number	If the implicit or explicit binding of a socket handle fails, then errors causing the bind failure will be

Measurement	Description	Measurement Unit	Interpretation
	measurement period.		captured by the event logs. The value of this measure denotes the count of such errors. Ideally, the value of this measure should be 0. In case of a non-zero value, use the detailed diagnosis of this measure to know what bind errors occurred. Typically, event IDs 1029 and 1030 are classified as bind errors.
Abort errors:	Indicates the number of abort errors that occurred during the last measurement period.	Number	An abort/cancel operation can be Winsock- initiated or transport-initiated. The value of this measure represents the count of both types of abort operations. A Winsock-initiated abort can occur due to the following reasons: • An abort due to unread receive data buffered after close. • An abort after a call to the shutdown function with the how parameter set to SD_RECEIVE and a call to the closesocket function with receive data pending. • An abort after a failed attempt to flush the endpoint. • An abort after an internal Winsock error occurred. • An abort due to a connection with errors and the application previously requested that the

Measurement	Description	Measurement Unit	Interpretation
			connection be aborted on certain circumstances. One example of this case would be an application that set SO_ LINGER with a timeout of zero and there is still unacknowledged data on the connection. • An abort on a connection not fully associated with accepting endpoint. • An abort on a failed call to the accept or AcceptEx function. • An abort due to a failed receive operation. • An abort due to a Plug and Play event. • An abort due to a failed flush request. • An abort due to a failed expedited data receive request. • An abort due to a failed send request. • An abort due to canceled send request. • An abort due to a canceled called to the TransmitPackets function. A transport-initiated abort can occur if a reset is indicated by the transport. Ideally, the value of this measure should be 0. In case of a non-zero

Measurement	Description	Measurement Unit	Interpretation
			value, use the detailed diagnosis of this measure to why aborts occurred. Typically, event IDs 1032 and 1033 are classified as abort errors.
Listen errors:	Indicates the number of listen errors that occurred during the last measurement period.	Number	Ideally, the value of this measure should be 0. In case of a non-zero value, use the detailed diagnosis of this measure to know what listen errors occurred. Typically, event IDs 1026 and 1037 are classified as listen errors.
Indication errors:	Indicates the number of indication errors that occurred during the last measurement period.	Number	An indicated operation can be: • A connection indicated operation: This occurs when an application receives a connection request. • A data indicated operation: This occurs when an application receives data on a connected socket. • Data indicated from transport operations: This occurs when an application posts a receive request and receives data. • Disconnect indicated from transport operations: This occurs when an application receives a disconnect indication. Errors in these processes are categorized under Indication errors. Ideally, the value of this measure should be 0. In case of a non-zero

Measurement	Description	Measurement Unit	Interpretation
			value, use the detailed diagnosis of this measure to know what indication errors occurred. Typically, event IDs 3000, 3001, 3003, 3004 are classified as indication errors.
Other errors:	Indicates the number of other errors that occurred during the last measurement period.	Number	Errors that cannot be classified as send, receive, connect, accept, bind, abort, listen, or indication, will be grouped under Other errors. Ideally, the value of this measure should be 0. In case of a non-zero value, use the detailed diagnosis of this measure to know what other errors occurred. Typically, event IDs 1000,1001,1002,1035 are classified as other errors.

3.4 The Remote Desktop Services Layer

In most environments, the Citrix XenApp server functions in conjunction with a Microsoft RDS server. To enable the administrators of Citrix environment to monitor the movement and resource usage of the **RDS Remote Desktop Services** layer. Figure 3.6 depicts the Microsoft RDS server tests that execute on this layer.

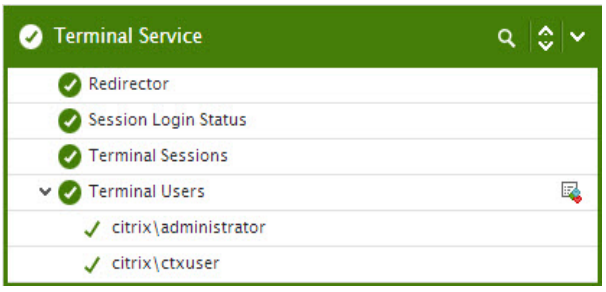


Figure 3.6: The tests associated with the Remote Desktop Services layer

These tests are the same as those mapped to the **Remote Desktop Services** layer of a Microsoft RDS server. These tests hence, have already been dealt with elaborately in the *Monitoring Microsoft*

RDS Servers chapter of the *Monitoring Microsoft Applications* document. So, let us proceed to look at the **Citrix Server** layer.

3.5 The Citrix Server Layer

Citrix server-related performance parameters are monitored by the tests mapped to the **Citrix Server** layer. This includes:

- The Citrix IMA architecture
- Processing and database updation capabilities of the server
- License usage
- Profile size
- User login and profile loading process
- The data and dynamic stores

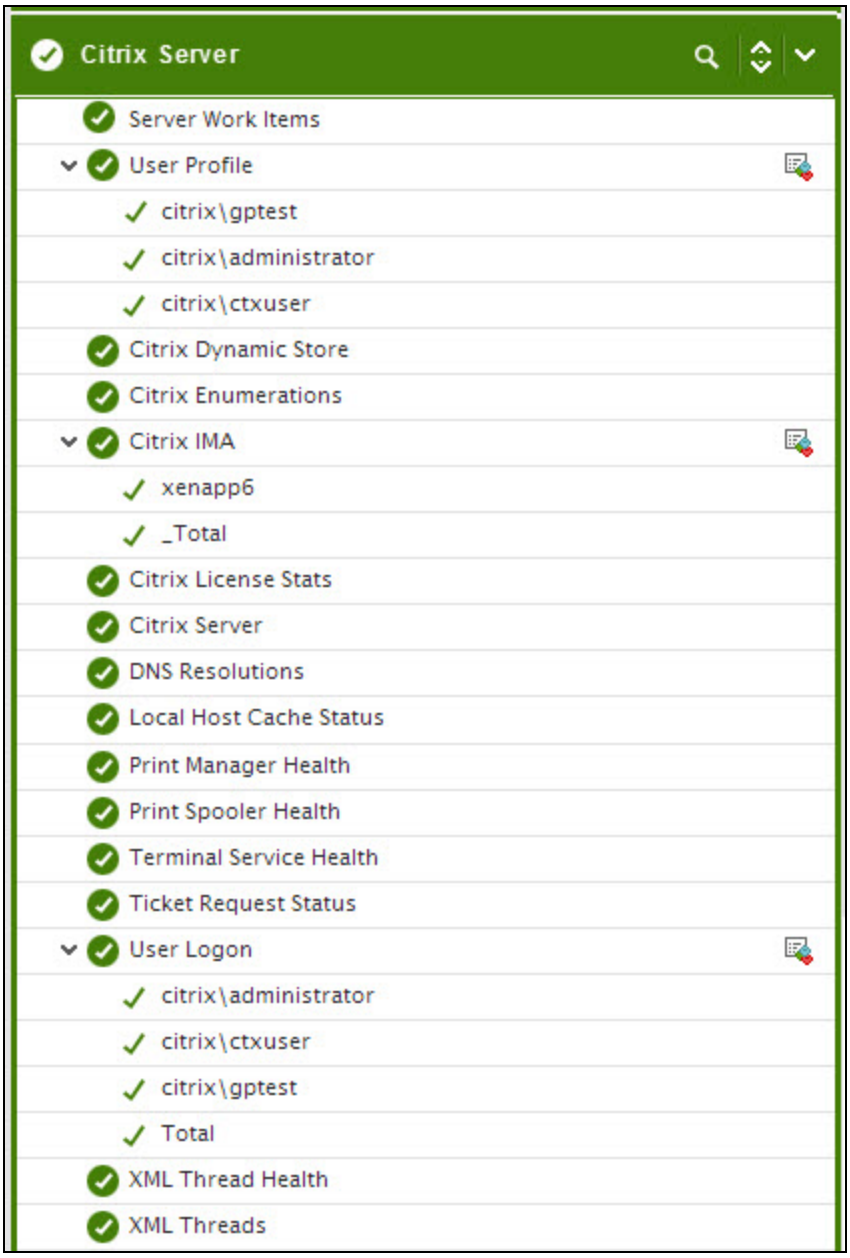


Figure 3.7: The tests associated with the Citrix Server layer

3.5.1 DNS Resolutions Test

This test performs a forward DNS lookup using the local host name to query the local DNS server in the computer's environment for the computer's IP address, and reports whether the lookup was successful or not.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **HEALTH MONITOR TEST PATH** - Citrix XenApp is bundled with a **Health Monitoring and Recovery** (HMR) test pack, which provides a standard set of tests that can be configured to monitor the health of many XenApp components and report failures. Since the eG agent runs one of the HMR tests to determine the status of the forward/reverse DNS lookups, you need to configure the eG agent with the full path to the folder containing the HMR test pack. By default, the **HEALTH MONITOR TEST PATH** is set to *default*. This implies that the eG agent runs the **HMR** test from its default location, which is: *C:\Progra~1\Citrix\HealthMon\Tests\Citrix*. However, if the **HMR** test pack is available in a different location in your Citrix environment, then indicate that location in the **HEALTH MONITOR TEST PATH** text box. For instance, your specification can be: *C:\LocalDir\Citrix\HealthMon\Tests\Citrix*.
5. **REVERSE LOOKUP ENABLED** - By default, this flag is set to **No**. This implies that the test will not report the status of the reverse DNS lookup by default. To enable the test to perform reverse DNS lookup and report its success/failure, set this flag to **Yes**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Forward lookup status:	Indicates whether the forward lookup of the IP address from the local DNS is successful or not.		This measure reports a value Success if the IP address lookup is successful and reports a value Failure if the lookup is not successful. The value Failure may also indicate that the returned IP address does not match with that of the IP address that is registered locally. The numeric values that correspond to the above-mentioned states are as follows:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>State</th><th>Numeric Value</th></tr>
<tr><td>Success</td><td>0</td></tr>
<tr><td>Failure</td><td>1</td></tr>
</table>
Note:

By default, this measure reports the above- mentioned states while indicating whether the forward lookup of the IP address is successful or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.
</td></tr>
<tr><td>Reverse lookup status:</td><td>Indicates whether the reverse lookup of the IP address is successful or not.</td><td></td><td>
This measure reports a value Success if the reverse lookup of the IP address is successful and reports a value Failure if the reverse lookup is not successful.

The numeric values that correspond to the above-mentioned states are as follows:

<table>
<tr><th>State</th><th>Numeric Value</th></tr>
<tr><td>Success</td><td>0</td></tr>
<tr><td>Failure</td><td>1</td></tr>
</table>
Note:

By default, this measure reports the above- mentioned states while
</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			indicating whether the reverse lookup of the IP address is successful or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.

3.5.2 Local Host Cache Status Test

Each XenApp server stores a subset of the data store in the Local Host Cache (LHC). The LHC performs two primary functions:

- Permits a server to function in the absence of a connection to the data store.
- Improves performance by caching information used by ICA Clients for enumeration and application resolution.

The following information is contained in the local host cache:

- All servers in the farm, and their basic information.
- All applications published within the farm and their properties.
- All Windows network domain trust relationships within the farm.
- All information specific to itself. (product code, SNMP settings, licensing information)

This test checks for **data consistency (duplicate values) and integrity (corrupt entries) of the XenApp server's local host cache**.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **HEALTH MONITOR TEST PATH** - Citrix XenApp is bundled with a **Health Monitoring and Recovery** (HMR) test pack, which provides a standard set of tests that can be configured to monitor the health of many XenApp components and report failures. Since the eG agent runs one of the HMR tests to determine the health of the local host cache (LHC), you need to configure the eG agent with the full path to the folder containing the HMR test pack. By default, the **HEALTH MONITOR TEST PATH** is set to *default*. This implies that the eG agent runs the HMR test from its default location, which is: *C:\Progra~1\Citrix\HealthMon\Tests\Citrix*. However, if the HMR test pack is available in a different location in your Citrix environment, then indicate that location in the **HEALTH MONITOR TEST PATH** text box. For instance, your specification can be: *C:\LocalDir\Citrix\HealthMon\Tests\Citrix*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
LHC initialized status:	Indicates whether the local host cache is initialized or not.		This measure reports a value Success if the local host cache is initialized successfully and reports a value Failure if the local host cache initialization is not successful. The numeric values that correspond to the above-mentioned states are as follows: State: Success – Numeric Value: 0 State: Failure – Numeric Value: 1 **Note:** By default, this measure reports the above- mentioned states while indicating whether the local host cache initialization is successful or not. However, the graph of this measure will represent success and

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>failure using the numeric equivalents- i.e., 0 and 1 - only.</td></tr>
<tr><td>LHC entry's integrity status:</td><td>Indicates whether the LHC entry is integrated successfully or not.</td><td></td><td>This measure reports the value Success if the LHC entry is integrated successfully and reports the value Failure if the LHC entry integration is not successful. Typically, this measure will report the value Failure if one/more corrupt entries are found in the local host cache. The only way you can fix a corruption in the local host cache is by deleting and recreating the local host cache file (which is an MS Access file).

The numeric values that correspond to the above-mentioned states are as follows:

<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Success</td><td>0</td></tr><tr><td>Failure</td><td>1</td></tr></table>
Note:

By default, this measure reports the above- mentioned states while indicating whether the LHC entry is integrated successfully or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>LHC context nodes status:</td><td>Indicates the health of the context nodes.</td><td></td><td>This measure reports the value Success if the health of the context nodes is good and reports the value Failure if the health of the context nodes is not good.

The numeric values that correspond to the above-mentioned states are as follows:

<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Success</td><td>0</td></tr><tr><td>Failure</td><td>1</td></tr></table>
Note:

By default, this measure reports the above- mentioned states while indicating whether the context nodes are healthy or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.</td></tr>
</table>

3.5.3 XML Thread Health Test

This test monitors the number of worker threads that are currently running on the Citrix XML service and alerts the administrator when the Citrix XML service is overloaded.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **HEALTH MONITOR TEST PATH** - Citrix XenApp is bundled with a **Health Monitoring and Recovery** (HMR) test pack, which provides a standard set of tests that can be configured to monitor the health of many XenApp components and report failures. Since the eG agent runs one of the HMR tests to monitor the load on the Citrix XML service, you need to configure the eG agent with the full path to the folder containing the HMR test pack. By default, the **HEALTH MONITOR TEST PATH** is set to *default*. This implies that the eG agent runs the HMR test from its default location, which is: *C:\Progra~1\Citrix\HealthMon\Tests\Citrix*. However, if the HMR test pack is available in a different location in your Citrix environment, then indicate that location in the **HEALTH MONITOR TEST PATH** text box. For instance, your specification can be: *C:\LocalDir\Citrix\HealthMon\Tests\Citrix*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Number of XML threads:	Indicates the number of worker threads that are running in the Citrix XML service.	Number	By default, the threshold limit for the number of working threads that are running in this Citrix XML service is set to 15. If this threshold value is violated, it indicates that the Web Interface/PN Agent connections would suffer. This measure would therefore be a good indicator to the administrator to identify the overload and rectify the same.

3.5.4 IMA Service Health Test

This test queries the Citrix IMA service and figures out whether the Citrix IMA service is running properly by enumerating the number of applications that are deployed in this Citrix XenApp server.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **HEALTH MONITOR TEST PATH** - Citrix XenApp is bundled with a **Health Monitoring and Recovery** (HMR) test pack, which provides a standard set of tests that can be configured to monitor the health of many XenApp components and report failures. Since the eG agent runs one of the HMR tests to determine the status of the Citrix IMA service, you need to configure the eG agent with the full path to the folder containing the HMR test pack. By default, the **HEALTH MONITOR TEST PATH** is set to *default*. This implies that the eG agent runs the HMR test from its default location, which is: *C:\Progra~1\Citrix\HealthMon\Tests\Citrix*. However, if the HMR test pack is available in a different location in your Citrix environment, then indicate that location in the **HEALTH MONITOR TEST PATH** text box. For instance, your specification can be: *C:\LocalDir\Citrix\HealthMon\Tests\Citrix*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Status:	Indicates the current health status of the Citrix IMA service.		This measure reports the value *Success* if the Citrix IMA service is in good health, and reports the value *Failure* if the Citrix IMA service is not operating properly. The numeric values that correspond to the above-mentioned states are as follows: State: Success – Numeric Value: 0 State: Failure – Numeric Value: 1

Measurement	Description	Measurement Unit	Interpretation
			Note: By default, this measure reports the above- mentioned states while indicating whether the Citrix IMA service is in good health or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.
Number of applications:	Indicates the number of applications that were deployed in this Citrix XenApp server.	Number	

3.5.5 Print Manager Health Test

The Citrix Print Manager Service manages the creation of printers and driver usage within the XenApp sessions.

This test reports the health of the Citrix Print Manager Service by enumerating the number of local session printers.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **HEALTH MONITOR TEST PATH** - Citrix XenApp is bundled with a **Health Monitoring and Recovery** (HMR) test pack, which provides a standard set of tests that can be configured to

monitor the health of many XenApp components and report failures. Since the eG agent runs one of the HMR tests to report the health of the Citrix Print Manager service, you need to configure the eG agent with the full path to the folder containing the HMR test pack. By default, the **HEALTH MONITOR TEST PATH** is set to *default*. This implies that the eG agent runs the HMR test from its default location, which is: *C:\Progra~1\Citrix\HealthMon\Tests\Citrix*. However, if the HMR test pack is available in a different location in your Citrix environment, then indicate that location in the **HEALTH MONITOR TEST PATH** text box. For instance, your specification can be: *C:\LocalDir\Citrix\HealthMon\Tests\Citrix*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Status:	Indicates the current health status of the Citrix Print Manager Service.		This measure reports the value Success if the Citrix Print Manager service is in good health, and reports the value Failure if the Citrix Print Manager service is not operating properly. The numeric values that correspond to the above-mentioned states are as follows: State: Success = 0; Failure = 1 **Note:** By default, this measure reports the above- mentioned states while indicating whether the Citrix Print Manager service is in good health or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.

3.5.6 Ticket Request Status Test

Once a user logs in to the Citrix web interface, he/she receives a list of applications to which they have access. When the user chooses one of the applications to open, the request is received by the web interface and forwarded to the local XML service. The XML service then asks the IMA service for the IP address of the least busy server that has the requested application published on it. The IMA service may have to contact the data collector for this information. In turn, the IMA service on the least loaded server contacts the terminal services on this system to obtain a ticket which provides the user with the permission to access the requested application.

This test reports the health of the Citrix XML Service by generating the requested ticket.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **HEALTH MONITOR TEST PATH** - Citrix XenApp is bundled with a **Health Monitoring and Recovery** (HMR) test pack, which provides a standard set of tests that can be configured to monitor the health of many XenApp components and report failures. Since the eG agent runs one of the HMR tests to check whether the Citrix server could obtain a ticket or not, you need to configure the eG agent with the full path to the folder containing the HMR test pack. By default, the **HEALTH MONITOR TEST PATH** is set to *default*. This implies that the eG agent runs the HMR test from its default location, which is: *C:\Progra~1\Citrix\HealthMon\Tests\Citrix*. However, if the HMR test pack is available in a different location in your Citrix environment, then indicate that location in the **HEALTH MONITOR TEST PATH** text box. For instance, your specification can be: *C:\LocalDir\Citrix\HealthMon\Tests\Citrix*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Status:	Indicates whether the Citrix server could obtain a ticket or not.		This measure reports the value Success if the Citrix server could obtain a ticket, and reports the value Failure if the server was denied a ticket. The numeric values that correspond to the above-mentioned states are as follows: State: Success = 0; Failure = 1 **Note:** By default, this measure reports the above- mentioned states while indicating whether the Citrix server could obtain a ticket or not. However, the graph of this measure will represent success and failure using the numeric equivalents - i.e., 0 and 1 - only.

3.5.7 Print Spooler Health Test

This test reports the health of the Microsoft Print Spooler by enumerating the printers that are available on the local server. This test additionally enumerates the available print drivers and the print processors. This test helps you to determine if there are any system printer issues that are to be addressed.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **HEALTH MONITOR TEST PATH** - Citrix XenApp is bundled with a **Health Monitoring and Recovery** (HMR) test pack, which provides a standard set of tests that can be configured to monitor the health of many XenApp components and report failures. Since the eG agent runs one of the HMR tests to determine the status of the Microsoft Print Spooler, you need to configure the eG agent with the full path to the folder containing the HMR test pack. By default, the **HEALTH MONITOR TEST PATH** is set to *default*. This implies that the eG agent runs the HMR test from its default location, which is: *C:\Progra~1\Citrix\HealthMon\Tests\Citrix*. However, if the HMR test pack is available in a different location in your Citrix environment, then indicate that location in the **HEALTH MONITOR TEST PATH** text box. For instance, your specification can be: *C:\LocalDir\Citrix\HealthMon\Tests\Citrix*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Printer status:	Indicates the health of the printer by enumerating the printers on the local server.		This measure reports the value Success if the printers on the local server are enumerated successfully, and reports the value Failure if the enumeration is unsuccessful. The numeric values that correspond to the above-mentioned states are as follows: State: Success = 0; Failure = 1 **Note:** By default, this measure reports the

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>above- mentioned states while indicating whether the printers on the local server are successfully enumerated or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.</td></tr>
<tr><td>Printer processors status:</td><td>Indicates the health of the printer processors by enumerating the printer processors.</td><td></td><td>This measure reports the value Success if the printer processors are enumerated successfully, and reports the value Failure if the enumeration is unsuccessful.

The numeric values that correspond to the above-mentioned states are as follows:

<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Success</td><td>0</td></tr><tr><td>Failure</td><td>1</td></tr></table>
Note:

By default, this measure reports the above- mentioned states while indicating whether the printer processors could be successfully enumerated or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.</td></tr>
<tr><td>Printer drivers status:</td><td>Indicates the health of the printer drivers by enumerating them.</td><td></td><td>This measure reports the value Success if the printer drivers are enumerated successfully, and</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>reports the value Failure if the enumeration is unsuccessful.

The numeric values that correspond to the above-mentioned states are as follows:

<table><tr><th>State</th><th>Numeric Value</th></tr><tr><td>Success</td><td>0</td></tr><tr><td>Failure</td><td>1</td></tr></table>
Note:

By default, this measure reports the above- mentioned states while indicating whether the printer drivers could be successfully enumerated or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.</td></tr>
</table>

3.5.8 Terminal Service Health Test

This test reports the health of the Terminal service by enumerating the list of all local RDP and ICA sessions running on the server. For each session, this test enumerates the session information such as user name, session state, logon times etc., The number of sessions established on the local server impacts the response time of this test.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **HEALTH MONITOR TEST PATH** - Citrix XenApp is bundled with a **Health Monitoring and Recovery** (HMR) test pack, which provides a standard set of tests that can be configured to monitor the health of many XenApp components and report failures. Since the eG agent runs one of the HMR tests to report the health of the Terminal service, you need to configure the eG agent with the full path to the folder containing the HMR test pack. By default, the **HEALTH MONITOR TEST PATH** is set to *default*. This implies that the eG agent runs the HMR test from its default location, which is: *C:\Progra~1\Citrix\HealthMon\Tests\Citrix*. However, if the HMR test pack is available in a different location in your Citrix environment, then indicate that location in the **HEALTH MONITOR TEST PATH** text box. For instance, your specification can be: *C:\LocalDir\Citrix\HealthMon\Tests\Citrix*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Status:	Indicates the health of the Terminal Service by enumerating the list of all local RDP and ICA sessions in the local server.		This measure reports the value Success if the health of the Terminal service is good and reports the value Failure if the Terminal service fails to enumerate the local RDP and ICA sessions on the local server. The numeric values that correspond to the above-mentioned states are as follows: State: Success = 0; Failure = 1 **Note:**

Measurement	Description	Measurement Unit	Interpretation
			By default, this measure reports the above- mentioned states while indicating whether the Terminal service is in good health or not. However, the graph of this measure will represent success and failure using the numeric equivalents- i.e., 0 and 1 - only.

3.5.9 Citrix Connection Test

This test performs an application-level ping to the Citrix server and measures the response from the server.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **SERVERIP** - The CtxConnectionTest performs an application-level ping to a Citrix server, and measures the response from the server. The IP address of that Citrix server has to be specified in the **SERVERIP** text box. By default, the IP of the **HOST** will be displayed here. This means that, by default, the Citrix **HOST** will try to ping its own self.
5. **COUNT** - Specify the number of packets to be sent by the test.

Measurements made by the test

Measurement	Desciption	Measurement Unit	Interpretation
Connection availability:	Indicates the availability of the Citrix server	Percent	A value of 100 % indicates that the Citrix server is responding to requests. 0 indicates that the server is not responding. A server might not respond if it is not up and running or if it is overloaded.
Packet loss on Citrix connection:	Indicates the percentage of packets sent that were replied by the server	Percent	While 0 indicates that the server is responding to requests, any value greater than 0 could indicate that the server is not able to keep up with its current load.
Avg Citrix connection time:	Response time is the time from packet transmission to reception. Average response time measures the average value of the response time based on replies returned by the server.	Secs	Increase in the average response time indicates slow-down of the server and potential issues in handling user requests by the server.
Max Citrix connection time:	This is the maximum of response times based on replies returned by the server.	Secs	If this value is consistently different from the average response time, further investigation of other server metrics may be necessary.

3.5.10 Citrix Server VDA Status Test

The Virtual Delivery Agent (VDA) enables connections to applications and desktops. The VDA is installed on the Citrix XenApp server that runs the applications or virtual desktops for the user. It enables the 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 Citrix XenApp server. The target Citrix XenApp server would therefore become an unusable resource. The VDA issues with respect to registration are logged in the event log of the target Citrix XenApp 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 Server VDA Status** test helps administrators to figure out which service has failed leading to VDA registration issues!

This test monitors the Virtual Delivery Agent installed on the target Citrix XenApp server 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.

Target of the test : Any Citrix server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the Citrix XenApp server monitored

Configurable parameters for the test

Parameters	Description
Test Period	How often should the test be executed. By default, this is 15 minutes.
Host	The host for which the test is to be configured.
Port	Refers to the port used by the Citrix server .
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.
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

Parameters	Description
	• 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 machine on which the VDA was installed.		The values that this measure can report and its corresponding numeric equivalents are listed in the table below: **Measure Value** / **Numeric Value** Failed: 0 Success: 1 **Note:** By default, this measure reports the **Measure Value**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.
User authentication status	Indicates the current status of the User authentication service.		The values that this measure can report and its corresponding numeric equivalents are listed in the table below: **Measure Value** / **Numeric Value** Failed: 0 Success: 1 **Note:** By default, this measure reports the **Measure Value**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

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>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 Measure Values 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>Note:
By default, this measure reports the Measure Values 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:</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>
Note:
By default, this measure reports the Measure Values 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>
Note:
By default, this measure reports the Measure Values 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>
Note:
</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			By default, this measure reports the **Measure Value**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.
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.

3.5.11 Citrix Authentication Test

This test emulates a user login process at the system level on a XenApp server and reports whether the login succeeded and how long it took.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every user account being checked

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 Citrix server
4. **USER** - This test emulates a user login process at the system level on a XenApp server. Therefore, specify the login name of a user with **interactive logon** and **logon locally privileges**.
5. **PASSWORD** - Enter the password that corresponds to the specified **USER**name.

6. **CONFIRM PASSWORD** – Confirm the specified **PASSWORD** by retyping it here.
7. **DOMAIN** - Specify the name of the domain to which the test will try to login. If the test is to login to a local host, specify 'none' here.

   Note:

   If users are spread across multiple domains, then, you can configure this test with multiple **DOMAIN** specifications; in this case, for every **DOMAIN**, a **USER-PASSWORD** pair might also have to be configured. Sometimes, you might want the test to login as specific users from the same domain, to check how long each user login takes. Both these scenarios require the configuration of multiple **DOMAIN**s and/ormultiple **USER** names and **PASSWORD**s. In order to enable users to specify these details with ease, eG Enterprise provides a special page; to access this page, click on the **Click here** hyperlink at the top of the parameters in the test configuration page. To know how to use this page, refer to Section **3.5.11.1** of this document.
8. **REPORT BY DOMAIN** - By default, this flag is set to **Yes**. This implies that by default, this test will report metrics for every domainname\username configured for this test. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the TEST to report metrics for the username alone, then set this flag to **No**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Availability:	Indicates whether the login was successful or not	Percent	A value of 100 % indicates that the login has succeeded. The value 0 is indicative of a failed login.
Authentication time:	Indicates the time it took to login	Secs	If this value is very high then it could be owing to a configuration issue (i.e the domain might not be configured properly) or a slow-down/unavailability of the primary domain server.

3.5.11.1 Configuring Multiple Users for the Citrix Authentication Test

Administrators of multi-domain environments might want to configure the Citrix Authentication test to emulate user logins from multiple **DOMAIN**s; in this case, for every **DOMAIN**, a **USER-PASSWORD** pair might have to be configured. In some other cases, administrators might want the test to login as

specific users from the same domain, to check how long each user login takes. Both these scenarios require the configuration of multiple **DOMAIN**s and/or multiple **USER** names and **PASSWORD**s. In order to enable users to specify these details with ease, eG Enterprise provides a special page; to access this page, click on the Click here hyperlink at the top of the parameters in the CitrixAuthentication test configuration page (see Figure 3.8).

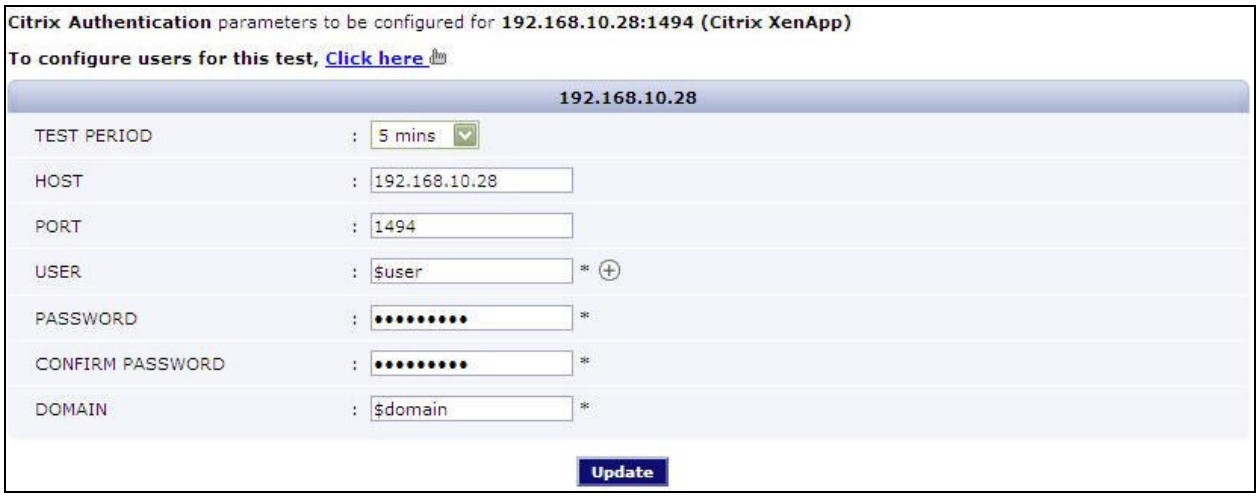


Figure 3.8: Configuring the Citrix Authentication Test

Upon clicking, Figure 3.9 will appear, using which the user details can be configured.

Figure 3.9: The Citrix Authentication test user configuration page

To add a user specification, do the following:

1. First, provide the name of the **Domain** from which logins are to be emulated (see Figure 3.9). If you are trying to login to a local host, then, specify *none* here.

2. The eG agent must then be configured with the credentials of a user with **interactive logon** and **logon locally privileges** in the specified **Domain** or local host. Provide the user credentials in the **User** and **Password** text boxes in Figure 3.9, and confirm the password by retyping it in the **Confirm Password** text box.

3. To add more users, click on the ⊕ button in Figure 3.9. This will allow you to add one more user specification as depicted by Figure 3.10.

Figure 3.10: Adding another user

4. Sometimes, you might want the CitrixAuthentication test to emulate logins from a single domain but as multiple users in that domain. For instance, you might want the test to login as user *eglabuser* and as user *labadmin* from the same egitlab domain. You can configure the eG agent with the credentials of both these users as shown by Figure 3.11.

Figure 3.11: Associating a single domain with different admin users

5. To clear all the user specifications, simply click the **Clear** button in Figure 3.11.
6. To remove the details of a particular user alone, just click the ⊖ button corresponding to that user specification in Figure 3.11.
7. To save the specification, just click on the **Update** button in Figure 3.11. This will lead you back to the test configuration page, where you will find the multiple domain names, user names, and passwords listed against the respective fields (see Figure 3.12).

Figure 3.12: The test configuration page displaying multiple domain names, user names, and passwords

3.5.12 Citrix Authentication Test

This test emulates a user login process at the system level on a XenApp server and reports whether the login succeeded and how long it took.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every user account being checked

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 Citrix server
4. **USER** - This test emulates a user login process at the system level on a XenApp server. Therefore, specify the login name of a user with **interactive logon** and **logon locally privileges**.
5. **PASSWORD** - Enter the password that corresponds to the specified **USER**name.
6. **CONFIRM PASSWORD** – Confirm the specified **PASSWORD** by retyping it here.
7. **DOMAIN** - Specify the name of the domain to which the test will try to login. If the test is to login to a local host, specify 'none' here.

Note:

If users are spread across multiple domains, then, you can configure this test with multiple **DOMAIN** specifications; in this case, for every **DOMAIN**, a **USER-PASSWORD** pair might also have to be configured. Sometimes, you might want the test to login as specific users from the same domain, to check how long each user login takes. Both these scenarios require the configuration of multiple **DOMAIN**s and/ormultiple **USER** names and **PASSWORD**s. In order to enable users to specify these details with ease, eG Enterprise provides a special page; to access this page, click on the **Click here** hyperlink at the top of the parameters in the test configuration page. To know how to use this page, refer to Section **3.5.12.1** of this document.

8. **REPORT BY DOMAIN** - By default, this flag is set to **Yes**. This implies that by default, this test will report metrics for every domainname\username configured for this test. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the TEST to report metrics for the username alone, then set this flag to **No**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Availability:	Indicates whether the login was successful or not	Percent	A value of 100 % indicates that the login has succeeded. The value 0 is indicative of a failed login.
Authentication time:	Indicates the time it took to login	Secs	If this value is very high then it could be owing to a configuration issue (i.e the domain might not be configured properly) or a slow-down/unavailability of the primary domain server.

3.5.12.1 Configuring Multiple Users for the Citrix Authentication Test

Administrators of multi-domain environments might want to configure the Citrix Authentication test to emulate user logins from multiple **DOMAIN**s; in this case, for every **DOMAIN**, a **USER-PASSWORD** pair might have to be configured. In some other cases, administrators might want the test to login as specific users from the same domain, to check how long each user login takes. Both these scenarios require the configuration of multiple **DOMAIN**s and/or multiple **USER** names and **PASSWORD**s. In order to enable users to specify these details with ease, eG Enterprise provides a special page; to access this page, click on the Click here hyperlink at the top of the parameters in the CitrixAuthentication test configuration page (see Figure 3.13).

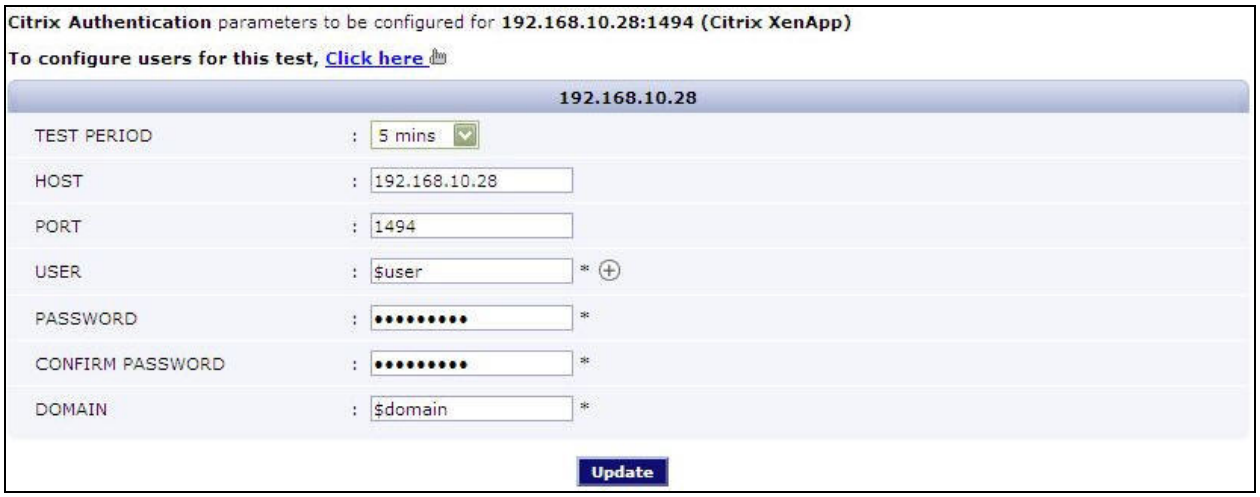


Figure 3.13: Configuring the Citrix Authentication Test

Upon clicking, Figure 3.14 will appear, using which the user details can be configured.

Figure 3.14: The Citrix Authentication test user configuration page

To add a user specification, do the following:

1. First, provide the name of the **Domain** from which logins are to be emulated (see Figure 3.14). If you are trying to login to a local host, then, specify *none* here.
2. The eG agent must then be configured with the credentials of a user with **interactive logon** and **logon locally privileges** in the specified **Domain** or local host. Provide the user credentials in the **User** and **Password** text boxes in Figure 3.14, and confirm the password by retyping it in the **Confirm Password** text box.
3. To add more users, click on the ⊕ button in Figure 3.14. This will allow you to add one more user specification as depicted by Figure 3.15.

Figure 3.15: Adding another user

4. Sometimes, you might want the CitrixAuthentication test to emulate logins from a single domain but as multiple users in that domain. For instance, you might want the test to login as user *eglabuser* and as user *labadmin* from the same egitlab domain. You can configure the eG agent with the credentials of both these users as shown by Figure 3.16.

Figure 3.16: Associating a single domain with different admin users

5. To clear all the user specifications, simply click the **Clear** button in Figure 3.16.
6. To remove the details of a particular user alone, just click the ⊖ button corresponding to that user specification in Figure 3.16.
7. To save the specification, just click on the **Update** button in Figure 3.16. This will lead you back to the test configuration page, where you will find the multiple domain names, user names, and passwords listed against the respective fields (see Figure 3.17).

Figure 3.17: The test configuration page displaying multiple domain names, user names, and passwords

3.5.13 Citrix Enumerations Test

This test reports the number of filtered application enumerations per second.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every Citrix 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 used by the Citrix server
4. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Filtered application enumerations:	Indicates the number of WI logons/ application enumerations handled by an XML Broker per second.	Enums/Sec	The value of this measure enables administrators to accurately assess the impact of growth / stress on the XML brokers and zone data collectors.

3.5.14 Citrix IMA Test

This test reports various statistics relating to the Citrix Independent Management Architecture (IMA). Citrix IMA is an architectural model and a protocol for server to server communications. IMA includes a collection of subsystems that define and control the execution of Citrix products. The functions enabled by IMA include:

- Central administration of all the Citrix servers
- Central license management and pooling without license gateways
- Centralized data store for all Citrix configurations
- Auditing of administration activities, etc.

This test reports the IMA-related communications from this Citrix server to other Citrix servers. One set of results is reported for each server to server communication.

Target of the test : Any Citrix MetaFrame XP server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on

the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Data received rate:	Represents the rate at which data is received by the server from another Citrix server in the farm.	KBytes/sec	Evaluate the IMA traffic periodically to explore alternative configurations (e.g., splitting a farm) to minimize network overheads. The IMA traffic between servers can be high if the indirect mode of data store access is used - in this case, only one server in the farm directly accesses the data store. All other servers rely on this server to access the data store
Data transmit rate:	Represents the rate at which IMA data is sent by a server to another server in the farm.	KBytes/sec	
Network connections:	Number of active IMA network connections from a server to another IMA server.	Number	

3.5.15 Citrix Server Test

This test generates statistics relating to a Citrix server.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every Citrix 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 used by the Citrix server
4. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Application enumerations:	Represents the number of application enumerations per second	Enums/Sec	The Citrix Program Neighborhood allows a user to get a listing of all available applications published in the farm. This enumeration of resources takes place automatically every time the user launches the Citrix Program Neighborhood. This metric reflects the rate of application enumerations. An unusually high number of

Measurement	Description	Measurement Unit	Interpretation
			numerations can slow down a Citrix server.
Application resolutions:	Represents the number of application resolutions per second	Resolutions/sec	When the user clicks the link to a published application, the link is resolved to an application. This metric reflects the workload on the server in terms of application accesses. The rate of application resolutions depends on the number of users connecting to the farm, duration for which the average user stays logged on, and the number of published applications. If the rate of application resolutions is excessively high, consider creating multiple zones in the farm to reduce the load on the data collector.
Datastore connection failure:	Indicates how long the XenApp server was disconnected from the datastore.	Mins	The data store of the XenApp server hosts centralized configuration data for a server farm. The data store is critical for central administration of the server farm. Hence, any loss of communication between a XenApp servers and its data store can result in inconsistencies in the configuration data. A high value of this measure is hence a cause for concern as it indicates that the XenApp server has been disconnected from the datastore for a long time.

Measurement	Description	Measurement Unit	Interpretation
Datastore reads:	The rate of data read from the IMA data store	KBytes/Sec	This metric reports the workload on the data store. Since it is a central repository for a farm, slowdown of the data store can impact the performance of the farm. Data store traffic is usually high during server startup.
Datastore writes:	The rate of data written into the IMA data store	KBytes/Sec	This metric reports the workload on the data store. Since it is a central repository for a farm, slowdown of the data store can impact the performance of the farm.
Dynamic store reads:	The rate of data read from the IMA Dynamic store	KBytes/Sec	The dynamic store maintains information that changes frequently such as current sessions, disconnected sessions, server load, etc. This metric denotes the read rate of data from the dynamic store.
Dynamic store writes:	The rate of data written into the IMA Dynamic store	KBytes/Sec	The dynamic store maintains information that changes frequently such as current sessions, disconnected sessions, server load, etc. This metric denotes the rate at which data is written to the dynamic store.
LH cache reads:	The rate of data read from the IMA Local Host Cache	KBytes/Sec	Each server has a subset of the data store called the local host cache. The local host cache performs two functions: • It permits the server to function in the absence of a connection to the data store.

Measurement	Description	Measurement Unit	Interpretation
			• Improves performance by caching information used by ICA clients for enumeration and application resolution. The larger the cache, greater the hits to the cache and fewer data store accesses. Comparing the read rate from the local host cache and the data store, the administrator can assess the cache efficiency.
LH cache writes:	The rate of data written into the IMA Local Host Cache written/sec	KBytes/Sec	
Zone elections:	Indicates the number of zone elections that have occurred	Number	Zones in a Citrix farm serve two purposes - (a) to collect data from member servers in a hierarchical structure; (b) efficiently distribute changes to all servers in the farm. The first server in a farm is the data collector of the farm by default. Elections within a zone are used to determine the data collector for the zone. Frequent zone elections in a zone can result in increased network traffic.
Zone elections won:	Indicates the number of times a Citrix server has won a zone election	Number	

3.5.16 Citrix License Test

The Citrix server supports two types of licenses- a product license and a connection license. The product license is a license to run a particular kind of Citrix product on a server. A server farm must

have a product license with one license count to run Citrix server software on each server in the server farm. The Citrix XenApp servers allocates product licenses from a pool of available licenses for a XenApp server farm.

A connection license is a license for client connections to Citrix servers. A server farm must have a connection license with one license count for each concurrent client connection to the Citrix servers in the farm.

This test reports the usage of both the connection and product licenses by the Citrix server.

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 : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **REREADLICENSE** – If this flag is set to **Yes**, then the eG agent will check for changes in license status everytime the test runs. If this flag is set to **No**, then the eG agent will not check for license changes.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Pool licenses in use:	All the Citrix servers in a server farm typically share a pool of licenses. This measure reports the number of licenses from the pool used by	Number	

Measurement	Description	Measurement Unit	Interpretation
	the current server.		
Assigned licenses:	Citrix allows a number of licenses from the pool to be assigned to a specific server. No other server can re-use these assigned licenses. This measure reports the number of licenses that are assigned to the current server.	Number	
Assigned licenses in use:	This reports the number of assigned licenses in use.	Number	If the number of assigned licenses in use is much lower than the allocated number of assigned licenses, the administrator may want to reduce the number of assigned licenses for this server.
Usage of assigned licenses:	This reports the % of assigned licenses in use.	Percent	Administrators may choose to be alerted when the assigned license usage reaches close to 100%, so that they may increase the number of assigned licenses if desired.

3.5.17 Citrix License Stats Test

This test shows the statistics of the license server while it is being accessed by the Citrix XenApp server.

Target of the test : Citrix XenApp server 4.0 and above

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Avg license checkin response time:	Indicates the average license check- in response time.	Secs	
Avg checkout response time:	Indicates the average license check- out response time.	Secs	
Last recorded checkin time:	Indicates the last recorded license check-in response time.	Secs	
Last recorded checkout time:	Indicates the last recorded license check-out response time.	Secs	
License server connection failure:	Indicates the duration for which the Citrix XenApp server was disconnected from the License server.	Mins	Any value greater than 0 implies that the Citrix XenApp server is having trouble connecting to the license server.

3.5.18 Citrix Data Store Test

The CitrixDataStore test monitors the Citrix XenApp server's datastore.

Target of the test : Citrix XenApp server 4.0 and above

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Errors found:	Indicates whether any errors have occurred in the datastore or not.	Number	While the value 1 indicates the existence of errors in the datastore, the value 0 indicates that no errors have occurred in the datastore.
Application errors:	Indicates the number of	Number	

Measurement	Description	Measurement Unit	Interpretation
	application errors found in the datastore.		
Groups errors found:	Indicates the number of group errors found in the datastore.	Number	
Server errors found:	Indicates the number of server errors found in the datastore.	Number	

3.5.19 Citrix Dynamic Store Test

This test monitors the Citrix XenApp server's dynamic store.

Target of the test : Citrix XenApp server 4.0 and above

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Gateway update count:	Indicates the number of dynamic store update packets sent to remote data collectors during the last measurement period.	Number	
Gateway update sent:	Indicates the number of bytes of data sent across gateways to remote data collectors during the last measurement period.	KB	
Query count:	Indicates the number of dynamic store queries that have been performed during the last measurement period.	Number	
Query request received:	Indicates the number of bytes of data received in dynamic store query request packets during the last measurement period.	KB	
Query response sent:	Indicates the number of bytes of data sent in response to dynamic store queries during the last measurement period.	KB	
Read rate:	Indicates the rate at which data was read from the IMA Dynamic store during the last measurement period.	Reads/Sec	

Measurement	Description	Measurement Unit	Interpretation
Write rate:	Indicates the rate at which data was written to the IMA Dynamic Store during the last measurement period.	Writes/Sec	
Update requests received:	Indicates the number of bytes of data received in dynamic store update packets during the last measurement period.	KB	
Update packets received:	Indicates the number of update packets received by the dynamic store during the last measurement period.	Number	
Update response sent:	Indicates the number of bytes of data sent in response to dynamic store update packets during the last measurement period.	KB	

3.5.20 Server Work Items Test

This test reports critical statistics related to the status of work items.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Resolution work items currently being executed:	Reports the number of resolution work items that are currently being executed.	Number	
Resolution work items ready for execution:	Indicates the number of resolution work items that are currently ready to be executed.	Number	
Work items pending execution:	Indicates the number of work items that are currently being executed.	Number	
Work items pending execution:	Indicates the current number of work items that are not yet ready to be executed.	Number	
Work items ready for execution:	Indicates the number of work items that are ready to be executed	Number	Attention is needed if this measure is sustained at 2 for one minute.

Measurement	Description	Measurement Unit	Interpretation
	currently by IMA Threads.		

3.5.21 User Profile Test

User profiles are the heart of the Citrix environment. User profiles contain the configuration settings, which bring the user desktop alive. One of the major problems in a server-based computing environment like Citrix is that the user's login process takes more time to open the user's desktop. This happens if the user profile size is huge. The **User Profile** test monitors the size of the Citrix user profiles and raises an alarm if the profile size exceeds the profile quota size.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every user profile on the Citrix 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 Citrix server
4. **PROFILESIZELIMIT** - Specify the profile quota size (in MB). The default value is 50 MB.
5. **EXCLUDE** - Provide a comma-separated list of users who need to be excluded from the analysis. By default, this parameter is set to *All_Users*, indicating that, by default, the test will not monitor the *All_Users* profile.
6. **CURRENTUSERSONLY** - If this is set to **true**, then the profile sizes of only those users who are currently logged into the server will be monitored. If this is set to **false**, eG Enterprise will perform profile monitoring for all the users to the server.
7. **FILESIZELIMIT** - Takes the file quota size (in KB). The default size is 10000 KB.
8. **EXCLUDE FOLDERS** – By default, when this test computes the size of a profile, it automatically excludes the following folders and their sub-folders from the computation: *AppData\Local,AppData\LocalLow,Recycle.Bin,SkyDrive,WorkFolders*. If need be, you can choose to include one/more of these default folders when computing the profile size; for this,

all you need to do is remove those specific folders from the default **EXCLUDE FOLDERS** specification. For example, to include the SkyDrive and WorkFolders folders, simply remove them from the default specification above. Also, if required, you can exclude more folders from the profile size computation, by appending the corresponding folder names / folder name patterns to this default list. For instance, your specification can be: *AppData\Local,AppData\LocalLow,Recycle.Bin,SkyDrive,WorkFolders,*Backup*,Favo*,*Desktop*. In the case of this sample specification, in addition to the default list of excluded folders, all folders with names that embed the string Backup, with names that begin with the strin Favo, and with names that end with the string Desktop, will be excluded from size computation. Moreover, all sub-folders within these folders will also be ignored during size computation.

9. **REPORT BY DOMAIN** – By default, this flag is set to **Yes**. This implies that by default, this test will report metrics for every domainname\username to the server. This way, administrators will be able to quickly determine which user belongs to which domain. If you want the test to report metrics for every username alone, then set this flag to **No**.

10. **USER PROFILE DIR** – By default, this parameter is set to none. This implies that for XenApp/Microsoft RDS servers operating on Windows 2008 and Windows 2012 platforms, the test will, by default, check the C:\Users directory for the user profile files. In some environments, the user profile-related files and folders may exist in a different directory. In such environments, you will have to specify the exact directory in which the user profiles exist, against the **USER PROFILE DIR** parameter.

11. **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
Is user profile exceeding quota?:	Indicates whether the profile size exceeds the profile quota size by comparing the current profile size with the configured **PROFILESIZELIMIT** parameter.	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.	MB	
Number of files in user's profile:	Indicates the number of files available in the user profile.	Number	
Large files in user's profile:	The number of files in the user profile, which exceed the allowable **FILESIZELIMIT** parameter.	Number	The detailed diagnosis of this measure, if enabled, lists all the files that have exceeded the configured filesizelimit.

Use the detailed diagnosis of the *Large files in user's profile* measure to know which files have exceeded the configured **FILESIZELIMIT**. If a profile takes too long to load, then using these diagnositics, administrators can identify the exact file in the profile that could be contributing to loading delay.

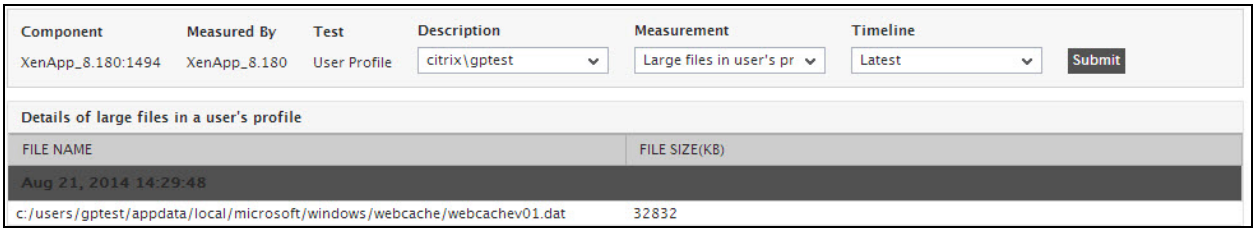


Figure 3.18: The detailed diagnosis of the Large files in user's profile measure

3.5.22 XML Threads Test

This test monitors the usage of XML threads, and reports whether or not the XML service has adequate threads for processing requests.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every Citrix 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 Citrix server
4. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Max XML threads:	Indicates the maximum number of XML threads.	Number	
Busy XML threads:	Indicates the number of units of work the XML service is currently processing.	Number	By default, the maximum number of requests that the XML service can process at any one time is 16. If this measure is sustained at 16 for one minute or longer, it indicates

Measurement	Description	Measurement Unit	Interpretation
			that all the XML threads have been used up and the XML service cannot service any more requests.
Current XML threads:	Indicates the current number of XML threads.	Number	

3.5.23 Windows User Logon Test

The process of a user logging into a Citrix or Microsoft RDS 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 XenApp/Microsoft RDS server at the same time. Hence, if a user complains that he/she is unable to access an application/desktop published on Citrix/Microsoft RDS, 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 2003 is to use the user environment debugging mechanism. To enable this on Windows 2003 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 **Windows User Logon** test periodically checks the userenv log file on Windows 2003 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 : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every user to the Citrix XenApp 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 Citrix server
4. **REPORT TOTAL** – By default, this flag is set to **No**. In this case therefore, the test will only report metrics for every user to the XenApp server. If this flag is set to **Yes**, then the test will report metrics for a Total descriptor – the metrics reported by this descriptor will be aggregated across all users to the XenApp server. This way, XenApp administrators will receive a system-wide overview of the health of the profile loading/unloading process.
5. **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 XenApp server. If you set this flag to **No**, then make sure that the **REPORT TOTAL FLAG** is set to '**Yes**'. Because, if both the **REPORT FOR EACH USER** and the **REPORT TOTAL** flags are set to **No, then the test will not run!** On the other hand, if only the **REPORT TOTAL** flag is set to **Yes**, the test will only report metrics for the Total descriptor. Moreover, if both the **REPORT TOTAL** and the **REPORT FOR EACH USER** flags are set to **Yes**, then the test will report metrics per user and will additionally report metrics for the Total descriptor as well.
6. **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.

7. **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.
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 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

Measurement	Description	Measurement Unit	Interpretation
			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? This measure will not be available for Citrix XenApp Servers operating on Windows 2003.
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. This measure will not be available for Citrix XenApp Servers operating on Windows 2003.
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. This measure will not be available for Citrix XenApp Servers operating on Windows 2003.
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.	MSecs	Compare the value of this measure across users to know which user's logon process spent maximum time in group policy processing. If this measure reports an unusually high value for any user, then, you

Measurement	Description	Measurement Unit	Interpretation
			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. This measure will not be available for Citrix XenApp Servers operating on Windows 2003.
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. This measure will not be available for Citrix XenApp Servers operating on Windows 2003.
Total group policy object file	Indicates the amount of time the logon process	MSecs	Compare the value of this measure across users to know which user's

Measurement	Description	Measurement Unit	Interpretation
accessed tme:	took to access group policy object files for this user during the last measurement period.		logon process spent maximum time in accessing the group policy object file. This measure will not be available for Citrix XenApp Servers operating on Windows 2003.
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

Measurement	Description	Measurement Unit	Interpretation
			detailed diagnosis of this measure. This will accurately pinpoint the session in which the profile took the longest to load. This measure will not be available for Citrix XenApp Servers operating on Windows 2003.
Profile load starts:	Indicates the number of times this user's profile was loaded in the last measurement period.	Number	This metric gives an idea of the rate at which users are logging in to the server.
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	This measure will be available only for Citrix XenApp servers operating on Windows 2003.

Measurement	Description	Measurement Unit	Interpretation
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.
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

Measurement	Description	Measurement Unit	Interpretation
			conclude that it is indeed the size of the profile which is affecting the profile load time. This measure will be available only for Citrix XenApp servers operating on Windows 2003.
Max profile load time:	Indicates the maximum time it took to load a profile during the last measurement period.	Msecs	This measure will be available only for Citrix XenApp servers operating on Windows 2003.
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 measure to know when a user's session was initiated and how long each session remained active on the XenApp server. From this, you can infer how many sessions were 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	

Measurement	Description	Measurement Unit	Interpretation
Avg user profile unload time:	Indicates the average time for unloading a profile during the last measurement period.	Msecs	This measure will be available only for Citrix XenApp servers operating on Windows 2003.
Max profile unload time:	Indicates the maximum time for unloading a profile during the last measurement period.	Msecs	This measure will be available only for Citrix XenApp servers operating on Windows 2003.
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 in the last measurement period.	Number	Compare the total number of starts to completions. if there is a significant discrepancy, this 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	

Note:

As stated earlier, the user logon process includes a series of steps – eg., domain discovery, authentication, GPO application, profile loading, etc. - that culminate in a user gaining access to an application deployed on a XenApp server. These individual steps may not always occur in sequence – i.e., one after another; in fact

usually, they occur parallelly. This is why, the value of the *Logon duration* measure will not be an aggregate of the time values reported by the other metrics of the **User Logon** test.

You can use the detailed diagnosis of the *Client side extension processed time* measure to know which client side extension was used to process which group policy for a particular user.

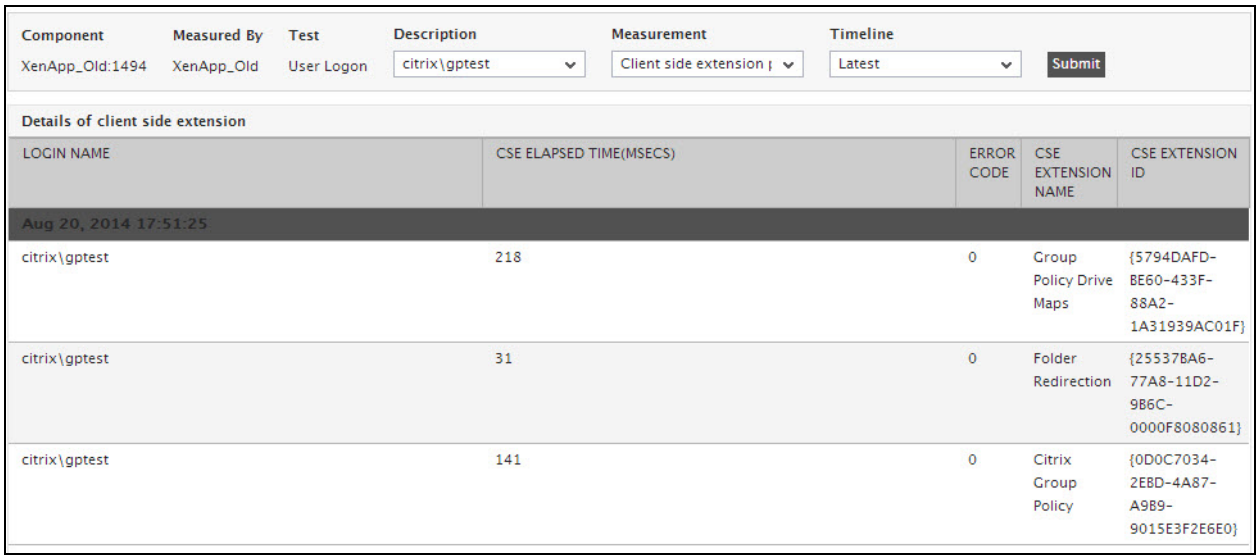


Figure 3.19: The detailed diagnosis of the Client side extension processed time measure

Using the detailed diagnosis of the Profile load starts measure, you can identify the user sessions in which the profile was loaded and the time at which the session was initiated.

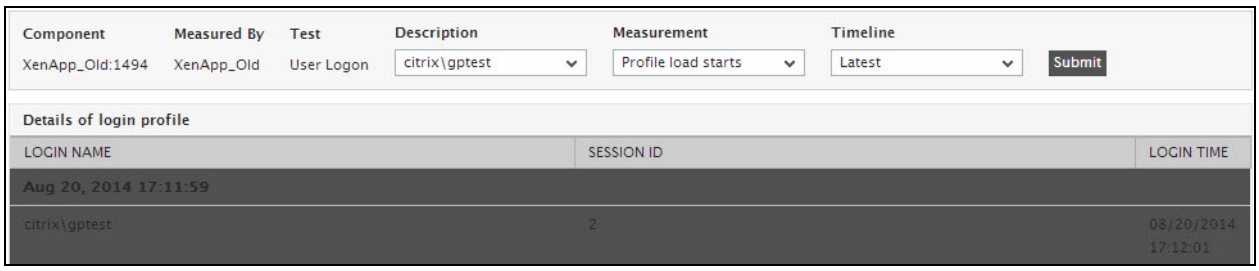


Figure 3.20: The detailed diagnosis of the Profile load starts measure

Use the detailed diagnosis of the *Profile unload starts* measure to know when a user's session was initiated and how long each session remained active on the XenApp server. From this, you can infer how many sessions were active for a user on the server and the duration of each session, and thus identify long-running sessions for the user.

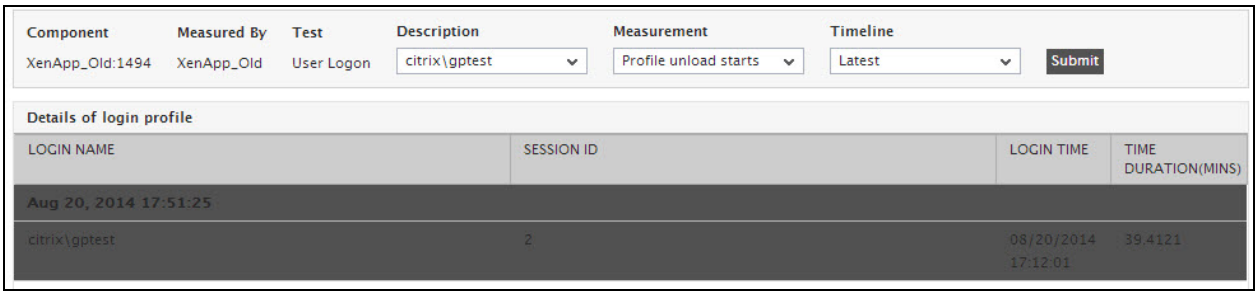


Figure 3.21: The detailed diagnosis of the Profile unload starts measure

To know the profile load time per user session, use the detailed diagnosis of the *User profile load time* measure. This will accurately pinpoint the session in which the profile took the longest to load.

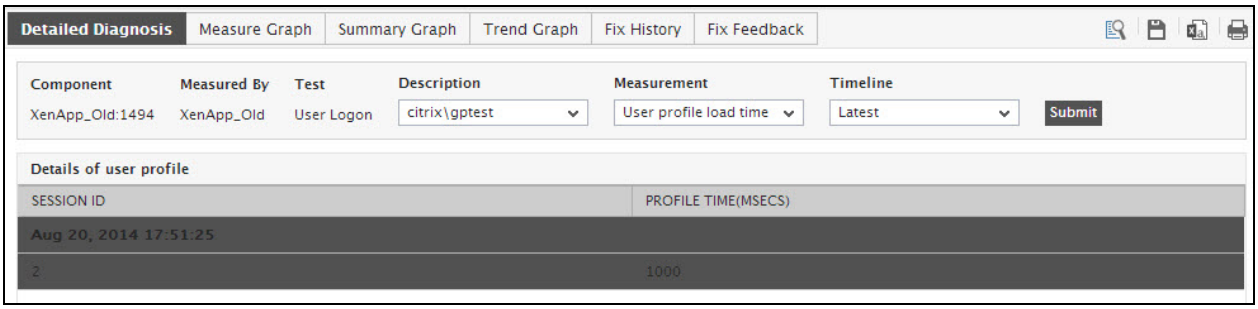


Figure 3.22: The detailed diagnosis of the User profile load time measure

3.5.24 Citrix XML Access Test

The Citrix XML Access Test verifies the interactions between the web interface, the XML service, and the IMA service.

A typical web interface interaction is composed of the following (see Figure 3.23):

- Client device users utilize a Web browser to view the Log in page and enter their user credentials.
- The NFuse server reads users' information and uses the Web Interface's classes to forward the information to the Citrix XML Service; this service can execute on the Citrix Web Interface or on each of the XenApp servers in a server farm. If the XML service is on the servers in a farm, the designated server acts as a broker between the NFuse server and the XenApp servers in the farm.
- The Citrix XML Service on the designated server then retrieves a list of applications from the servers that users can access. These applications comprise the user's application set. The Citrix XML Service retrieves the application set from the Independent Management Architecture (IMA) system and Program Neighborhood Service, respectively.

- The Citrix XML Service then returns the user's application set information to the Web Interface's classes running on the server.
- The user then clicks on the application of interest to him/her to access it.

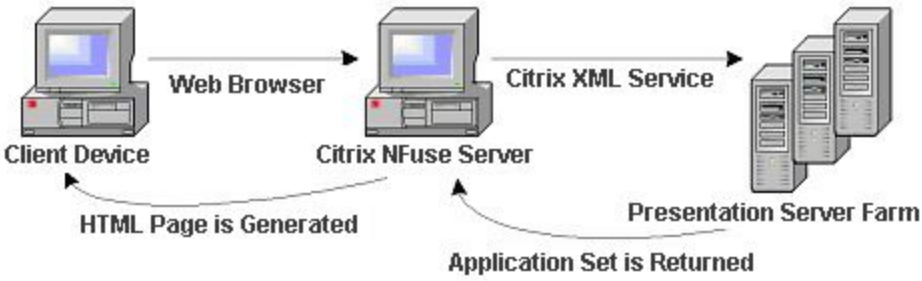


Figure 3.23: A typical web interface interaction

If the Citrix XML service executes on the XenApp servers in a farm, then you can use this test to evaluate the availability and responsiveness of the XML service. This test emulates a user accessing an XML port for a list of applications available to him/her. By emulating a request, this test checks that the entire application enumeration process involving the XML service and IMA service of Citrix is functioning properly. This test is disabled by default. 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 : Any Citrix Web Interface

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every Citrix Web Interface 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 Citrix server
4. **USER** - This test emulates a user logging into the NFuse server and requesting for a list of applications available to him/her. Therefore, in the **USER** text box, provide a valid user name which the test should use for logging into the NFuse server.

5. **PASSWORD** - Provide the **PASSWORD** of the specified **USER**.
6. **CONFIRM PASSWORD** - Confirm the password by retyping it in the **CONFIRM PASSWORD** box.
7. **SSL** - The web interface through which these tests are executing may be configured for HTTP or HTTPS access. If HTTPS access is configured, then this parameter should be set to **YES**.
8. **DOMAIN** - Provide the domain to which the user logs in.
9. **DOMAINTYPE** - A Citrix web interface can be set up to authenticate users by connecting to a Windows domain, or a Unix domain, or a Novell domain. The **DOMAINTYPE** value represents the type of domain being used to validate the user. The default value is "NT". For Unix, use "UNIX" and for Novell, use "NDS".
10. **XMLPORT** - Specify the port on which the Citrix XML Service is executing.
11. **NO OF TRIES** and **SLEEP TIME** - In environments where network connections are normally fuzzy and latencies are to be expected, the availability and response time checks performed by this test, may not always report accurate results. False alarms may hence be generated. In such environments therefore, you may want the test to try connecting to the XML service a few more times before reporting the availability and responsiveness of the service. To instruct the test to do so, you can use the **NO OF TRIES** and **SLEEP TIME** parameters. In the **NO OF TRIES** text box, indicate the number of times the test should try reconnecting to the XML service, and in the **SLEEP TIME** text box, specify how long (in seconds) the test should wait for a response from the service before attempting to reconnect. Both these parameters are set to 1 by default.
12. **TIMEOUT** - Specify the duration (in seconds) for which the test needs to wait for a response from the server. At the end of this duration, the test will timeout. The default is 30 seconds.
13. **ENCODING FORMAT** -

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Connection availability:	Tracks if the Citrix XML service is available to handle any requests.	Percent	If the TCP connection to the XML service port fails, this metric has a value of 0. Otherwise, it has a value of 100.
Authentication status:	Indicates if the user authentication succeeded.	Percent	It has a value of 100 if the user was authenticated, and a value of 0 if the authentication failed. If the user

Measurement	Description	Measurement Unit	Interpretation
			login is valid, yet authentication fails, the problem then lies with the Citrix IMA service's communication with the domain controller/active directory server.
Application enumeration status:	This metric indicates if the Citrix web interface was able to enumerate the applications available for the user logging in.	Percent	A value of 0 indicates that application enumeration failed, while a value of 100 denotes that the application enumeration operation succeeded. If authentication succeeds but application enumeration fails, then the problem is most likely to be in the Citrix XML service, its interaction with the IMA service, or with the IMA service itself.
TCP connection time:	Indicates the time taken to establish a TCP connection to the Citrix XML service.	Secs	If this value is significantly high, it could probably be because the network latency is high or the Citrix web interface host is overloaded.
Total response time:	Represents the total time taken for a user to login to the Citrix web interface and enumerate all the applications.	Secs	The value of this metric indicates the responsiveness of the Citrix web interface and its connectivity to the XML service.

3.5.25 Citrix XML Tickets Test

Once a user logs in to the Citrix web interface, he/she receives a list of applications to which they have access. When the user chooses one of the applications to open, the request is received by the web interface and forwarded to the local XML service. The XML service then asks the IMA service for the IP address of the least busy server that has the requested application published on it. The IMA service may have to contact the data collector for this information. In turn, the IMA service on the least loaded server contacts the terminal services on this system to obtain a ticket which provides the user with the permission to access the requested application.

The CitrixXmlTicket test is used to validate that the XML to IMA service interaction and the interaction between the IMA service and the terminal service on each system are working as expected. This testconnects to the web interface (specified by the xmlHost and xmlPort parameters) and issues an XML request asking the XML service for permission to login and access the application.

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 : Any Citrix server

Agent deploying the test : An external agent

Outputs of the test : One set of results for every Citrix 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 Citrix server
4. **USER** - This test connects to the web interface and issues an XML request asking the XML service for permission to login and access the application. Therefore, in the **USER** text box, provide a valid user name which the test should use for connecting to the web interface.
5. **PASSWORD** - Provide the **PASSWORD** of the specified **USER**.
6. **CONFIRM PASSWORD** - Confirm the password by retyping it in the **CONFIRM PASSWORD** box.
7. **SSL** - The web interface through which these tests are executing may be configured for HTTP or HTTPS access. If HTTPS access is configured, then this parameter should be set to **YES**.
8. **DOMAIN** - Provide the domain to which the user logs in.
9. **DOMAINTYPE** - A Citrix web interface can be set up to authenticate users by connecting to a Windows domain, or a Unix domain, or a Novell domain. The **DOMAINTYPE** value represents the type of domain being used to validate the user. The default value is "NT". For Unix, use "UNIX" and for Novell, use "NDS" in the domainType setting.
10. **XMLHOST** - Provide the IP/hostname of the web interface to which this test will attempt to connect.

11. **XMLPORT** - Provide the port number (respectively) of the web interface to which this test will attempt to connect.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Connection availability:	Tracks if the Citrix Nfuse service is available to handle any requests.	Percent	If the TCP connection to the XML service port fails, this metric has a value of 0. Otherwise, it has a value of 100.
Authentication status:	Indicates if the user authentication succeeded.	Percent	It has a value of 100 if the user was authenticated, and a value of 0 if the authentication failed. If the user login is valid, yet authentication fails, the problem then lies with the Citrix IMA service's communication with the domain controller/active directory server.
Ticket status:	Indicates if the Citrix XenApp server (actually the IMA service) was able to communicate with the terminal service and retrieve a ticket approving the user's access to the application of interest.	Percent	A value of 0 indicates that a valid ticket was not received.
TCP connection time:	Indicates the time taken to establish a TCP connection to the Citrix XML service port.	Secs	If this value is significantly high, it could probably be because the network latency is high or the Citrix web interface host is overloaded.
Response time for Citrix ticket generation:	Represents the total time taken for a user to login to the Citrix web interface and request to access an application.	Secs	The value of this metric indicates the responsiveness of the Citrix IMA service.

3.5.26 User Profile Management Test

User logon is a complex and resource intensive process on a Citrix XenApp system, and is a key determinant of the quality of a user's experience with the Citrix XenApp environment. This process is initiated when a XenApp farm load balancing algorithm selects the system 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 Citrix XenApp farm, causing significant loss of revenue and reputation in mission- critical environments.

One of the common causes for delays in user logons are delays in the loading of user profiles. To reduce the time taken to load profiles and thus minimize the user logon time, many Citrix administrators in recent times have been using 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 Citrix 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 XenApp farm greatly depends upon how efficient the service is.

To ascertain the efficiency of the Citrix Profile Management service, administrators may have to periodically track the logon/logoff duration and profile size of each user to a Citrix XenApp server and determine whether/not the Profile management service has succeeded in minimizing both user logon times and profile sizes. The **User Profile Management** 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 XenApp experience of such users.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every user to the Citrix 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 Citrix server

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Logon Duration:	Indicates the duration of logon processing for this user.	Secs	This value helps to measure the reduction in logon times when the Profile Management service 'streams' the profile. Ideally therefore, this value should be low. A high value or a consistent increase in the value of this measure could indicate that profile loading still takes a lot of time at logon - this could be owing to a large profile size. You can then check the value reported by the Logon Bytes measure to know the profile size at logon. If profile sizes continue to grow at logon despite the use of Profile management, it is indicative of the ineffectiveness of profile management. You may then have to fine-tune the feature to further reduce the profile size by

Measurement	Description	Measurement Unit	Interpretation
			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 XenApp server. If profile sizes continue to grow despite the use of Profile management, it is indicative of the ineffectiveness of profile management. You may then have to fine-tune the feature to further reduce the profile size by excluding more unnecessary files from the profile.
Logoff Duration:	Indicates the duration of logoff processing for this user.	Secs	A low value is desired for this measure. A high value could indicate that the profile management service takes too long to update the user's store with changes in the user settings. This could be because of a bad network connection between the XenApp server 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	MB	This measure provides a fair idea

Measurement	Description	Measurement Unit	Interpretation
	user's profile when it is copied to the user store at logoff.		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 XenApp server is spending too much time - is it when setting up the local profile? or is it when deleting the local profile?
Delete Local Profile Duration:	Indicates the time spent deleting this user's local profiles during the initial migration.	Secs	
Processed Logon Files Under 1KB:	Indicates the number of locally copied files for this user's profile that are synchronized during logon and categorized by the file size of 1KB.	Number	All the Processed Logon Files measures help Citrix 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 Citrix 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	

Measurement	Description	Measurement Unit	Interpretation
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	All the Processed Logon Files measures help Citrix administrators to understand whether/not 'profile streaming' (performed by the Profile Management service) has helped in the reducing the number of locally copied files during logon. All the Processed Logoff Files measures help Citrix administrators to understand how many files changed when the user session was in progress.
Processed Logoff Files from 10KB to 100KB:	Indicates the number of locally copied files for this user's profile that are synchronized during logoff and categorized by the file size ranging from 1KB to 10KB.	Number	
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	Number	

Measurement	Description	Measurement Unit	Interpretation
	from 100KB to 1MB.		
Processed Logon Files from 1MB to 5MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logon and categorized by the file size ranging from 1MB to 5MB.	Number	
Processed Logoff Files from 1MB to 5MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logoff and categorized by the file size ranging from 1MB to 5MB.	Number	All the Processed Logon Files measures help Citrix administrators to understand whether/not 'profile streaming' (performed by the Profile Management service) has helped in the reducing the number of locally copied files during logon. All the Processed Logoff Files measures help Citrix administrators to understand how many files changed when the user session was in progress.
Processed Logon Files Above 5MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logon and categorized by the file size above 5MB.	Number	
Processed Logoff Files Above 5MB:	Indicates the number of locally copied files for this user's profile that are synchronized during logoff and categorized by the file size above 5MB.	Number	

3.5.27 Data Store Check Test

When a XenApp server farm is deployed, it must have an associated data store. The data store provides a repository of persistent information, including:

- Farm configuration information
- Published application configurations
- Server configurations
- Citrix administrator accounts
- Printer configurations

Servers in a farm query the data store for configuration information when attempting to come online. If the data store is unavailable or is inaccessible for long hours, servers in the farm will remain offline the whole time, thus denying users access to their critical applications. To avoid this, administrators can run the **Data Store Check** test at frequent intervals, check whether/not the server is able to connect to the data store, and in this way, detect connection failures before farm users complain. In the event of a connection failure, administrators can also use the detailed metrics collected by this test to determine the reason for the connection failure and resolve it.

Target of the test : Any Citrix server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the Citrix server monitored

Configurable parameters for the test

1. **TEST PERIOD** – How often should the test be executedor
2. **HOST** – The host for which the test is to be configured
3. **PORT** – Refers to the port used by the Citrix server
4. **DSCHECKPATH** – This test uses XenApp's **Data Store Checker** tool to verify whether/not the monitored XenApp server is able to connect to the data store. To enable the test to use this tool, you need to specify the full path to the location of **DSCheck.exe** in the **DSCHECKPATH** text box. For instance, your path can be: *C:\Program Files (x86)\Citrix\system32*.
5. **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 abnormalfrequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Connectivity status:	Indicates whether the server succeeded or failed in establishing a connection with the data store.		The values that this measure can take and their corresponding numeric values are as follows: Measure Value / Numeric Value Failure: 0 Success: 1 If the value reported is *Failure*, you can use the detailed diagnosis of this test to determine the reason for the connection failure. **Note:** By default, this measure reports the above-mentioned **Measure Value**s to indicate the connectivity status of the data store. However, the graph of this measure will represent the same using the numeric equivalents only.

3.5.28 Citrix Server Load Test

The *Load evaluator* is a thread in the IMA Service on a XenApp Server that calculates the load index for that server. The load index is an integer value from 0 to 10,000 that represents how busy the XenApp server is.

Citrix administrators need to continuously track changes to the load index of a Citrix server, so that they can quickly isolate current/potential overload conditions on a server. Load index monitoring will also enable administrators understand the dynamics of load on the server, so that they can, if need be, reconfigure the load rules associated with the load evaluator according to the changes observed in load trends. For such load-level insights, administrators can use the **Citrix Server Load** test. If a load evaluator is configured for a monitored XenApp server, then this test will reveal the load index value of that server and will instantly alert administrators if the server is fully loaded. This way, the test warns administrators of a probable overload condition on the server.

Target of the test : Any Citrix server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the Citrix server monitored

Configurable parameters for the test

1. **TEST PERIOD** – How often should the test be executedor
2. **HOST** – The host for which the test is to be configured
3. **PORT** – Refers to the port used by the Citrix server

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Server load:	Indicates the current load index value of the server.	Number	The current load index is a result of the calculations and sum of the values of all the rules in each and every load evaluator which applies to the server. Each load evaluator contains one or more rules. Each rule defines an operational range for the server or published application to which its evaluator is assigned. Some of these rules are as follows:

<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>Rule</th><th>Description</th></tr>
<tr><td>Application User Load</td><td>Limits the number of users allowed to connect to a selected published application. The default value to report full load is 100.</td></tr>
<tr><td>Context Switches</td><td>Defines a range of context switches per second for a selected server. The default value to report full load is 16000.</td></tr>
<tr><td>CPU Utilization</td><td>Defines a range of processor utilization, as a percentage, for a selected server. The default value to report full load is 90 percent.</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>Rule</th><th>Description</th></tr>
<tr><td>Disk Data I/O</td><td>Defines a range of data throughput, in kilobytes per second, for a selected server. The default full load value is 32767 kilobytes per second.</td></tr>
<tr><td>Disk Operations</td><td>Defines a range of disk operation, in read/write cycles per second, for a selected server. The default full load value is 100 operations per second.</td></tr>
<tr><td>Memory Usage</td><td>Defines a range of memory usage by a server. The default full load value is</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>Rule</th><th>Description</th></tr>
<tr><td></td><td>90.</td></tr>
<tr><td>Page Fault</td><td>Defines a range of page faults per second for a selected server. A page fault occurs when the operating system tries to access data that was moved from physical memory to disk. The default full load value is 2000.</td></tr>
<tr><td>Page Swaps</td><td>Defines a range of page swaps per second for a selected server. A page swap occurs when the operating system moves data between</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>Rule</th><th>Description</th></tr>
<tr><td></td><td>physical memory and the swap file. The default full load value is 100.</td></tr>
<tr><td>Server User Load</td><td>Limits the number of users allowed to connect to a selected server. The default full load value is 100 and represents the maximum number of users the system can support on a server.</td></tr>
</table>

The base algorithm for establishing actual load is:

Highest_Load + (Average_Other_ Loads * .1)

The resultant value is reported as the value of this measure.

If the value of this measure is in the range of 0 to 9998, it implies that the</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			server load is normal. On the other hand, if the measure value touches or exceeds 10000, it implies that server load is at 100%. In this case, XenApp automatically removes the load- managed server from the internal list of available servers. The next request for an ICA connection to a published application is routed to the next available load- managed server in the list. One important factor to understand is once any single rule reaches its maximum value, the load value for that server becomes 10,000, effectively removing the individual server from contention for new sessions.
Is load evaluator configured ?	Indicates whether or not any load evaluator has been configured for the server.		If the Server load measure reports the value 99999, this measure will return the value No, indicating that no load evaluator has been configured for the server. On the other hand, if the Server load measure reports a value between 0 and 9998 or the value 10000, then, this measure will report the value Yes; this indicates that a load evaluator has been configured for the server. The numeric values that correspond to these Yes and No measure values are listed 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>Yes</td><td>1</td></tr>
<tr><td>No</td><td>0</td></tr>
</table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the server has been configured with a load evaluator. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.
</td></tr>
<tr><td>Is server fully loaded?</td><td>Indicates whether/not the server is fully loaded.</td><td></td><td>If the Server load measure reports the value 10000, this measure will return the value Yes, indicating that the server is fully loaded. One important factor to understand is once any single rule associated with the load evaluator of the server reaches its maximum value, the load value for that server becomes 10000.

On the other hand, if the Server load measure reports a value between 0 and 9998, then, this measure will report the value No; this indicates that the server is not fully loaded.

The numeric values that correspond to these Yes and No measure values are listed 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>Yes</td><td>1</td></tr>
<tr><td>No</td><td>0</td></tr>
</table>
Note:

By default, this measure reports the Measure Values listed in the table above to indicate whether/not the server is fully loaded. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.
</td></tr>
<tr><td>Is there a license/edition mismatch?</td><td>Indicates if an incorrect server edition is being used or a license mismatch has been detected.</td><td></td><td>
If the Server load measure reports the value 20000, this measure will return the value Yes, indicating that the AppCenter console contains an incorrect server edition or a license mismatch.

On the other hand, if the Server load measure reports a value between 0 and 9998, the value 10000, or the value 99999, then this measure will report the value No; this indicates that no such mismatch has been detected.

The numeric values that correspond to these Yes and No measure values are listed below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Yes</td><td>1</td></tr>
<tr><td>No</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 if the server edition is incorrect or the license does not match. In the graph of this measure however, the same is represented using the corresponding numeric equivalents only.

3.5.29 ICA/RDP Listeners Test

The listener component runs on the XenApp/Terminal server and is responsible for listening for and accepting new ICA/RDP client connections, thereby allowing users to establish new sessions on the XenApp/Terminal server. If this listener component is down, users may not be able to establish a connection with the XenApp server!

This is why, if a user to the XenApp server complains of the inaccessibility of the server, administrators should first check whether the Citrix listener component is up and running or not. The **ICA/RDP Listeners** test helps administrators perform this check. This test tracks the status of the default listener ports and reports whether any of the ports is down.

Target of the test : A Citrix XenApp server

Agent deploying the test : Internal agent

Outputs of the test : One set of outputs for every listener port configured

Configurable parameters for this 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
4. **SESSION IDS** – The default listener ports - 65536,65537,65538 – will be displayed here by default. You can override this default specification by adding more ports or by removing one/more existing ports.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Is listener down?:	Indicates whether/not this listener port is down.		This measure reports the value Yes if the listener port is down and No if the port is up and running. The numeric values that correspond to these measure values are as follows: **Measure Value** / **Numeric Value** Yes: 0 No: 1 **Note:** By default, this measure reports the above-mentioned **Measure Value**s to indicate the status of a listener port. However, the graph of this measure will represent the same using the numeric equivalents only.

3.6 The Citrix Applications Layer

Using the tests mapped to this layer, the resource usage per application executing on the Citrix server can be measured.

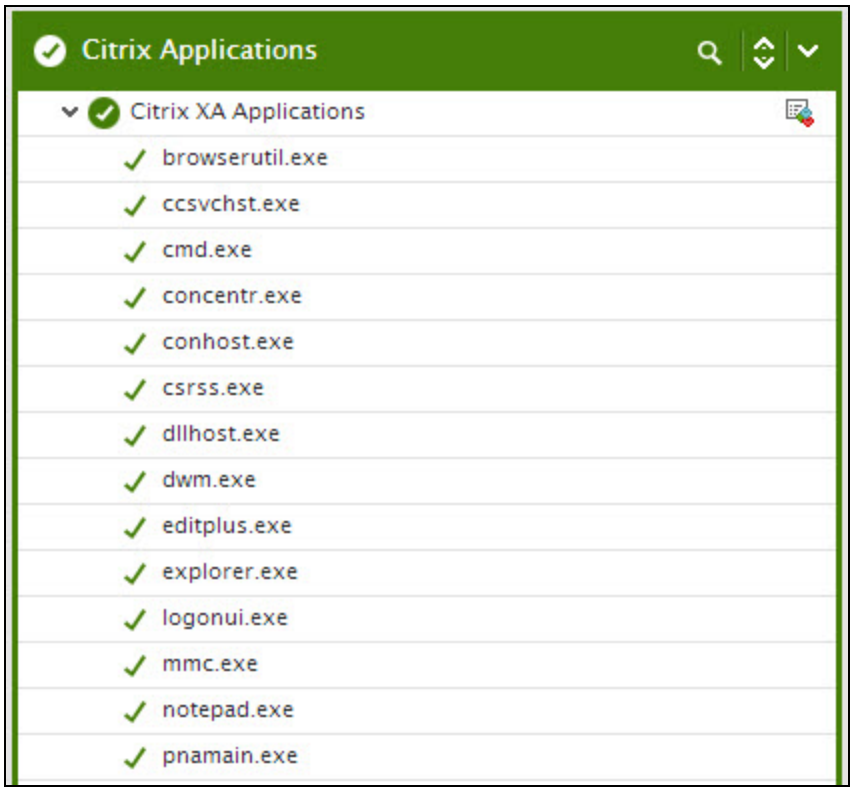


Figure 3.24: Tests associated with the Citrix Applications layer

3.6.1 Citrix XA Applications Test

This test reports statistics pertaining to the different applications executing on a Citrix server and their usage by Citrix clients. One set of results is reported for each application.

Note:

This test will report metrics only if the XenApp server being monitored uses the .Net framework v3.0 (or above).

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results is reported for each application

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 Citrix server
4. **APPS** - By default, the **APPS** text box will contain 'all'. This means that, by default, the eG Enterprise system will monitor all the applications running on a Citrix server. Alternatively, you can provide a comma-separated list of applications that require monitoring. For example: *winword.exe, acrobat.exe*. To monitor the published applications only, specify 'published'.
5. **APPSBYNAME** - This parameter is relevant only if the "apps" parameter is "published" - that is, the agent is monitoring only published applications. By default, this parameter is set to "**no**", which means the agent monitors the applications by process name (e.g., msword, iexplore, sfttray, excel, etc.). If this parameter is set to "**yes**", the agent reports by published application name (e.g., Microsoft Word instead of "msword").

   This parameter is particularly relevant if a virtual client like the Softgrid client is deployed on Citrix. In this case, all the user processes will run the Softgrid client (ie, sfttray.exe) and by just monitoring the process names, administrators will not be able to differentiate Microsoft Word instances from Microsoft Excel instances being served by the Softgrid client. If the appsbyname parameter is "yes", the agent compares the full process command including arguments with the published application information and is able to differentiate applications that may be served using the same executable program.
6. **SHOWPUBLISHEDDESKTOPS** - By default, this flag is set to **No**. If set to **Yes**, then the detailed diagnosis of the test, which typically reveals the users accessing an application and the resource usage of each such user, will now additionally indicate the exact published desktop that has been used by the user to access the application.
7. **REPORTBYCLIENTNAME** - By default, this flag is set to **No**. If set to **Yes**, then an additional **CLIENT NAME** column will appear in the detailed diagnosis of this test. This column will indicate the host name of the client machine from which the users accessed the configured applications. When many users access an application on a Citrix XenApp server using the same login credentials, then multiple rows of information in the detailed diagnosis will display the same **Username**. Under such circumstances, it would be more useful to have the detailed diagnosis also indicate the client machine from which each user accessed that application. To achieve this, set the **REPORTBYCLIENTNAME** flag to **Yes**.
8. **APPS REDISCOVER PERIOD** - By default, the test rediscovers the applications running on a Citrix server, once in a day; this is why, the **APPS REDISCOVER PERIOD** is set to *1440* by default. You can override this default setting by specifying a different duration (in minutes) in the **APPS REDISCOVER PERIOD** text box.
10. **CTXAPPDISCTIMERANGE** - Typically, when monitoring a Citrix server/farm on which numerous applications have been deployed, the processing overheads of this test may increase every time the test performs application discovery. You may hence prefer to rediscover the

applications on these servers/farms only during such times the user activity/load on the server/farm is low. To schedule application rediscovery during the 'low-activity' time window of a XenApp server, you can use the **CTXAPPDISCTIMERANGE** parameter. Here, specify a time range in the following format: *StartingHrs-Ending Hrs*. The *Hrs* here should be in the 24-hour format. For instance, to make sure that the test performs application rediscovery only during 8PM and 11PM every day, your **CTXAPPDISCTIMERANGE** specification will be: *20-23*. **Note that you cannot suffix your 'Hrs' specification with 'Minutes' or 'Seconds'.**

By default, the **CTXAPPDISCTIMERANGE** is *none*; this implies that applications are by default rediscovered only in the frequency specified against **APPS REDISCOVER PERIOD**. However, if a valid time range is provided against the **CTXAPPDISCTIMERANGE** parameter, then this time range will automatically override the **APPS REDISCOVER PERIOD**.

11. **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** – A Citrix XenApp server (v6.5) can run in the **controller mode** or the **worker mode**. In the controller mode, the XenApp server can perform all farm management tasks. However, in the worker mode, a XenApp server can only host user sessions.

    By default, the **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** parameters are set to *none*. If the XenApp server being monitored is the **controller** in a farm, then this default value will automatically apply. In other words, in this case, you can leave the values of these parameters as *none*. On the other hand, if the target XenApp server is a **worker** in the farm, then first, you will have to configure the name of the domain in which the XenApp server operates in the **DOMAIN NAME** text box. Then, you need to configure the test with the credentials of a user with **Citrix Farm Administrator** rights, using the **DOMAIN USER** and **DOMAIN PASSWORD** text boxes. Finally, you will have to confirm the **DOMAIN PASSWORD** by retyping it in the **CONFIRM PASSWORD** text box.

    Note:

    If the XenApp server is a worker in the farm, then, in addition to configuring the domain name, domain user, domain password parameters, the following pre-requisites should also be fulfilled for this test to report metrics:

    - Make sure that the Citrix XenApp Commands Remoting service is running in the Controller server.
    - The port 2513 must be open on the Controller server in the farm.

13. **SHOW WORKER GROUPS** - Worker groups are collections of XenApp servers, residing in the same farm, that are managed as a single unit. You can publish applications to a worker group. If you want to know the worker group to which every auto-discovered application has been published, then set this

parameter to **Yes**. Once this is done, then the descriptors (i.e., the auto-discovered applications) of this test will be grouped by the name of the worker group to which they belong. By default, this parameter is set to **No**.

14. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who accessed an application on the server. This way, administrators will be able to quickly determine which user logged into the server from which domain. If you want the detailed diagnosis to display only the username of these users, set this flag to **No**.
15. **ENABLE BROWSER MONITORING**– By default, this flag is set to **No**, indicating that the eG agent does not monitor browser activity on the XenApp server. If this flag is set to **Yes**, then, whenever one/more IE (Internet Explorer) browser instances on the XenApp server are accessed, the detailed diagnosis of the *Processes running* measure will additionally reveal the URL being accessed via each IE instance and the resources consumed by every URL. Armed with this information, administrators can identify the web sites that are responsible for excessive resource usage by an IE instance.
16. **SHOW ONLY ACTIVE APPS** – Using this flag, you can indicate whether the test should monitor all applications or applications that are currently active on the server. By default, this flag is set to **Yes**, indicating that only the currently active applications will be monitored by the eG agent. To monitor all applications, you need to set this flag to **No**.
17. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

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

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

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Processes	Number of instances of	Number	This value indicates if too many or

Measurement	Description	Measurement Unit	Interpretation
running:	the published application currently executing on the Citrix server		too few instances corresponding to an application are executing on the host.
Cpu usage:	Percentage of CPU used by the published application	Percent	A very high value could indicate that the specified application is consuming excessive CPU resources.
Memory usage:	This value represents the ratio of the resident set size of the memory utilized by the application to the physical memory of the host system, expressed as a percentage.	Percent	A sudden increase in memory utilization for an application may be indicative of memory leaks in the application.
Handle count:	Indicates the number of handles opened by this application.	Number	An increasing trend in this measure is indicative of a memory leak in the application.
Number of threads:	Indicates the number of threads that are used by this application.	Number	
I/O data rate:	Indicates the rate at which this application is reading and writing bytes in I/O operations.	Kbytes/Sec	This value counts all I/O activity generated by an application and includes file, network and device I/Os.
I/O data operations:	Indicates the rate at which this application is issuing read and write data to file, network and device I/O operations.	Operations/Sec	

Measurement	Description	Measurement Unit	Interpretation
I/O read data rate:	Indicates the rate at which this application is reading data from file, network and device I/O operations.	Kbytes/Sec	
I/O write data rate:	Indicates the rate at which this application is writing data to file, network and device I/O operations.	Kbytes/Sec	
Page fault rate:	Indicates the total rate at which page faults are occurring for the threads of this application.	Faults/Sec	A page fault occurs when a thread refers to a virtual memory page that is not in its working set in main memory. This may not cause the page to be fetched from disk if it is on the standby list and hence already in main memory, or if it is in use by another process with whom the page is shared.
Working set memory used:	Indicates the current size of the working set of this application.	MB	The Working Set is the set of memory pages touched recently by the threads in a process/application. If free memory in the server is above a threshold, pages are left in the Working Set of an application even if they are not in use. When free memory falls below a threshold, pages are trimmed from Working Sets. If they are needed they will then be soft-faulted back into the Working Set before leaving main memory. Comparing the working set across applications indicates which application is taking up excessive memory.

The detailed diagnosis of the *Processes running* measure, if enabled, lists the applications running on the XenApp server, the process ids that correspond to each running application instance, the user who accessed each instance, and the overall resource usage of each of instances. This information enables the Citrix administrator to identify the processes that are utilizing resources excessively and those that may be leaking memory. In the event of a server overload/memory leak, the Citrix administrator might decide to terminate these processes (see Figure 3.25). In addition, the detailed diagnosis reveals the location from which each process instance runs (i.e., the **IMAGE PATH**). If multiple versions of an application are published in different locations on the XenApp server and a user runs each of these versions, then the **IMAGE PATH** will indicate the exact application version each process instance corresponds to – resource-hungry versions can thus be identified.

Lists The Processes Executed By A User On A Citrix Server

TIME	PID	PROCESS NAME	CPU(%)	MEMORY(%)	IO READS (KBPS)	IO WRITES (KBPS)	PAGE FAULTS (FAULT/S)	VIRTUAL MEMORY (MB)	HANDLES	PUBLISHED DESKTOP	PARENT PID	USERNAME	IMAGE PATH	WEBSITE TITLE	WEBSITE URL	WEBSITE DOMAIN
Jun 03, 2014 18:45:38																
	1308	wfshell	0	0.7832	0	0	0	141.37	444	-	4308	-	C:\Program Files (x86)\Citrix\System32\wfshell.exe	-	-	-
	2580	csrss	0	0.2649	0	0	0	55.82	244	-	6488	-	%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,20480,768 Windows=On SubSystemType=Windows ServerDll=basesrv,1 ServerDll=winsrv:UserServerDllInitialization,3 ServerDll=winsrv:ConServerDllInitialization,2 ServerDll=sxssrv,4 ProfileControl=Off MaxRequestThreads=16	-	-	-
	5464	winlogon	0	0.5989	0	0	0	101.14	264	-	4172	-	winlogon.exe	-	-	-
	6728	winlogon	0	0.5901	0	0	0	99.66	265	-	6488	-	winlogon.exe	-	-	-

Figure 3.25: The detailed diagnosis of the Processes running measure

Moreover, if one or more browser instances are found to consume excessive CPU, memory and disk I/O resources on a server or a desktop, then for each such browser instance, administrators can now see a mapping of browser process to URL being accessed, as well as the resources used by each browser process in the detailed diagnosis. Armed with this information, administrators can determine the steps required to avoid excessive resource usage by browser instances – e.g., whether specific web sites are responsible for this, whether users are accessing web sites (e.g., youtube, facebook, etc.) that they should not be accessing from a corporate network, etc.

Note:

- The eG agent will perform browser activity monitoring only if the **ENABLE BROWSER MONITORING** flag is set to **Yes**.
- The eG agent will monitor browser activity only of the browser being accessed is **Internet Explorer**.

3.6.2 App-V Applications Test

This test reports statistics pertaining to the different applications executing on an App-V client and their usage. In addition, this test also reports the statistics pertaining to the processes running on the APP-V client.

Note:

This test will report metrics only when the App-V Client is installed on the Citrix XenApp Server.

Target of the test : An App-V Client on the target Citrix XenAPP Server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each application of the target App-V Client that is to be 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. **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.
5. **EXTENDED STATISTICS** – By default, this test provides you with detailed measures on the resource utilization of each application. If you wish to obtain only the CPU and memory related measures, then set the **EXTENDED STATISTICS** flag to **No**.
6. **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 enabled/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

   - The eG manager license should allow the detailed diagnosis capability
   - Both the normal and abnormal frequencies configured for the detailed diagnosis measures

should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total size:	Indicates the total size of this virtual application package.	MB	The detailed diagnosis of this measure lists the Version of the application, Application ID, Version ID of the application and the application path.
Is loading?:	Indicates whether this application is currently loading or not on the App-V client.		This measure reports a value True if the application is currently being loaded and a value False if otherwise. These measure values and their corresponding numeric values are listed in the table below: Measure Value / Numeric Value True / 0 False / 1 **Note:** By default, this measure reports the values *Yes* or *No* to indicate whether this application is currently being loaded on the client or not. The graph of this measure however is represented using the numeric equivalents - 0 or 1.
Loaded percentage:	Indicates the percentage of this application that is currently being loaded	Percent	

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>on the App-V client.</td><td></td><td></td></tr>
<tr><td>In use?:</td><td>Indicates whether this application is currently in use or not.</td><td></td><td>This measure reports a value True if the application is currently in use and a value False if otherwise.

These measure values and their corresponding numeric values are listed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>True</td><td>0</td></tr><tr><td>False</td><td>1</td></tr></table>
Note:

By default, this measure reports the values Yes or No to indicate whether this application is currently in use. The graph of this measure however is represented using the numeric equivalents - 0 or 1.</td></tr>
<tr><td>Any user based pending tasks available?</td><td>Indicates whether any tasks are pending for the user using this application.</td><td></td><td>This measure reports a value Yes if any tasks are pending for the user using the application and a value No if otherwise.

These measure values and their corresponding numeric values are listed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>True</td><td>0</td></tr><tr><td>False</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 values Yes or No to indicate whether any tasks are currently pending for the user using this application. The graph of this measure however is represented using the numeric equivalents - 0 or 1.</td></tr>
<tr><td>Any global based pending tasks available:</td><td>Indicates whether any global tasks are pending for this application.</td><td></td><td>This measure reports a value Yes if any tasks are pending for the user using the application and a value No if otherwise.

These measure values and their corresponding numeric values are listed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Yes</td><td>0</td></tr><tr><td>No</td><td>1</td></tr></table>
Note:

By default, this measure reports the values Yes or No to indicate whether any tasks are currently pending for the user using this application. The graph of this measure however is represented using the numeric equivalents - 0 or 1.</td></tr>
<tr><td>Processes running:</td><td>Indicates the number of instances of this</td><td>Number</td><td>This value indicates if too many or too few instances corresponding to</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
	application currently executing.		an application are executing on the host. The detailed diagnosis of this measure, if enabled, displays the complete list of processes executing, the users executing them, and their individual resource utilization.
CPU utilization:	Indicates the percentage of CPU used by this application.	Percent	A very high value could indicate that the specified application is consuming excessive CPU resources.
Memory utilization:	This value represents the ratio of the resident set size of the memory utilized by the application to the physical memory of the host system, expressed as a percentage.	Percent	A sudden increase in memory utilization for an application may be indicative of memory leaks in the application.
Handle count:	Indicates the number of handles opened by this application.	Number	An increasing trend in this measure is indicative of a memory leak in the process.
I/O data rate:	Indicates the rate at which processes are reading and writing bytes in I/O operations.	Kbytes/Sec	This value counts all I/O activity generated by each process and includes file, network and device I/Os.
I/O data operations:	Indicates the rate at which this application process is issuing read and write data to file, network and device I/O operations.	Operations/Sec	
I/O read data rate:	Indicates the rate at which the process is	Kbytes/Sec	

Measurement	Description	Measurement Unit	Interpretation
	reading data from file, network and device I/O operations.		
I/O write data rate:	Indicates the rate at which the process is writing data to file, network and device I/O operations.	Kbytes/Sec	
Number of threads:	Indicates the number of threads that are used by this application.	Number	
Page fault rate:	Indicates the total rate at which page faults are occurring for the threads of all matching application processes.	Faults/Sec	A page fault occurs when a thread refers to a virtual memory page that is not in its working set in main memory. This may not cause the page to be fetched from disk if it is on the standby list and hence already in main memory, or if it is in use by another process with whom the page is shared.
Virtual memory used:	Indicates the amount of virtual memory that is being used by the application.	MB	
Memory working set:	Indicates the current size of the working set of a process.	MB	The Working Set is the set of memory pages touched recently by the threads in the process. If free memory in the computer is above a threshold, pages are left in the Working Set of a process even if they are not in use. When free memory falls below a threshold, pages are trimmed from

Measurement	Description	Measurement Unit	Interpretation
			Working Sets. If they are needed they will then be soft-faulted back into the Working Set before leaving main memory. If a process pattern matches multiple processes, the memory working set reported is the sum of the working sets for the processes that match the specified pattern. Detailed diagnosis for this test provides details of the individual processes and their individual working sets. Comparing the working set across processes indicates which process (es) are taking up excessive memory. By tracking the working set of a process over time, you can determine if the application has a memory leak or not.

3.6.3 Citrix XA Application Launches Test

To know which published applications on the XeAnApp server are currently accessed by users and how many instances of each application have been launched presently, use the **Citrix XA Application Launches** test. Detailed diagnostics, if enabled, reveal the users accessing the published applications and the thin clients from which the users are connecting to the XenApp server.

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 : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 'published application' on the XenApp server 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** – Refers to the port used by the Citrix server
4. **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** – A Citrix XenApp server (v6.5) can run in the **controller mode** or the **worker mode**. In the controller mode, the XenApp server can perform all farm management tasks. However, in the worker mode, a XenApp server can only host user sessions.

   By default, the **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** parameters are set to *none*. If the XenApp server being monitored is the **controller** in a farm, then this default value will automatically apply. In other words, in this case, you can leave the values of these parameters as *none*. On the other hand, if the target XenApp server is a **worker** in the farm, then first, you will have to configure the name of the domain in which the XenApp server operates in the **DOMAIN NAME** text box. Then, you need to configure the test with the credentials of a user with **Citrix Farm Administrator** rights, using the **DOMAIN USER** and **DOMAIN PASSWORD** text boxes. Finally, you will have to confirm the **DOMAIN PASSWORD** by retyping it in the **CONFIRM PASSWORD** text box.

   Note:

   If the XenApp server is a worker in the farm, then, in addition to configuring the domain name, domain user, domain password parameters, the following pre-requisites should also be fulfilled for this test to report metrics:

   - Make sure that the Citrix XenApp Commands Remoting service is running in the Controller server.
   - The port 2513 must be open on the Controller server in the farm.

5. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who logged into the Citrix server. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Application instances running:	Represents the number of instances of this published application that are running currently.	Number	Use the detailed diagnosis of this measure to know which users are currently accessing the application and the clients from which the users are connecting.

3.6.4 Application Process Launches Test

When a user complains that it is taking too long to launch applications on Citrix, 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** test provides these valuable insights to the administrators. This test auto-discovers all the applications that are currently launched on the Citrix server, 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 : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every application on the XenApp server that is currently launched. This can be an application published on a XenApp server or that which runs within a published desktop on the XenApp 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 number at which the specified **HOST** listens to. By default, this is 1494.

4. **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** – A Citrix XenApp server (v6.5) can run in the **controller mode** or the **worker mode**. In the controller mode, the XenApp server can perform all farm management tasks. However, in the worker mode, a XenApp server can only host user sessions.

   By default, the **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** parameters are set to *none*. If the XenApp server being monitored is the **controller** in a farm, then this default value will automatically apply. In other words, in this case, you can leave the values of these parameters as none. On the other hand, if the target XenApp server is a **worker** in the farm, then first, you will have to configure the name of the domain in which the XenApp server operates in the **DOMAIN NAME** text box. Then, you need to configure the test with the credentials of a user with **Citrix Farm Administrator** rights, using the **DOMAIN USER** and **DOMAIN PASSWORD** text boxes. Finally, you will have to confirm the **DOMAIN PASSWORD** by retyping it in the **CONFIRM PASSWORD** text box.

   Note:

   If the XenApp server is a worker in the farm, then, in addition to configuring the domain name, domain user, domain password parameters, the following pre-requisites should also be fulfilled for this test to report metrics:

   - Make sure that the Citrix XenApp Commands Remoting service is running in the Controller server.
   - The port 2513 must be open on the Controller server in the farm.

5. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user session that logged out. This default setting ensures that administrators are able to quickly determine the domains to which the users who logged out belonged. You can set this flag to **No** if you want detailed diagnosis to display only the username of the users who logged out.

6. **SERVER VERSION** – By default, this parameter is set to XA6 for this test. **Do not change this default setting**.

7. **EXCLUDE** – By default, this parameter is set to none. This means that the test will monitor all the applications that are launched on the XenApp server, by default. If you want the test to disregard certain applications when monitoring, then provide a comma-separated list of process names that correspond to the applications you want to ignore, in the **EXCLUDE** text box. For instance, your specification can be: *winword.exe,js.exe,taskmgr.exe* . Your specification can include wild card patterns as well. For example: **win*,js*,*task*

8. **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**.

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

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

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

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
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 know 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

Measurement	Description	Measurement Unit	Interpretation
			which user experienced this delay in launching, use the detailed diagnosis of the Launch count measure.

3.6.5 Outlook Add-ins 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 Microsoft RDS server, 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** test provides these valuable insights to the administrators. This test auto-discovers all the add-ins integrated with the Microsoft Outlook published on the Microsoft RDS 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 RDS 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 Microsoft RDS server being monitored

Configurable parameters for the test

Parameters	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
Port	The port number at which the specified host listens to.
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.
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
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

Measurement	Description	Measurement Unit	Interpretation
			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.

3.7 The Citrix Users layer

To accurately assess the individual user experience on the Citrix server, use the tests mapped to the **Citrix Users** layer.

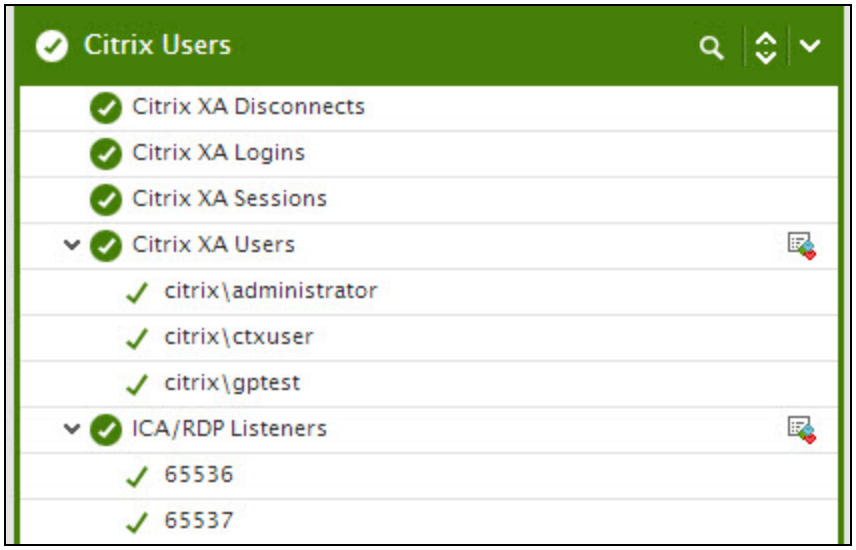


Figure 3.26: The test associated with the Citrix Users layer

3.7.1 Citrix XA Users Test

A Citrix environment is a shared environment in which multiple users connect to a Citrix server/server farm and access a wide variety of applications. When server resources are shared, excessive resource utilization by a single user could impact the performance for other users. Therefore, continuous monitoring of the activities of each and every user on the server is critical. Towards this end, the **Citrix XA Users** test assesses the traffic between the user terminal and the server, and also monitors the resources taken up by a user's session on the server. The results of

this test can be used in troubleshooting and proactive monitoring. For example, when a user reports a performance problem, an administrator can quickly check the bandwidth usage of the user's session, the CPU/memory/disk usage of this user's session as well as the resource usage of other user sessions. The administrator also has access to details on what processes/applications the user is accessing and their individual resource usage. This information can be used to spot any offending processes/ applications.

Note:

This test will report metrics only if the XenApp server being monitored uses the .Net framework v3.0 (or above).

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every user logged into the Citrix 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** – Refers to the port used by the Citrix MF XP server.
4. **USERNAMES** - Specify the name of the user whose performance statistics need to be generated. By default, "all" will be displayed here, indicating that the eG agent, by default, reports statistics pertaining to all users who are currently logged in. Multiple user names can be specified as a comma-separated list. In such cases, the eG agent will report statistics for the users listed in the arguments only.
5. **FARMNAME** - If the Citrix server for which this test is being configured belongs to a Citrix farm, then provide the name of the Citrix farm server that controls it, in the **FARMNAME** text box. While specifying the **FARMNAME**, ensure that you provide the same name that was specified against the **HOST/NICK NAME** field while managing the Citrix farm server using the eG Enterprise system. In the event of a name mismatch, eG will be unable to extract the required measures for this test. By default, 'none' will be displayed here.
6. **FARMPORT** – Specify the port number at which the Citrix farm listens.
7. **APPSBYNAME** - By default, this flag is set to **No** - i.e., the detailed diagnosis for a user reports the process name(s) being run by the user. If this parameter is set to **Yes**, the agent compares the full process command including arguments with the published application information and

reports the process that the user is running plus the application that the user is accessing (e.g., MSWord (sfftray) - in this example, MSWord is the published application name, and sfftray is the Softgrid client executable that is streaming this application from a Softgrid server).

8. **SHOWPUBLISHEDDESKTOPS** - By default, this flag is set to **No**. If set to **Yes**, then the detailed diagnosis of the test, which typically lists the resource-intensive processes/applications accessed by a user, will additionally indicate the exact published desktop that has been used by the user or used to access the application.
9. **REPORT TOTAL** - By default, this flag is set to **No**. If set to **Yes**, then the test will report measures for only a *Total* descriptor. For this descriptor, the test will report the aggregate resource usage across all users to the Citrix server. The default setting (**No**) of the flag on the other hand, implies that the test reports a set of metrics for each user to the server, by default.
10. **REPORTBYCLIENTNAME** - By default, this flag is set to **No**. If set to **Yes**, this test will report metrics for each client machine from which users logged into the XenApp server - i.e., the host name of the client machines will be the descriptors of this test. In this case therefore, the **User name** column of the detailed diagnosis of this test will indicate the names of the users who logged into the XenApp server.

    On the other hand, if the **REPORTBYCLIENTNAME** flag is set to **No**, then user names will be the descriptors of the test, and the **User name** column in the detailed diagnosis will display a '-' (hyphen).

11. **APPS REDISCOVER PERIOD** - By default, the test rediscovers the applications running on a Citrix server, every 15 minutes; this is why, the **APPS REDISCOVER PERIOD** is set to **15** by default. You can override this default setting by specifying a different duration (in minutes) in the **APPS REDISCOVER PERIOD** text box.
12. **CTXAPPDISCTIMERANGE** - Typically, when monitoring a Citrix server/farm on which numerous applications have been deployed, the processing overheads of this test may increase every time the test performs application discovery. You may hence prefer to rediscover the applications on these servers/farms only during such times the user activity/load on the server/farm is low. To schedule application rediscovery during the 'low-activity' time window of a XenApp server, you can use the **CTXAPPDISCTIMERANGE** parameter. Here, specify a time range in the following format: *StartingHrs-Ending Hrs*. The *Hrs* here should be in the 24-hour format. For instance, to make sure that the test performs application rediscovery only during 8PM and 11PM every day, your **CTXAPPDISCTIMERANGE** specification will be: *20-23*. **Note that you cannot suffix your 'Hrs' specification with 'Minutes' or 'Seconds'.**

    By default, the **CTXAPPDISCTIMERANGE** is *none*; this implies that applications are by default

rediscovered only in the frequency specified against **APPS REDISCOVER PERIOD**. However, if a valid time range is provided against the **CTXAPPDISCTIMERANGE** parameter, then this time range will automatically override the **APPS REDISCOVER PERIOD**.

13. **SEPARATE PROCESS** - By default, this parameter is set to **Auto**. This implies that by default, this test auto-discovers the operating system on which the target Citrix server is running. Based on the auto-discovered OS, the test uses either the eG agent process itself to collect the required metrics or spawns a separate process for this purpose. If the target server is discovered to be executing on a Windows 2003 (or earlier) platform, then the eG agent process itself will collect the metrics reported by this test. On the other hand, if the target server is found to execute on Windows 2008 (or above), then a separate process is spawned for metrics collection. Alternatively, you can set this flag to **true** or **yes**. In this case, metrics collection is performed by a separate process, regardless of the operating system of the Citrix server. If you set this flag to **false** or **no** on the other hand, then the eG agent process collects the metrics, regardless of the operating system of the Citrix server.
14. **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** – A Citrix XenApp server (v6.5) can run in the **controller mode** or the **worker mode**. In the controller mode, the XenApp server can perform all farm management tasks. However, in the worker mode, a XenApp server can only host user sessions.

    By default, the **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** parameters are set to *none*. If the XenApp server being monitored is the **controller** in a farm, then this default value will automatically apply. In other words, in this case, you can leave the values of these parameters as *none*. On the other hand, if the target XenApp server is a **worker** in the farm, then first, you will have to configure the name of the domain in which the XenApp server operates in the **DOMAIN NAME** text box. Then, you need to configure the test with the credentials of a user with **Citrix Farm Administrator** rights, using the **DOMAIN USER** and **DOMAIN PASSWORD** text boxes. Finally, you will have to confirm the **DOMAIN PASSWORD** by retyping it in the **CONFIRM PASSWORD** text box.

    Note:

    If the XenApp server is a worker in the farm, then, in addition to configuring the domain name, domain user, domain password parameters, the following pre-requisites should also be fulfilled for this test to report metrics:

    - Make sure that the Citrix XenApp Commands Remoting service is running in the Controller server.
    - The port 2513 must be open on the Controller server in the farm.
15. **SHOW WORKER GROUPS** - Worker groups are collections of XenApp servers, residing in the

same farm, that are managed as a single unit. You can publish applications to a worker group. If you want to know the worker group to which every application accessed by a user has been published, then set this parameter to **Yes**. If both the **SHOW WORKER GROUPS** and **APPSBYNAME** flags are set to **Yes**, the detailed diagnosis of this test will display the worker group name along with the name of the application accessed by the user. By default, this parameter is set to **No**.

16. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, this test will report metrics for every domainname\username. This way, administrators will know which user logged in from which domain. If you want the test to report metrics for every username only, then set this flag to **No**.

18. **ENABLE BROWSER MONITORING** – By default, this flag is set to **No**, indicating that the eG agent does not monitor browser activity on the XenApp server. If this flag is set to **Yes,** then, whenever one/more IE (Internet Explorer) browser instances on the XenApp server are accessed, the detailed diagnosis of the *User sessions* measure will additionally reveal the URL being accessed via each IE instance and the resources consumed by every URL. Armed with this information, administrators can identify the web sites that are responsible for excessive resource usage by an IE instance.

19. **COLLECT EXTENDED METRICS** – By default, this parameter is set to **No**, indicating that the test will report only a standard set of user experience metrics. To enable the test to collect additional metrics per user, set this flag to **Yes**.

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

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

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

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
User sessions:	Represents the current number of sessions for a particular user	Number	A value of 0 indicates that the user is not currently connected to the Citrix server. Use the detailed diagnosis of this measure to know the details of the sessions.
Screen refresh latency - last:	Represents the average client latency for the last request from a user. The latency is measured by the Citrix server based on packets sent to and from each client during a session - this includes network delay plus server side processing delays. The value reported is the average of the last latencies for all the current sessions of a user.	Secs	
Screen refresh latency - avg:	Indicates the average time interval measured at the client between the first step (user action) and the last step (graphical response displayed) of this user's interactions with the server. The value reported is the average of the latencies for all the current sessions of a user.	Secs	This is a measurement of the screen lag that a user experiences while interacting with the XenApp server. In other words, is the latency detected from when the user hits a key unti the response is displayed. Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when interacting with the XenApp server. If both the Screen refresh latency and Client network latency measures report high values, it implies that network slowness is contributing to

Measurement	Description	Measurement Unit	Interpretation
			user-perceived Citrix slowness (i.e., the problem is not due to the Citrix servers, but probably due to the network connection that the user is connecting from - e.g., a wireless WAN). If Screen refresh latency is high and Client network latency is low, this implies that there is a bottleneck in the Citrix stack that is causing user experience to be poor (e.g., overloaded server or virtual platform, slowness in storage, etc.). Slowness can also occur because of client-side processing delays on the receiver end.
Screen refresh latency -deviation:	The latency deviation represents the difference between the minimum and maximum measured latency values for a session. The value reported is the average of the latency deviations for all the current sessions of a user.	Secs	Ideally, the deviation in latencies over a session should be minimum so as to provide a consistent experience for the user. **This measure will be reported only if the COLLECT EXTENDED METRICS flag is set to 'Yes'.**
Memory usage for user's processes:	This value represents the ratio of the resident set size of the memory utilized by the user to the physical memory of the host system, expressed as a percentage. If a user is connected via multiple sessions, the value reported is the sum of all memory utilizations across all the sessions.	Percent	This value indicates the percentage of memory resources that are used up by a specific user. By comparing this value across users, an administrator can identify the most heavy users of the Citrix server. Check the detailed diagnosis to view the offending processes/applications.

Measurement	Description	Measurement Unit	Interpretation
CPU usage for user's processes:	The CPU utilization for a session is the percentage of time that all of the threads/processes of a user session used the processor to execute instructions. If a user is connected via multiple sessions, the value reported is the sum of all CPU utilizations across all the sessions. Also, in multi-processor environments, the average CPU usage per processor is reported as the value of this measure – i.e., if your Citrix server is using an 8-core processor and the total CPU usage of a user across all his/her sessions amounts to 40%, then this measure will report CPU usage as 5 % (40/8 processors = 5).	Percent	This measure serves as a good indicator of CPU usage in load-balanced environments, where the user load is balanced across all processors. Excessive CPU usage by a user can impact performance for other users. This is why, a high value for this measure is a cause for concern. In such cases, check the detailed diagnosis to view the offending processes / applications.
Input bandwidth:	Indicates the average bandwidth used for client to server communications for all the sessions of a user	KB/Sec	
Output bandwidth:	Indicates the average bandwidth used for server to client communications for all the sessions of a user	KB/Sec	
Input line speed:	Indicates the average line speed from the client to the server for all the sessions of a user	KB/Sec	
Output line speed:	Indicates the average line speed from the server to the client for all the sessions of	KB/Sec	

Measurement	Description	Measurement Unit	Interpretation
	a user		
Input compression:	Indicates the average compression ratio for client to server traffic for all the sessions of a user	Number	
Output compression:	Indicates the average compression ratio for server to client traffic for all the sessions of a user	Number	
I/O reads for user's processes:	Indicates the rate of I/O reads done by all processes being run by a user.	KBps	These metrics measure the collective I/O activity (which includes file, network and device I/O's) generated by all the processes being executed by a user. When viewed along with the system I/O metrics reported by the DiskActivityTest, these measures help you determine the network I/O. Comparison across users helps identify the user who is running the most I/O-intensive processes. Check the detailed diagnosis for the offending processes/applications.
I/O writes for user's processes:	Indicates the rate of I/O writes done by all processes being run by a user.	KBps	
Page faults for user's processes:	Indicates the rate of page faults seen by all processes being run by a user.	Faults/Sec	Page Faults occur in the threads executing in a process. A page fault occurs when a thread refers to a virtual memory page that is not in its working set in main memory. If the page is on the standby list and hence already in main memory, or if the page is in use by another process with whom the page is shared, then the page fault will not cause the page to be fetched from disk. Excessive page faults could result in decreased performance. Compare values across users to figure out which user is causing most page faults.

Measurement	Description	Measurement Unit	Interpretation
Handles used by user's processes:	Indicates the total number of handles being currently held by all processes of a user.	Number	A consistent increase in the handle count over a period of time is indicative of malfunctioning of programs. Compare this value across users to see which user is using a lot of handles. Check detailed diagnosis for further information.
Audio bandwidth input:	Indicates the bandwidth used while transmitting sound/audio to this user.	Kbps	Comparing these values across users will reveal which user is sending/receiving bandwidth-intensive sound/audio files over the ICA channel. To minimize bandwidth consumption, you may want to consider *disabling client audio mapping.*
Audio bandwidth output:	Indicates the bandwidth used while receiving sound/audio from this user.	Kbps	
COM bandwidth input:	Indicates the bandwidth used when sending data to this user's COM port.	Kbps	Comparing these values across users will reveal which user's COM port is sending/receiving bandwidth-intensive data over the ICA channel. **This measure will be reported only if the COLLECT EXTENDED METRICS flag is set to 'Yes'.**
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. If bandwidth consumption is too high, you may want to consider disabling client drive mapping on the client device. Client drive mapping allows users logged on to a virtual desktop from a client device to access their local drives transparently from the ICA session. Alternatively, you can conserve bandwidth by even refraining from accessing large files

Measurement	Description	Measurement Unit	Interpretation
			with client drive mapping over the ICA connection. **These measures will be reported only if the COLLECT EXTENDED METRICS flag is set to 'Yes'.**
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	
Speed screen data channel bandwidth input:	Indicates the bandwidth used from this user to the server 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. **These measures will be reported only if the COLLECT EXTENDED METRICS flag is set to 'Yes'.**
Speed screen data channel bandwidth output:	Indicates the bandwidth used from server to this user for data channel traffic.	Kbps	
HDX media stream for flash data bandwidth input:	Indicates the bandwidth used from this user to server 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 server to this user for flash data traffic	Kbps	
HDX media stream for flash v2 data bandwidth input:	Indicates the bandwidth used from this user to server for flash v2 data	Kbps	Comparing the values of these measures across users will reveal which user has been

Measurement	Description	Measurement Unit	Interpretation
	traffic.		transmitting/receiving bandwidth-intensive flash v2 data.
HDX media stream for flash v2 data bandwidth output:	Indicates the bandwidth used from the server to this user for flash v2 data traffic	Kbps	
Speed screen multimedia acceleration bandwidth input	Indicates the bandwidth used from this user to the server for multimedia traffic.	Kbps	Comparing the values of these measures across users will reveal which user has been transmitting/receiving bandwidth-intensive multimedia files. **This measure will be reported only if the COLLECT EXTENDED METRICS flag is set to 'Yes'.**
Speed screen multimedia acceleration bandwidth output	Indicates the bandwidth used from the server to this user for multimedia traffic.	Kbps	
PN bandwidth input:	Indicates the bandwidth used from this user to virtual desktop by Program Neighborhood to obtain application set details.	Kbps	Comparing the values of these measures across users will reveal which user has been transmitting/receiving bandwidth-intensive PN traffic. **These measures will be reported only if the COLLECT EXTENDED METRICS flag is set to 'Yes'.**
PN bandwidth output:	Indicates the bandwidth, used from the virtual desktop to this user by Program Neighborhood to obtain application set details.	Kbps	
CPU time used by user's sessions:	Indicates the percentage of time, across all processors, this user hogged the CPU.	Percent	The CPU usage for user's processes measure indicates the percentage of overall server CPU time that a user is using. For example, if a user is taking up one of the server's CPUs for 100% of the time and there are 8 CPUs on the server, CPU usage for user's processes will be 12.5% (100/800). While 12.5% may seem to be a low number, the fact that the user is taking

Measurement	Description	Measurement Unit	Interpretation
			up one of the CPUs of the server is significant. Hence, CPU time used by user's session measure is a better indicator of CPU usage by users. In the above example, since the user is consuming 100% of one processor, CPU time used by user's session will be 100%. A high value of this measure or a consistent increase in the value of this measure demands attention. Use the detailed diagnosis to know what CPU intensive activities are being performed by the user.
Input bandwidth usage:	Indicates the percentage HDX bandwidth consumed by client to server traffic of this user.	Percent	Compare the value of these measures across users to know which user is consuming the maximum HDX bandwidth.
Output bandwidth usage:	Indicates the percentage HDX bandwidth consumption of this user.	Percent	Compare the value of this measure across users to know which user is consuming the maximum HDX bandwidth.
ThinWire bandwidth input:	Indicates the bandwidth used from client to server for ThinWire traffic.	Kbps	Typically, ICA traffic is comprised of many small packets, as well as a some large packets. Large packets are commonly generated for initial session screen paints and printing jobs, whereas the ongoing user session is principally comprised of many small packets. For the most part, these small packets are the highest priority ICA data called Thinwire. Thinwire incorporates mouse movements and keystrokes. Compare the value of these measures across users to know which user's keystrokes and mouse movements are generating bandwidth-intensive traffic.

Measurement	Description	Measurement Unit	Interpretation
Thinwire bandwith output:	Indicates the bandwidth used from server to client for ThinWire traffic.	Kbps	**This measure will be reported only if the COLLECT EXTENDED METRICS flag is set to 'Yes'.**
Seamless bandwidth input:	Indicates the bandwidth used from client to server for published applications that are not embedded in a session window.	Kbps	Compare the value of these measures across users to know which user is accessing bandwidth-intensive applications that are not in a session window.
Seamless bandwidth output:	Indicates the bandwidth used from server to client for published applications that are not embedded in a session window.	Kbps	**This measure will be reported only if the COLLECT EXTENDED METRICS flag is set to 'Yes'.**
Resource shares:	Indicates the total number of resource shares used by this user.	Number	By comparing the value of this measure across users, you can identify the user who is hogging the resources.
Total bandwidth:	Indicates the total bandwidth usage of the sessions of this user.	Kbps	Compare the value of this measure across users to know which user is consuming the maximum bandwidth.
Total time in session:	Indicates the time that has elapsed since this user logged in.	Mins	Compare the value of this measure across users to know which user has been logged in for the longest time.
Active time in last measure period:	Indicates the percentage of time in the last measurement period during which this user actively used the server.	Percent	Ideally, the value of this measure should be 100%. A low value for this measure denotes a high level of inactivity recently.
Time since last activity:	Indicates the time that has elapsed since this user performed an action on the server.	Mins	A high value for this measure indicates that the user has been idle for a long time. Compare the value of this measure across users to know which user has been idle for the longest time.
Total idle time in session:	Indicates the total time for which this user was idle	Mins	If the value of this measure is the same as the value of the *Total time in*

Measurement	Description	Measurement Unit	Interpretation
	during the session.		*session* measure for a user, it means that the user has been idle throughout the session. If the value of this measure is close to the value of the *Total time in session* measure for a user, it implies that the user has been idle for a long time. If the value of this measure is much lesser than the value of the *Total time in session* measure for a user, it means that the user has been active for most part of the session.
Working set memory for user's processes:	Indicates the current size of the working set of this user's processes	MB	The Working Set is the set of memory pages touched recently by the threads in a process. If free memory in the server is above a threshold, pages are left in the Working Set of a process even if they are not in use. When free memory falls below a threshold, pages are trimmed from Working Sets. If they are needed they will then be soft-faulted back into the Working Set before leaving main memory. If multiple processes are running in the user's session, the memory working set reported is the sum of the working sets for all the user's processes. Comparing the working set across users indicates which user(s) are taking up excessive memory. Check the detailed diagnosis to view the offending processes/applications.
Virtual memory of user's processes	Indicates the amount of virtual memory used by the user's sessions.	MB	Compare the value of this measure across users to know which user is consuming the maximum virtual memory.

Measurement	Description	Measurement Unit	Interpretation
Processes running in the user's session:	Indicates the count of processes running in this user's session.	Number	
Client network latency	Indicates the latency experienced by this user when transmitting/receiving data over the ICA channel.	Secs	This measure represents the network latency detected between the ICA client and the Citrix XenApp server being monitored. If both the Screen refresh latency and Client network latency measures report high values, it implies that network slowness is contributing to user-perceived Citrix slowness (i.e., the problem is not due to the Citrix servers, but probably due to the network connection that the user is connecting from - e.g., a wireless WAN). If Screen refresh latency is high and Client network latency is low, this implies that there is a bottleneck in the Citrix stack that is causing user experience to be poor (e.g., overloaded server or virtual platform, slowness in storage, etc.). Slowness can also occur because of client-side processing delays on the receiver end.
Frame rate:	Indicates the rate at which frames are processed during this user session.	Frames/Sec	FPS is how fast your graphics card can output individual frames each second. It is the most time-tested and ideal measure of performance of a GPU. Higher the value of this measure, healthier is the GPU.
Framehawk frame rate:	Indicates the rate at which frames are processed by the Framehawk virtual channel, if it is enabled for this user session.	Frames/Sec	The Framehawk virtual channel optimizes the delivery of virtual desktops and applications to users on broadband wireless connections, when high packet loss or congestion

Measurement	Description	Measurement Unit	Interpretation
			occurs. A high value is desired for this measure, as it indicates faster delivery of applications to users, which in turn makes for a better user experience. You can compare the value of this measure with that of the Frame rate measure of a user to ascertain whether/not the Framehawk virtual channel has indeed enhanced that user's experience with applications deployed on XenApp. If this comparison reveals that the value of this measure is higher than that of the Frame rate measure, it is a clear indicator of the effectiveness of the Framehawk virtual channel. **Note:** This measure will report the value 0 if Framehawk is not enabled for a user or if the device from which the user is accessing the application does not support Framehawk.
Framehawk network bandwidth:	Indicates the bandwidth consumption of this user session when the Framehawk virtual delivery channel is used.	KB	This is a good measure of the effectiveness of Framehawk in optimizing the bandwidth usage over the virtual delivery channel. A low value is desired for this measure. **Note:** This measure will report the value 0 if Framehawk is not enabled for a user or if the device from which the user is accessing the application does not support Framehawk.
Framehawk latency:	Indicates the latency experienced by this user	Secs	To judge the effectiveness of Framehawk, compare the value of this

Measurement	Description	Measurement Unit	Interpretation
	session when the Framehawk virtual delivery channel is used.		measure with that of the ICA network latency measure for a Framehawk-enabled user. If the comparison reveals a lower value for this measure, it implies that Framehawk has succeeded in minimizing the latencies over the delivery channel. **Note:** This measure will report the value 0 if Framehawk is not enabled for a user or if the device from which the user is accessing the application does not support Framehawk.
Framehawk network loss:	Indicates the percentage of packet loss experienced by this user session when the Framehawk virtual delivery channel is used.	Percent	If the value of this measure is very low, it indicates that Framehawk has been very effective in minimizing the loss ofpackets that typically occur when data is transmitted or received over a channel. **Note:** This measure will report the value 0 if Framehawk is not enabled for a user or if the device from which the user is accessing the application does not support Framehawk.

Note:

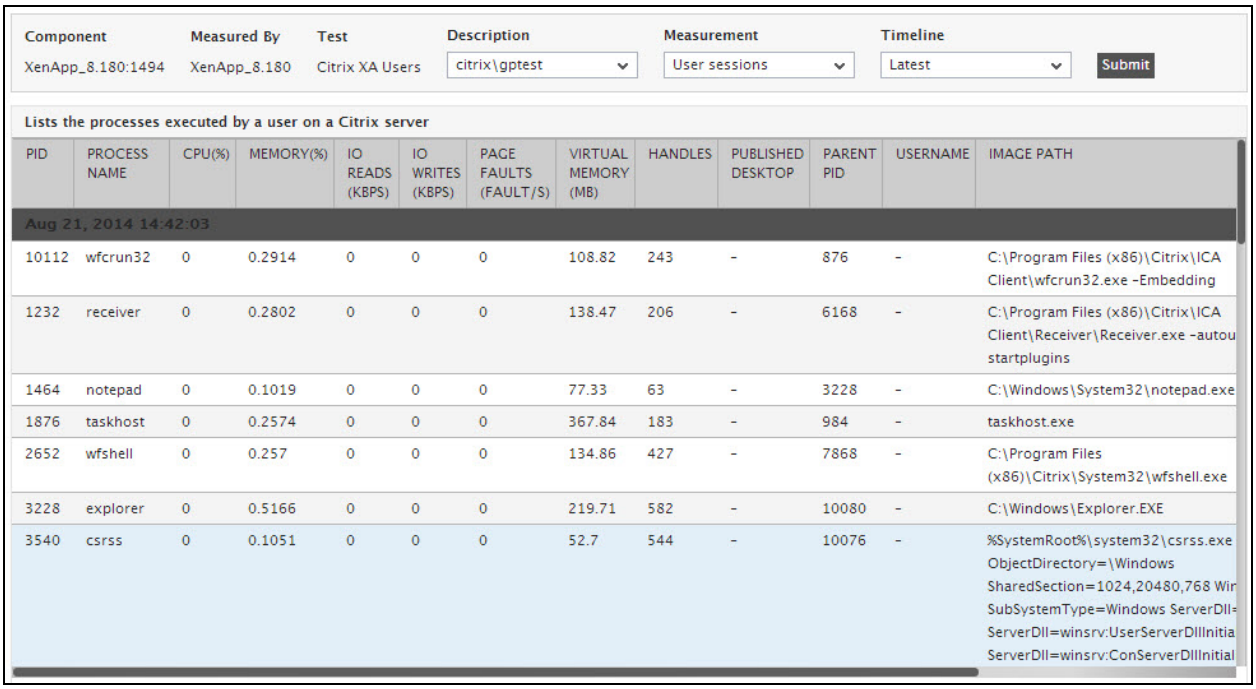
When a Citrix user being monitored by the eG agent logs out of the Citrix server, then the name of the user will not be displayed as a descriptor of the this test in the eG monitor interface.

The detailed diagnosis of the *User sessions* measure (and the *CPU usage* of *user's processes* and *Memory usage of user's processes* measures), if enabled, provides the list of processes executed by a user on the Citrix server, and the CPU and memory utilization of such processes (see Figure 3.27). This information enables the Citrix administrator to identify the processes that are utilizing resources excessively and those that may be leaking memory. In the event of a server overload/memory leak, the Citrix administrator might decide to terminate these processes (see Figure 3.25). In addition, the

detailed diagnosis reveals the location from which each application instance runs (i.e., the **IMAGE PATH**). If multiple versions of an application are published in different locations on the XenApp server and a user runs each of these versions, then the **IMAGE PATH** will indicate the exact application version each process instance corresponds to – resource-hungry versions can thus be identifed. Where one/more instances of the Internet Explorer browser are running, the detailed diagnosis additionally displays the website URL accessed using each IE instance, the domain of every URL, and the website title. In the event of excessive resource usage by an IE instance, this information will shed light on the resource-intensive web site that was being accessed.

Note:

- The eG agent will perform browser activity monitoring only if the **ENABLE BROWSER MONITORING** flag is set to **Yes**.
- The eG agent will monitor browser activity only of the browser being accessed is **Internet Explorer**.



Component	Measured By	Test	Description	Measurement	Timeline	
XenApp_8.180:1494	XenApp_8.180	Citrix XA Users	citrix\gptest	User sessions	Latest	Submit

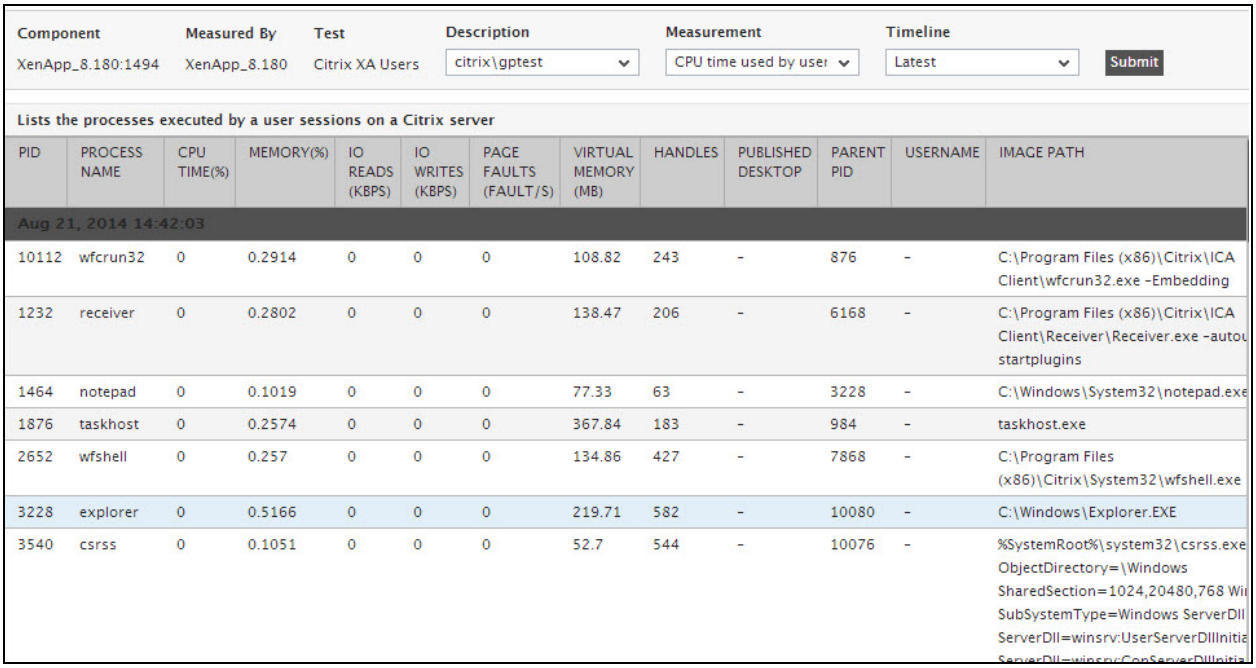
Lists the processes executed by a user on a Citrix server

PID	PROCESS NAME	CPU(%)	MEMORY(%)	IO READS (KBPS)	IO WRITES (KBPS)	PAGE FAULTS (FAULT/S)	VIRTUAL MEMORY (MB)	HANDLES	PUBLISHED DESKTOP	PARENT PID	USERNAME	IMAGE PATH
Aug 21, 2014 14:42:03												
10112	wfcrun32	0	0.2914	0	0	0	108.82	243	-	876	-	C:\Program Files (x86)\Citrix\ICA Client\wfcrun32.exe -Embedding
1232	receiver	0	0.2802	0	0	0	138.47	206	-	6168	-	C:\Program Files (x86)\Citrix\ICA Client\Receiver\Receiver.exe -autou startplugins
1464	notepad	0	0.1019	0	0	0	77.33	63	-	3228	-	C:\Windows\System32\notepad.exe
1876	taskhost	0	0.2574	0	0	0	367.84	183	-	984	-	taskhost.exe
2652	wfshell	0	0.257	0	0	0	134.86	427	-	7868	-	C:\Program Files (x86)\Citrix\System32\wfshell.exe
3228	explorer	0	0.5166	0	0	0	219.71	582	-	10080	-	C:\Windows\Explorer.EXE
3540	csrss	0	0.1051	0	0	0	52.7	544	-	10076	-	%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,20480,768 Wir SubSystemType=Windows ServerDll= ServerDll=winsrv:UserServerDllInitia ServerDll=winsrv:ConServerDllInitial

Figure 3.27: The detailed diagnosis of the User sessions measure

The detailed diagnosis of the *CPU time used by user's sessions measure*, if enabled, provides the list of processes executed by a user on the Citrix server, and the percentage of time for which each process was hogging the CPU. This percentage denotes the total percentage of time the process was using the CPU resources across all the processors that are supported by the XenApp server. This leads you to the exact process that is draining the CPU resources of the server. In addition, the

detailed diagnosis reveals the location from which each application instance runs (i.e., the **IMAGE PATH**). If multiple versions of an application are published in different locations on the XenApp server and a user runs each of these versions, then the **IMAGE PATH** will indicate the exact application version each process instance corresponds to – resource-hungry versions can thus be identifed. Where one/more instances of the Internet Explorer browser are running, the detailed diagnosis additionally displays the website URLs accessed using each IE instance, the domain of every URL, and the website title. In the event of excessive resource usage by an IE instance, this information will shed light on the resource-intensive web site that was being accessed.



Component	Measured By	Test	Description	Measurement	Timeline	
XenApp_8.180:1494	XenApp_8.180	Citrix XA Users	citrix\gptest	CPU time used by user	Latest	Submit

Lists the processes executed by a user sessions on a Citrix server

PID	PROCESS NAME	CPU TIME(%)	MEMORY(%)	IO READS (KBPS)	IO WRITES (KBPS)	PAGE FAULTS (FAULT/S)	VIRTUAL MEMORY (MB)	HANDLES	PUBLISHED DESKTOP	PARENT PID	USERNAME	IMAGE PATH
Aug 21, 2014 14:42:03												
10112	wfcrun32	0	0.2914	0	0	0	108.82	243	-	876	-	C:\Program Files (x86)\Citrix\ICA Client\wfcrun32.exe -Embedding
1232	receiver	0	0.2802	0	0	0	138.47	206	-	6168	-	C:\Program Files (x86)\Citrix\ICA Client\Receiver\Receiver.exe -autou startplugins
1464	notepad	0	0.1019	0	0	0	77.33	63	-	3228	-	C:\Windows\System32\notepad.exe
1876	taskhost	0	0.2574	0	0	0	367.84	183	-	984	-	taskhost.exe
2652	wfshell	0	0.257	0	0	0	134.86	427	-	7868	-	C:\Program Files (x86)\Citrix\System32\wfshell.exe
3228	explorer	0	0.5166	0	0	0	219.71	582	-	10080	-	C:\Windows\Explorer.EXE
3540	csrss	0	0.1051	0	0	0	52.7	544	-	10076	-	%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,20480,768 Wi SubSystemType=Windows ServerDll ServerDll=winsrv:UserServerDllInitia

Figure 3.28: The detailed diagnosis of the CPU time used by user's sessions measure

Note:

- The eG agent will perform browser activity monitoring only if the **ENABLE BROWSER MONITORING** flag is set to **Yes**.
- The eG agent will monitor browser activity only of the browser being accessed is **Internet Explorer**.

3.7.2 Citrix XA Disconnects Test

A user session is terminated when a user logs off from the Citrix/Terminal server or when the session is abruptly interrupted (e.g., due to server, network, or application errors). When a user logs off, all the applications started by the user are terminated. However, when a user disconnects, the applications started by the user will keep running on the server consuming resources. Hence, the

number of disconnected sessions on a Citrix/Terminal server should be kept to a minimum. Abrupt disconnects can significantly impact the end user experience, and hence, it is important to monitor the number of disconnected sessions at any point of time.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results is reported for each Citrix 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 used by the Citrix server
4. **RECONNECTPERIOD** - This parameter is used by the test while computing the value for the **Quick reconnects by users** measure. This measure counts all the users who reconnected to the Citrix server within the short period of time (in minutes) specified against **RECONNECTPERIOD**.
5. **REPORT BY DOMAIN NAME** - By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who disconnected from the server recently. This way, administrators will be able to quickly determine which user belongs to which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.
6. **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**.
7. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

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

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

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total disconnected sessions:	Indicates the total number of sessions that are in the disconnected state.	Number	
New disconnects:	Indicates the number of sessions that were disconnected in the last measurement period.	Number	The detailed diagnosis for this measure indicates the user, session ID, and client type for each newly disconnected session. This information can be used to track whether specific users are being disconnected often
Quick reconnects by users:	Indicates the number of users who reconnected soon after a disconnect.	Number	The detailed diagnosis of this measure, if enabled lists the users who have reconnected quickly.

3.7.3 Citrix XA Logins Test

The **Citrix XA Logins** test monitors the new logins to the Citrix server.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results is reported for each Citrix 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 used by the Citrix server

4. **REPORTUSINGMANAGERTIME** - By default, this flag is set to **Yes**. This indicates that the user login time displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports will be based on the eG manager's time zone by default. Set this flag to **No** if you want the login times displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports to be based on the Citrix server's local time.

5. **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** – A Citrix XenApp server (v6.5) can run in the **controller mode** or the **worker mode**. In the controller mode, the XenApp server can perform all farm management tasks. However, in the worker mode, a XenApp server can only host user sessions.

   By default, the **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWOR** and **CONFIRM PASSWORD** parameters are set to *none*. If the XenApp server being monitored is the **controller** in a farm, then this default value will automatically apply. In other words, in this case, you can leave the values of these parameters as *none*. On the other hand, if the target XenApp server is a **worker** in the farm, then first, you will have to configure the name of the domain in which the XenApp server operates in the **DOMAIN NAME** text box. Then, you need to configure the test with the credentials of a user with **Citrix Farm Administrator** rights, using the **DOMAIN USER** and **DOMAIN PASSWORD** text boxes. Finally, you will have to confirm the **DOMAIN PASSWORD** by retyping it in the **CONFIRM PASSWORD** text box.

   Note:

   If the XenApp server is a worker in the farm, then, in addition to configuring the domain name, domain user, domain password parameters, the following pre-requisites should also be fulfilled for this test to report metrics:

   - Make sure that the Citrix XenApp Commands Remoting service is running in the Controller server.
   - The port 2513 must be open on the Controller server in the farm.

6. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user session that logged out. This default setting ensures that administrators are able to quickly determine the domains to which the users who logged out belonged. You can set this flag to **No** if you want detailed diagnosis to display only the username of the users who logged out.

7. **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**.

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.

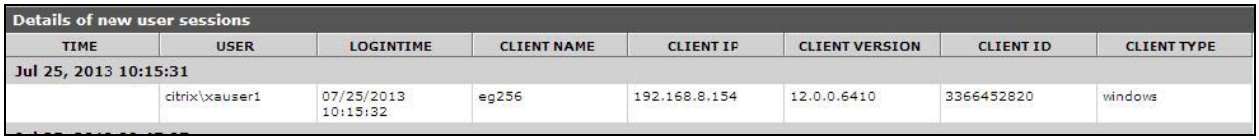
   The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

   - 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 made by the test

Measurement	Description	Measurement Unit	Interpretation
New logins:	Indicates the number of new logins to the Citrix server in the last measurement period.	Number	A consistent zero value could indicate a connection issue. You can use the detailed diagnosis of this test to know which users logged in recently.
Percent new logins:	Indicates the percentage of current sessions that logged in during the last measurement period.	Percent	
Sessions logging out:	Indicates the number of sessions that logged out.	Number	If all the current sessions suddenly log out, it indicates a problem condition that requires investigation. The detailed diagnosis of this measure lists the sessions that logged out.

Using the detailed diagnosis of the *New logins* measure, you can not only identify the users who logged in recently, but can also figure out when each user logged in and from which client machine.

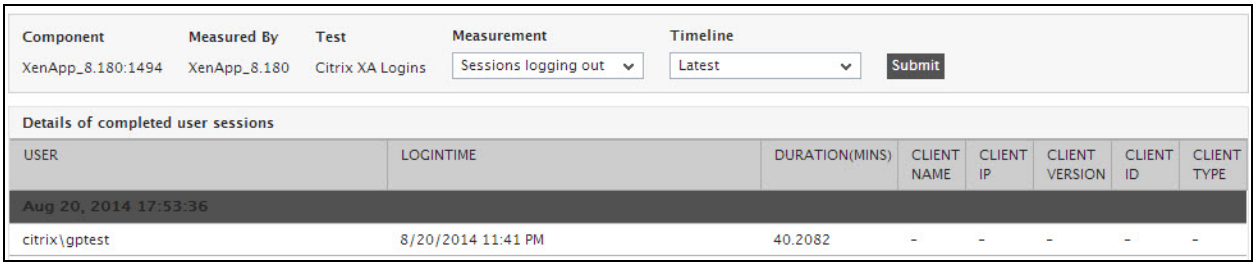


Details of new user sessions

TIME	USER	LOGINTIME	CLIENT NAME	CLIENT IP	CLIENT VERSION	CLIENT ID	CLIENT TYPE
Jul 25, 2013 10:15:31							
	citrix\xauser1	07/25/2013 10:15:32	eg256	192.168.8.154	12.0.0.6410	3366452820	windows

Figure 3.29: The detailed diagnosis of the New logins measure

With the help of the detailed diagnosis of the *Sessions logged out* measure, you can identify the users who logged out, when every user logged in and from which client machine, and the duration of each user's session. Abnormally long sessions on the server can thus be identified.



Component	Measured By	Test	Measurement	Timeline	
XenApp_8.180:1494	XenApp_8.180	Citrix XA Logins	Sessions logging out	Latest	Submit

Details of completed user sessions

USER	LOGINTIME	DURATION(MINS)	CLIENT NAME	CLIENT IP	CLIENT VERSION	CLIENT ID	CLIENT TYPE
Aug 20, 2014 17:53:36							
citrix\gptest	8/20/2014 11:41 PM	40.2082	-	-	-	-	-

Figure 3.30: The detailed diagnosis of the Sessions logged out measure

3.7.4 Citrix XA Sessions Test

This test reports performance statistics related to Citrix user sessions.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **REPORTUSINGMANAGERTIME** - By default, this flag is set to **Yes**. This indicates that the user login time displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports will be based on the eG manager's time zone by default. Set this flag to **No** if you want the login

times displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports to be based on the Citrix server's local time.

5. **IGNORE DOWN SESSION IDS –** By default, this parameter is set to 65536,65537,65538 – these are nothing but the default ports at which the listener component listens. If any of these ports go down, then by default, this test will not count any of the sessions that failed when attempting to connect to that port as a **Down session**. You can override this default setting by adding more ports or by removing one/more existing ports.

6. **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** – A Citrix XenApp server (v6.5) can run in the **controller mode** or the **worker mode**. In the controller mode, the XenApp server can perform all farm management tasks. However, in the worker mode, a XenApp server can only host user sessions.

   By default, the **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** parameters are set to *none*. If the XenApp server being monitored is the **controller** in a farm, then this default value will automatically apply. In other words, in this case, you can leave the values of these parameters as *none*. On the other hand, if the target XenApp server is a **worker** in the farm, then first, you will have to configure the name of the domain in which the XenApp server operates in the **DOMAIN NAME** text box. Then, you need to configure the test with the credentials of a user with **Citrix Farm Administrator** rights, using the **DOMAIN USER** and **DOMAIN PASSWORD** text boxes. Finally, you will have to confirm the **DOMAIN PASSWORD** by retyping it in the **CONFIRM PASSWORD** text box.

   Note:

   If the XenApp server is a worker in the farm, then, in addition to configuring the domain name, domain user, domain password parameters, the following pre-requisites should also be fulfilled for this test to report metrics:

   - Make sure that the Citrix XenApp Commands Remoting service is running in the Controller server.
   - The port 2513 must be open on the Controller server in the farm.

7. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who logged into the Citrix server. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.

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 made by the test

Measurement	Description	Measurement Unit	Interpretation
Established sessions:	Indicates the number of active Citrix user sessions currently on the server.	Number	This measure gives an idea of the server workload in terms of active sessions. Tracking the number of active sessions with time, a Citrix administrator can obtain information that can help him/her plan the capacity of their Citrix environment. The detailed diagnosis capability, if enabled, lists the active and inactive sessions on the Citrix server.
Idle sessions:	Indicates the number of sessions that are initialized and are currently ready to accept connections.	Number	To optimize the performance of a server, two default (idle) sessions are initialized before any client connections are made. For performance reasons, the number of idle sessions should be less than ten. Note that this test does not differentiate between RDP and ICA sessions.
Connected sessions:	Indicates the current number of sessions that are connected, but no user has logged on to	Number	A consistent increase in the value of this measure could indicate that users are having trouble logging in. Further investigation may hence be

Measurement	Description	Measurement Unit	Interpretation
	the server.		required. Note that this test does not differentiate between RDP and ICA sessions.
Connecting sessions:	Indicates the number of sessions that are in the process of connecting.	Number	A very high value for this measure indicates a problem with the session or connection. Note that this test does not differentiate between RDP and ICA sessions.
Disconnected sessions:	Indicates the number of sessions from which users have disconnected, but which are still active and can be reconnected.	Number	Too many disconnected sessions running indefinitely on a Citrix server cause excessive consumption of the server resources. To avoid this, a session limit is typically configured for disconnected sessions on the Citrix server. When a session limit is reached for a disconnected session, the session ends, which permanently deletes it from the server. Note that this test does not differentiate between RDP and ICA sessions.
Listen sessions:	Indicates the current number of sessions that are ready to accept connections.	Number	Note that this test does not differentiate between RDP and ICA sessions.
Shadow sessions:	Indicates the current number of sessions that are remotely controlling other sessions.	Number	A non-zero value for this measure indicates the existence of shadow sessions that are allowed to view and control the user activity on another session. Such sessions help in troubleshooting/resolving problems with other sessions under their control.

Measurement	Description	Measurement Unit	Interpretation
Down sessions:	Indicates the current number of sessions that could not be initialized or terminated.	Number	Ideally, the value of this measure should be 0. By default, if sessions to any of these ports – 65536, 65537, 65538 – could not be initialized or terminated, they will not be counted as a 'down session'.
Init sessions:	Indicates the current number of sessions that are initializing.	Number	A high value for this measure could indicate that many sessions are currently experiencing initialization problems.

The detailed diagnosis capability of the *Active sessions* measure, if enabled, lists the active and inactive sessions on the Citrix server.

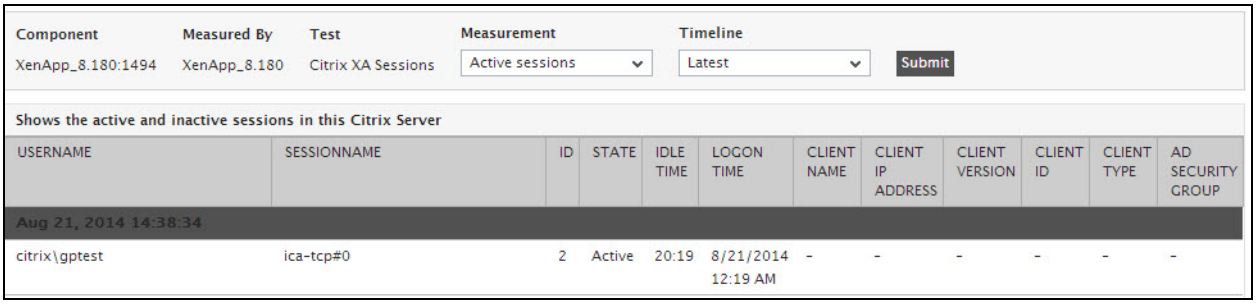


Figure 3.31: The detailed diagnosis of the Active sessions measure of a Citrix server

3.7.5 Citrix XA Receivers Test

If a user complains of slowness when accessing applications/dekstops launched on a Citrix server, administrators may instantly want to know which type of client device that user is connecting from – is it a mobile phone? a laptop? a tablet? what is its IP address? what is its version? This knowledge will ease the troubleshooting pains of administrators as it will clearly indicate if the slowdown occurred owing to the usage of an unsupported or an outdated device. To obtain this knowledge, administrators can use the **Citrix Receivers** test. With the help of this test, administrators can identify the client devices that are connecting via Citrix Receiver, determine which user is logging into the Citrix environment using which device, and in the process, figure out if any device-related issues are contributing to a user's unsatisfactory experience with Citrix.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every client type/client version auto-discovered

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 Citrix server.
4. **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** – A Citrix XenApp server (v6.5) can run in the **controller mode** or the **worker mode**. In the controller mode, the XenApp server can perform all farm management tasks. However, in the worker mode, a XenApp server can only host user session

   By default, the **DOMAIN NAME, DOMAIN USER, DOMAIN PASSWORD**, and **CONFIRM PASSWORD** parameters are set to *none*. If the XenApp server being monitored is the **controller** in a farm, then this default value will automatically apply. In other words, in this case, you can leave the values of these parameters as *none*. On the other hand, if the target XenApp server is a **worker** in the farm, then first, you will have to configure the name of the domain in which the XenApp server operates in the **DOMAIN NAME** text box. Then, you need to configure the test with the credentials of a user with **Citrix Farm Administrator** rights, using the **DOMAIN USER** and **DOMAIN PASSWORD** text boxes. Finally, you will have to confirm the **DOMAIN PASSWORD** by retyping it in the **CONFIRM PASSWORD** text box.

   Note:

   If the XenApp server is a worker in the farm, then, in addition to configuring the domain name, domain user, domain password parameters, the following pre-requisites should also be fulfilled for this test to report metrics:

   - Make sure that the Citrix XenApp Commands Remoting service is running in the Controller server.
   - The port 2513 must be open on the Controller server in the farm.
5. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who logged into the Citrix server. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.

6. **REPORT BY RECEIVER TYPE -** By default, this flag is set to **No**. This implies that by default, this test will report one set of metrics for every client version. To make sure that the test reports metrics for each client type instead, set this flag to **Yes.**
7. **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
Users connected from this type:	Indicates the number of users who are currently connected to Citrix via devices of this type/version.	Number	Use the detailed diagnosis of this measure to know which user connected via devices of a particular type/version.

3.7.6 Citrix Clients Test

This test measures the client connections to and from a Citrix server. This test is disabled by default.

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 : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server
4. **SERVERIP** - By default, the **SERVERIP** field will display the IP address of the Citrix server.
5. **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
Current connections:	The number of TCP connections currently established by clients to the Citrix server	Number	This measure directly indicates the loading on the Citrix server from clients. Typically one connection is established per active session to the Citrix server.
New connections added:	The number of new TCP connections initiated by clients to the Citrix server during the last measurement period	Number	Tracking the new connections over time can provide an indication of when clients login to the Citrix server. A spurt of connections and disconnections may be indicative of sporadic failures of the Citrix server.
Old connections removed:	The number of TCP	Number	A large number of sudden

Measurement	Description	Measurement Unit	Interpretation
	connections that were removed because the clients may have disconnected from the Citrix server during the last measurement period		connection drops may be early warning indicators of problems with the Citrix server.
Avg duration of current connections:	The average time from when a connection is established to when the corresponding connection is disconnected. The duration of a connection is measured from its start time to the current time. The accuracy of this measurement is limited by the frequency at which this test is run.	Secs	This value can provide an indicator of how long clients stay connected to a Citrix server. This information together with the number of simultaneous clients can be useful for capacity planning in Citrix environments (i.e., how to size the Citrix server). The detailed diagnosis capability, if enabled, lists the clients currently connected to the Citrix server.

3.7.7 ICA Client Access Test

A Citrix environment is a shared environment in which multiple users connect to a Citrix XenApp server from remote terminals using the ICA protocol. One of the key factors influencing user experience in such an environment is the latency seen by the users when connecting to the server. High network latencies or packet losses during transmission can cause significant slow-downs in request processing by the server. Hence, monitoring latencies between the server and individual client terminals is important.

The IcaClient test is executed by the eG agent on a Citrix XenApp server. This test auto-discovers the users who are currently logged on to the server and the IP address from which they are connecting to the Citrix server. For each user, the test monitors the quality of the link between the client and the Citrix server.

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 Citrix server may regard a user session as active,

even though the network link connecting the user terminal to the Citrix server has failed. The IcaClientTest alerts administrators to such situations.

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 Citrix XenApp server

Agent deploying the test : An internal agent

Outputs of the test : One set of outputs for every user currently connected to the Citrix server

Configurable parameters for this 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
4. **DISPLAYDOMAIN** - By default, the **DISPLAYDOMAIN** flag is set to **Yes;** this indicates that the **ICA Client Access** test, by default, will report metrics for every domainname\username who is currently connected to the server. This way, administrators can quickly figure out which user is connecting to the server from which domain. You can set this flag to **No** to ensure that this test reports metrics for each username only.
5. **PACKETSIZE** - The size of packets used for the test (in bytes)
6. **PACKETCOUNT** – The number of packets exchanged between the Citrix server and the user terminal during the test
7. **TIMEOUT** - How long after transmission should a packet be deemed lost (in seconds)
8. **PACKETINTERVAL** - Represents the interval (in milliseconds) between successive packet transmissions during the execution of this test.
9. **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 the Citrix server. 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 the network connection between a user terminal and the Citrix server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Number of user 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 Citrix server.
Avg network latency:	Indicates the average delay between transmission of a request by the agent on a Citrix server and receipt of the response back from the 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 Citrix server.
Min network latency:	Indicates the minimum delay between transmission of a request by the agent on a Citrix server 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 server and a user's terminal.
Packet loss:	Indicates the percentage of packets lost during data exchange between the Citrix server 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 Citrix server.

Note:

- If the same user is connecting to the Citrix server from multiple client terminals, the value of the *Number of user sessions*, *Avg network latency*, and *Packet loss* measures will be averaged across all the sessions of that user. The *Min network latency* 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.
- By default, the ICA Client Access test spawns a maximum of one thread at a time for monitoring each of the ICA connections to a Citrix server. Accordingly, the **MaxIcaClientThreads** parameter in the **eg_tests.ini** file (in the **<EG_INSTALL_DIR>\manager\config** directory) is set to 1 by default. In large Citrix farms however, numerous concurrent users attempt to connect to the Citrix server from multiple remote client terminals. To enhance the efficiency of the test and to make sure that it scales to monitor the large number of ICA connections to the Citrix server, you might want to consider increasing the value of the **MaxIcaClientThreads** parameter. If this parameter is set to say, 15, then, it implies that the test will spawn a maximum of 15 threads at one shot, thus monitoring 15 ICA connections to the Citrix server, simultaneously.
- Sometimes, the ICA Client Access test may not work on Citrix XenApp v6.5. This is because, some installations of Citrix XenApp v6.5 may not support the performance object named **ICA Session**, which the test uses for reporting metrics. In such cases, follow the steps given below to enable the **ICA Session** performance object and its counters:
- Login to the Windows system that hosts the Citrix XenApp server v6.5.
- Open the command prompt as **Run as administrator**.
- Issue the following command at the prompt:

  regsvr32 c:\windows\system32\icaperf.dll

3.7.8 Wyse Terminals Test

Wyse thin clients are secure access devices that provide a simpler and easier way to deliver the productivity and application flexibility of a PC without the PC downside.

Users can connect to a Citrix server/server farm from a Wyse terminal to access critical applications. Whenever a user complaints of issues with his/her terminal, you can use this test to figure out which terminal the user is connecting from, whether that terminal is up and running, and if so, for how long.

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 Citrix XenApp server

Agent deploying the test : An internal agent

Outputs of the test : One set of outputs for every Wyse terminal user currently connected to the Citrix server

Configurable parameters for this 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
4. **SNMPPORT** - The port number through which the Wyse terminal exposes its SNMP MIB. The default value is 161.
5. **SNMPVERSION** – By default, the eG agent supports SNMP version 1. Accordingly, the default selection in the **SNMPVERSION** list is **v1**. However, if a different SNMP framework is in use in your environment, say SNMP **v2** or **v3**, then select the corresponding option from this list.
6. **SNMPCOMMUNITY** – The SNMP community name that the test uses to communicate with the Cisco router. This parameter is specific to SNMP **v1** and **v2** only. Therefore, if the **SNMPVERSION** chosen is **v3**, then this parameter will not appear.
7. **USERNAME** – This parameter appears only when **v3** is selected as the **SNMPVERSION**. SNMP version 3 (SNMPv3) is an extensible SNMP Framework which supplements the SNMPv2 Framework, by additionally supporting message security, access control, and remote SNMP configuration capabilities. To extract performance statistics from the MIB using the highly secure SNMP v3 protocol, the eG agent has to be configured with the required access privileges – in other words, the eG agent should connect to the MIB using the credentials of a user with access permissions to be MIB. Therefore, specify the name of such a user against the **USERNAME** parameter.
8. **AUTHPASS** – Specify the password that corresponds to the above-mentioned **USERNAME**. This parameter once again appears only if the **SNMPVERSION** selected is **v3**.
9. **CONFIRM PASSWORD** – Confirm the **AUTHPASS** by retyping it here.
10. **AUTHTYPE** – This parameter too appears only if **v3** is selected as the **SNMPVERSION**. From the **AUTHTYPE** list box, choose the authentication algorithm using which SNMP v3 converts the specified **USERNAME** and **PASSWORD** into a 32-bit format to ensure security of SNMP transactions. You can choose between the following options:
    - **MD5** – Message Digest Algorithm
    - **SHA** – Secure Hash Algorithm
11. **ENCRYPTFLAG** – This flag appears only when **v3** is selected as the **SNMPVERSION**. By default,

the eG agent does not encrypt SNMP requests. Accordingly, the **ENCRYPTFLAG** is set to **NO** by default. To ensure that SNMP requests sent by the eG agent are encrypted, select the **YES** option.

12. **ENCRYPTTYPE** – If the **ENCRYPTFLAG** is set to **YES**, then you will have to mention the encryption type by selecting an option from the **ENCRYPTTYPE** list. SNMP v3 supports the following encryption types:
    - **DES** – Data Encryption Standard
    - **AES** – Advanced Encryption Standard
13. **ENCRYPTPASSWORD** – Specify the encryption password here.
14. **CONFIRM PASSWORD** – Confirm the encryption password by retyping it here.
15. **TIMEOUT** - Specify the duration (in seconds) within which the SNMP query executed by this test should time out in the **TIMEOUT** text box. The default is 10 seconds.
16. **DATA OVER TCP –** By default, in an IT environment, all data transmission occurs over UDP. Some environments however, may be specifically configured to offload a fraction of the data traffic – for instance, certain types of data traffic or traffic pertaining to specific components – to other protocols like TCP, so as to prevent UDP overloads. In such environments, you can instruct the eG agent to conduct the SNMP data traffic related to the equalizer over TCP (and not UDP). For this, set the **DATA OVER TCP** flag to **Yes**. By default, this flag is set to **No**.
17. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

    The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:
    - The eG manager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Uptime of Wyse	Indicates how long the	Secs	A low reported by this measure

Measurement	Description	Measurement Unit	Interpretation
terminal:	Wyse terminal from which this user is connecting has been up and running.		could indicate that the Wyse terminal has rebooted recently.

The detailed diagnosis of the *Uptime of Wyse terminal* measure reveals the name, serial number, IP address, and MAC address of the Wyse terminal from which the user is currently connecting to the Citrix server.

Time	SystemName	SystemDescription	SerialNumber	IP	MAC
Feb 10, 2009 15:05:18					
	wt0080647cab0	v10L 6.1.0_23_0	0FYDH5001970	192.168.10	0:80:64:74:7c:ab0

Figure 3.32: The detailed diagnosis of the Uptime of Wyse terminal measure

3.7.9 WEM Startup Details Test

One of the common reasons for poor user logon experience in a Citrix XenApp/XenDesktop environment is the delay in profile loading and group policy application. Using Citrix Workspace Environment Management (WEM), this delay can be greatly minimized!

Citrix WEM uses intelligent resource management and Profile Management technologies to provide the best logon experience to users in Citrix XenApp and XenDesktop deployments.

Figure 1 depicts the architecture of Citrix WEM. The WEM Administration Console is where policies are defined and managed, resources are created and assigned, and users are authorized. The settings so defined are communicated to a WEM Broker, which stores the same in a SQL server backend. WEM Agents are deployed on VDAs or physical Windows devices. These agents communicate with the WEM Broker and enforce the settings you configured. An Active Directory server is used to push the settings to users.

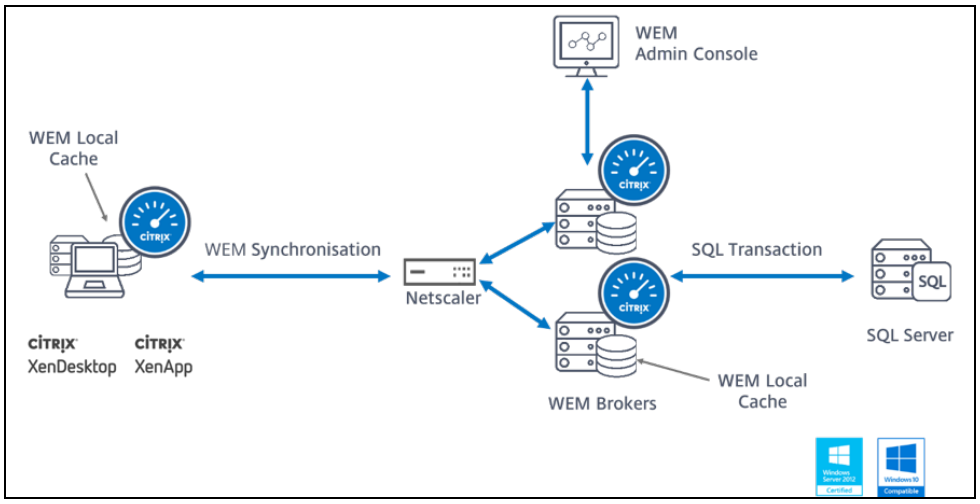


Figure 3.33: Architecture of Citrix WEM

Typically, the WEM agents offload the critical logon processing steps – eg., group policy application, logon script execution, drive/printer mapping, etc. – and perform them after the logon, thus significantly improving logon speed (see Figure 2).

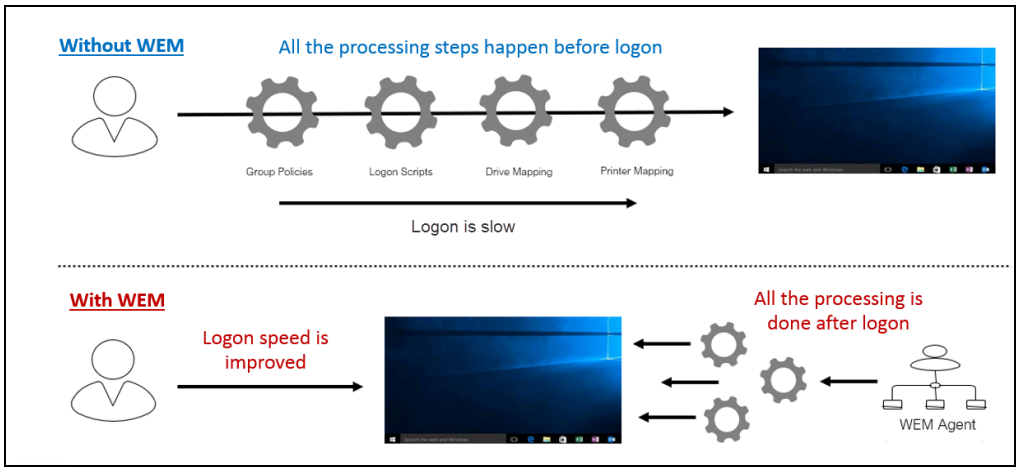


Figure 3.34: How Citrix WEM helps minimize logon time

This is why, where WEM is employed, user logons will be quick and hassle-free. However, if WEM clients – i.e., the WEM agents - experience delays or errors in logon processing, it can cause serious performance issues post logon. In other words, user profile loading, logon script execution, drive mapping etc., can become very slow. Because of such issues, a Citrix user will be unable to access the application/desktop profile, even after logging in quickly.

Therefore, to assure users of a high quality experience with Citrix at all times, administrators of WEM-enabled environments should continuously monitor the WEM processing times on the clients. This is exactly where the WEM Startup Details test helps!

For each user of a desktop OS machine (provisioned on a XenApp server) that hosts a WEM agent, this test reports on the overall WEM processing duration, the time taken by the WEM agent to perform initial processing (this includes tasks such as detecting and reading initial configuration, reading SQL configuration, etc.), and the time taken by the WEM agent to perform main processing (this includes processing of main instructions - eg., mapping network/virtual drives, processing environmental variables, application launching etc.). In the process, administrators are proactively alerted to a delay in WEM processing. The root-cause of the delay is also accurately pinpointed - is it because WEM agent took too long to perform initial processing? or was too much time spent on processing the main instructions? if initial processing was delayed, then was the delay due to a bottleneck when the WEM agent was reading the initial configuration? or was it due to a delay when reading other settings such as SQL configuration, environmental settings, kiosk settings, etc.? if main processing was delayed, then was the delay because the WEM agent was slow detecting the OS/agent log settings? or was it owing to a delay in processing the critical instructions such as network drive mapping, virtual drive mapping, external task processing, application launching, etc? Errors in processing that could be slowing down WEM start-up are also highlighted, so that administrators can easily rectify them. This way, the **WEM Startup Details** test promptly captures and reports performance bottlenecks on WEM clients that can impact overall user experience with Citrix XenApp, thus prompting administrators to rapidly initiate remedial measures.

Target of the test : A Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each user who is currently logged into the desktop OS machines hosting a WEM agent

Configurable parameters for the test

Parameters	Description
Test period	How often should the test be executed
Host	The host for which the test is to be configured. By default, this is portal.office.com
Port	Specify the port at which the configured host listens
Report by Domain Name	By default, this flag is set to **Yes**. This implies that by default, the *domainname**username* of each user logged into a desktop OS machine, will be displayed as a descriptor of this test. This way, administrators will be able to quickly determine which user logged into the desktop from which domain. If you want the only the *username* of these users (and not domain name\username) to be displayed as descriptors, set this flag to **No**.

Parameters	Description
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 the test runs if no problems are reported, 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.
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 enabled/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
Information messages	Indicates the number of information messages logged in the event logs during WEM processing for this user.	Number	
Warnings	Indicates the number of warning messages logged in the WEM logs during WEM processing for this user.	Number	Ideally, the value of this measure should be 0. If a non-zero value is reported, use the detailed diagnosis of this measure to view the warning messages logged in the WEM logs.
Errors	Indicates the number of errors messages logged in the logs during WEM processing for this user.	Number	Ideally, the value of this measure should be 0. A non-zero value is a cause for concern, as it indicates that one/more errors have occurred. You may want to use the detailed diagnosis of this measure to know what errors occurred.

Measurement	Description	Measurement Unit	Interpretation
Total WEM processing duration	Indicates the total time taken for WEM processing for this user.	Seconds	A consistent rise in the value of this measure could indicate a bottleneck in WEM processing for that user. To know where the bottleneck lies, compare the value of the *Total init processing duration* and *WEM agent processing duration* metrics. This will indicate where WEM agent spent maximum time - in starting up? or in post logon activities (eg., drive mapping, printer mapping, etc.)?
Total Init processing duration	Indicates the total time taken by WEM to perform initial processing for this user.	Seconds	The init processing duration is the time taken by the WEM agent to start-up. This includes the time taken by the WEM agent to detect the start-up settings and then read its configuration. If the value of this measure is abnormally high, it could indicate a delay in WEM agent start-up. To diagnose the root-cause of the delay, compare the value of all metrics grouped under the section *Init Processing Breakup* in the **Layers** tab page of the eG monitoring console. This will point you to where the bottleneck lies - did the agent take too long to detect start-up settings? did it take a long time to read the start-up configuration? or did the WEM agent experience any slowness when reading SQL configuration, Microsoft USV settings, environmental settings, system utilities settings, system monitoring settings, or kiosk settings?
WEM agent processing	Indicates the time taken by the WEM agent to perform post-logon activities for this user.	Seconds	The WEM agent processing duration is the sum of the time taken by the WEM agent to process the instructions it detected and read during the init processing/start-up stage, and the time

Measurement	Description	Measurement Unit	Interpretation
			taken to perform post-logon tasks such as virtual drives mapping, network drives mapping, registry entry processing, etc. If the value of this measure is abnormally high, it could indicate a delay in processing by the WEM agent. To diagnose the root-cause of the delay, compare the value of all metrics grouped under the section *WEM Agent Processing Breakup* in the **Layers** tab page of the eG monitoring console. This will point you to where the bottleneck lies - did the agent take too long to process the initial configuration it read? did it take a long time to map virtual drives / network drives? did processing of environment variables, registry entries, printers, ports, DNS, file system operations, file associations, or ini file operations take too long? was external task processing slow?
Initial processing duration	Indicates the time taken by the WEM agent to detect desktop settings - i.e., detect the IP address, version, and OS of the desktop, the user desktop folder, the user taskbar, icons, shortcuts, etc.	Seconds	If the *Total init processing* duration measure reports an abnormally high value, then compare the value of the *Initial processing duration* measure with all other time values reported by the *Init Processing Breakup* section in the **Layers** tab page of the eG monitoring console, to figure out if a delay in discovering desktop settings is what caused initial processing by the WEM processing to slow down.
Reading agent configuration processing duration	Indicates the time taken by the WEM agent to read its initial configuration .	Seconds	If the *Total init processing* duration measure reports an abnormally high value, then compare the value of the *Reading agent configuration processing duration* measure with all other time values reported by the *Init*

Measurement	Description	Measurement Unit	Interpretation
			Processing Breakup section in the **Layers** tab page of the eG monitoring console to figure out if a delay in reading the configuration is what caused initial processing by WEM to slow down.
SQL configuration setting duration	Indicates the time taken by the WEM agent to read the SQL configuration settings for this user.	Seconds	If the *Total init processing* duration measure reports an abnormally high value, then compare the value of the *SQL configuration setting duration* measure with all other time values reported by the *Init Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in reading the SQL configuration is what caused initial processing by the WEM agent to slow down. If the comparative analysis reveals that the delay is indeed owing to a bottleneck when reading the SQL configuration, then use the detailed diagnosis of this measure to view the messages pertaining to SQL configuration reads that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to SQL configuration reads is revealed, accurately pointing you to the exact step/instruction that took the longest time.
Microsoft USV setting duration	Indicates the time taken by the WEM agent to read the Microsoft USV settings for this user.	Seconds	If the *Total init processing* duration measure reports an abnormally high value, then compare the value of the *Microsoft USV setting duration* measure with all other time values reported by the *Init Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in reading the Microsoft USV settings is what caused initial processing by the WEM agent to slow down.

Measurement	Description	Measurement Unit	Interpretation
			If the comparative analysis reveals that the delay is indeed owing to a bottleneck when reading the Microsoft USV settings, then use the detailed diagnosis of this measure to view the messages pertaining to Microsoft USV setting reads that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to Microsoft USV setting reads is revealed, accurately pointing you to the exact step/instruction that took the longest time.
Environmental setting duration	Indicates the time taken by the WEM agent to read the environmental settings for this user.	Seconds	If the *Total init processing* duration measure reports an abnormally high value, then compare the value of the *Environmental setting duration* measure with all other time values reported by the *Init Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in reading the environmental settings is what caused initial processing by the WEM agent to slow down. If the comparative analysis reveals that the delay is indeed owing to a bottleneck when reading the environmental settings, then use the detailed diagnosis of this measure to view the messages pertaining to environmental setting reads that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to environmental setting reads is revealed, accurately pointing you to the exact step/instruction that took the longest time.
System utilities	Indicates the time taken	Seconds	If the *Total init processing* duration

Measurement	Description	Measurement Unit	Interpretation
setting duration	by the WEM agent to read the system utilities settings for this user.		measure reports an abnormally high value, then compare the value of the *System utilities setting duration* measure with all other time values reported by the *Init Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in reading the system utilities settings is what caused initial processing by the WEM agent to slow down. If the comparative analysis reveals that the delay is indeed owing to a bottleneck when reading the system utilities settings, then use the detailed diagnosis of this measure to view the messages related to reading the system utilities settings that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to system utilities setting reads is revealed, accurately pointing you to the exact step/instruction that took the longest time.
System monitoring setting duration	Indicates the time taken by the WEM agent to read the system monitoring settings for this user.	Seconds	If the *Total init processing* duration measure reports an abnormally high value, then compare the value of the *System monitoring setting duration* measure with all other time values reported by the *Init Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in reading the system monitoring settings is what caused initial processing by the WEM agent to slow down. If so, then you can use the detailed diagnosis of this measure to view the messages related to reading the system monitoring settings that are logged in the

Measurement	Description	Measurement Unit	Interpretation
			Citrix WEM Agent.log. The time taken to process each instruction pertaining to system monitoring setting reads is revealed, accurately pointing you to the exact step/instruction that took the longest time.
Kiosk setting duration	Indicates the time taken by the WEM agent to read the kiosk settings for this user.	Seconds	If the *Total init processing* duration measure reports an abnormally high value, then compare the value of the *Kiosk setting duration* measure with all other time values reported by the *Init Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in reading the kiosk settings is what caused initial processing by the WEM agent to slow down. If the comparative analysis reveals that the delay is indeed owing to a bottleneck when reading the kiosk settings, then use the detailed diagnosis of this measure to view the messages related to reading the kiosk settings that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to kiosk setting reads is revealed, accurately pointing you to the exact step/instruction that took the longest time.
WEM agent main log initial processing duration	Indicates the time taken by the WEM agent to detect WEM broker settings, the WEM agent log settings, and desktop settings, before proceeding to process the main instructions (eg., network drive and	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *WEM agent main log initial processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in

Measurement	Description	Measurement Unit	Interpretation
	virtual drive mapping, application launching, registry entry processing, etc.) for this user.		processing desktop/logging/broker settings is what caused the WEM processing to slow down.
Environment variables processing duration	Indicates the time taken by the WEM agent to process environment variables for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *Environment variables processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in processing environment variables is what caused WEM processing to slow down. If this processing is indeed found to be the reason for slowing down the WEM agent, then use the detailed diagnosis of this measure to identify the exact environment variables responsible for the delay.
Registry entry processing duration	Indicates the time taken by the WEM agent to process registry entries for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *Registry entry processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in processing registry entries is what caused WEM processing to slow down.
Networks processing duration	Indicates the time taken by the WEM agent to process network drive mappings for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *Networks processing duration* measure with all other time values reported by the *WEM*

Measurement	Description	Measurement Unit	Interpretation
			Agent Processing Breakup section in the **Layers** tab page of the eG monitoring console to figure out if a delay in processing network drive mappings is what caused WEM processing to slow down. If the comparative analysis reveals that the processing of network drive mappings is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure for messages pertaining to network drive mapping that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to network drive mapping is revealed, accurately pointing you to the exact step/instruction that took the longest time.
Virtual drives processing duration	Indicates the time taken by the WEM agent to process virtual drive mappings for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *Virtual drives processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in processing virtual drive mappings is what caused WEM processing to slow down. If the comparative analysis reveals that the processing of virtual drive mappings is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to virtual drive mapping that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to virtual drive mapping is revealed, accurately pointing you to the exact

Measurement	Description	Measurement Unit	Interpretation
			step/instruction that took the longest time.
Printers processing duration	Indicates the time taken by the WEM agent to process printers for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *Printers processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in printer processing is what caused WEM processing to slow down. If the comparative analysis reveals that the processing of printers is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to printer processing that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to printer processing is revealed, accurately pointing you to the exact step/instruction that took the longest time.
Port processing duration	Indicates the time taken by the WEM agent to process ports for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *Port processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in processing ports is what caused WEM processing to slow down. If the comparative analysis reveals that the processing of ports is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to port

Measurement	Description	Measurement Unit	Interpretation
			processing that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to port processing is revealed, accurately pointing you to the exact step/instruction that took the longest time.
File system operation processing duration	Indicates the time taken by the WEM agent to process file system operations for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *File system operation processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in file system operation processing is what caused WEM processing to slow down. If the comparative analysis reveals that the processing of file system operations is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to file system operations that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to the processing of file system operations is revealed, accurately pointing you to the exact step/instruction that took the longest time.
Ini file operations processing duration	Indicates the time taken by the WEM agent to process ini file operations for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *Ini file operations processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring

Measurement	Description	Measurement Unit	Interpretation
			console to figure out if a delay in ini file operation processing is what caused WEM processing to slow down. If the comparative analysis reveals that the processing of ini file operations is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to ini file operation processing that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to ini file operation processing is revealed, accurately pointing you to the exact step/instruction that took the longest time.
User DNS processing duration	Indicates the time taken by the WEM agent to perform DNS processing for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *User DNS processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in DNS processing is what caused WEM processing to slow down. If the comparative analysis reveals that DNS processingis indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to DNS processing that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to DNS processing is revealed, accurately pointing you to the exact step/instruction that took the longest time.
File association processing duration	Indicates the time taken by the WEM agent to	Seconds	If the *WEM agent processing* measure

Measurement	Description	Measurement Unit	Interpretation
	process file associations for this user.		reports an abnormally high value, then compare the value of the *File association processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in file association processing is what caused WEM processing to slow down. If the comparative analysis reveals that the processing of file associations is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to file association processing that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to file association processing is revealed, accurately pointing you to the exact step/instruction that took the longest time.
External task processing duration	Indicates the time taken by the WEM agent to process external tasks for this user.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *External task processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in external task processing is what caused WEM processing to slow down. If the comparative analysis reveals that the processing of external tasks is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to external task processing

Measurement	Description	Measurement Unit	Interpretation
			that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to external task processing is revealed, accurately pointing you to the exact step/instruction that took the longest time.
Application processing duration	Indicates the total duration to launch the applications configured in this user's work space environment.	Seconds	If the *WEM agent processing* measure reports an abnormally high value, then compare the value of the *Application processing duration* measure with all other time values reported by the *WEM Agent Processing Breakup* section in the **Layers** tab page of the eG monitoring console to figure out if a delay in launching of applications is what caused WEM processing to slow down. If the comparative analysis reveals that application launch processing is indeed what is slowing down the WEM agent, then use the detailed diagnosis of this measure to view the messages pertaining to application launch processing that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to application launch processing is revealed, accurately pointing you to the exact step/instruction that took the longest time.

Use the detailed diagnosis of the *External task processing duration* measure to view the messages related to the processing of external tasks that are logged in the Citrix WEM Agent.log. The time taken to process each instruction pertaining to external task processing is revealed, accurately pointing you to the exact step/instruction that took the longest time.

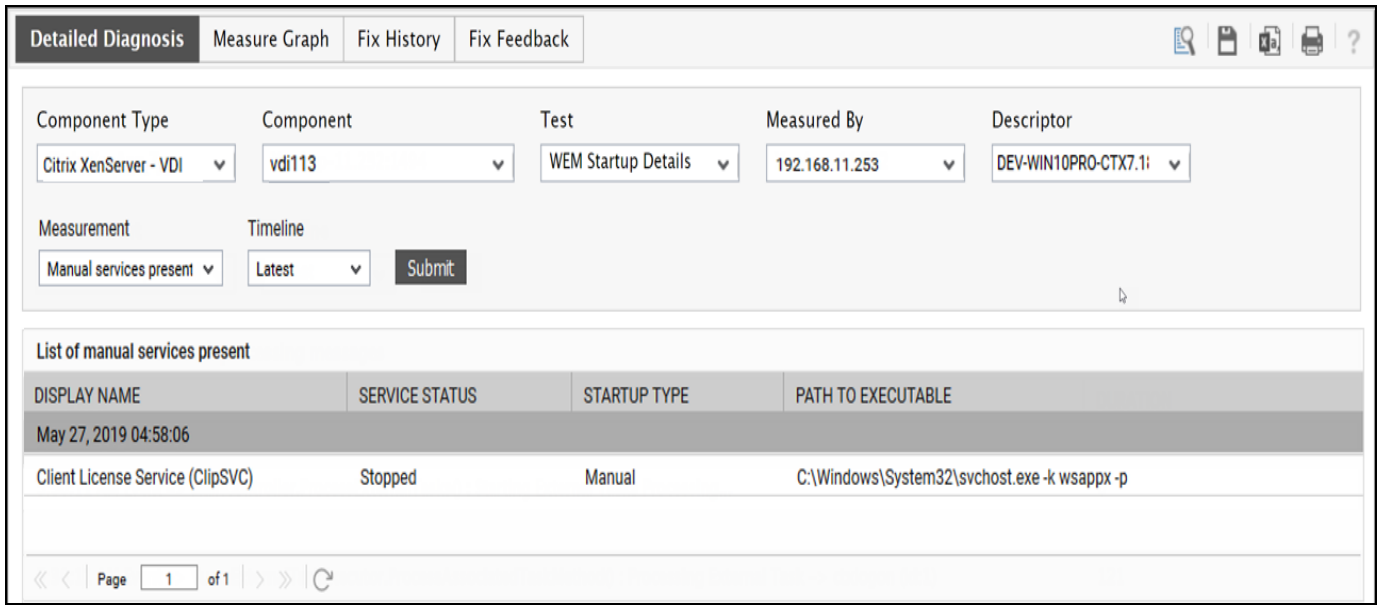


Figure 3.35: The detailed diagnosis of the External task processing duration measure

3.8 Troubleshooting the eG Citrix Monitor

As mentioned already, the eG agent monitoring Citrix XenApp servers of version 6.0/6.5 uses powershell scripts to run tests and pull out metrics from these servers. If the XenApp tests fail, then, first check whether the Powershell SDK pre-exists on the eG agent host. If not, download the SDK from **http://community.citrix.com/display/xa/XenApp+6+PowerShell+SDK**, and install it on the monitored XenApp server. Once the SDK is installed, the eG agent should be able to execute the powershell scripts on the monitored Citrix XenApp servers (v6.0/6.5) without any additional configuration. However, if the tests continue to fail, then check whether any Active Directory Group Policy has been configured to prevent the execution of powershell scripts. If so, you can do either of the following:

a. Change the Group Policy definition to allow the execution of the powershell scripts, (OR)

b. Reconfigure the target XenApp server alone to allow script execution

Each of these steps have been detailed below:

3.8.1 Changing Group Policy Definition

To modify the Active Directory Group Policy to allow script execution, do the following:

1. Login to the Active Directory server.
2. On Windows 2008, follow the Start -> Programs -> Administrative Tools -> Group Policy Management menu sequence.
3. From the tree-structure in the left panel of the window that appears, select the node that represents the group policy of interest to you.
4. Right-click on the group policy and select the **Edit** option.
5. The window depicted by Figure 3.36 will then appear. In the left panel of this window, expand the node representing the policy you have chosen to modify, and then follow the node sequence, **Computer Configuration -> Administrative Templates -> Windows Components -> Windows Powershell** (as indicated by Figure 3.36).

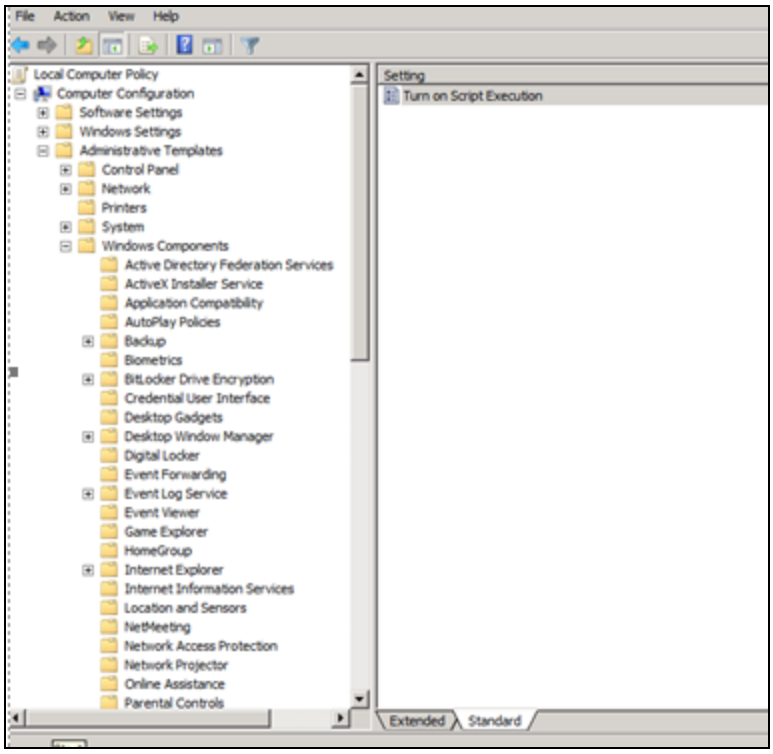


Figure 3.36: Editing the group policy

6. The right panel will change to display a **Turn on Script Execution** setting (see Figure 3.36). Right-click on that setting and select **Edit**. This will invoke Figure 3.37.
7. Select the **Enabled** option from Figure 3.37 to turn on script execution, and then click the **Apply** and **OK** buttons to save the changes.

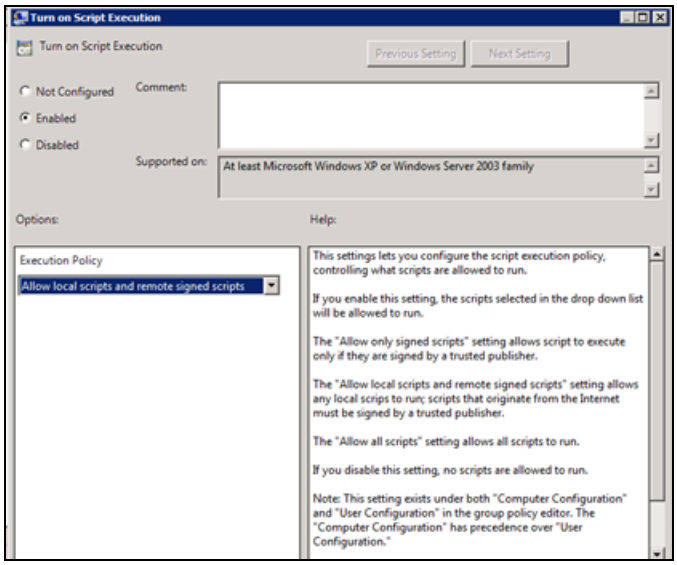


Figure 3.37: Turning on script execution

3.8.2 Reconfiguring the monitored Citrix XenApp server

Typically, if the powershell script execution policy has been set to **Restricted** for a XenApp server, then, upon installing an eG agent on that server, the execution policy will automatically change to **RemoteSigned**. This will enable the eG agent to execute powershell scripts on that server and report metrics.

Note:

If the execution policy for a XenApp server has already been set to **Unrestricted** or **RemoteSigned**, the eG agent setup process will not alter that setting.

However, if you later define an AD group policy setting that restricts script execution, then the group policy setting will over-ride the server-specific setting. In such cases, the XenApp tests will fail. If you do not want to change the Group Policy definition to allow script execution, then, you can set the script execution policy of the target XenApp server alone to **RemoteSigned**, so that the eG agent on that server can execute powershell scripts on the server. For this, do the following:

a. Login to the agent host.

b. First, check the execution policy of the XenApp server. For this, from the PowerShell command prompt, switch to the root directory, and issue the following command:

get-ExecutionPolicy

c. If the output of this command is **RemoteSigned**, it indicates that the eG agent has the privileges

required for script execution. On the other hand, if the output of this command is **Restricted**, you may have to change the policy to **RemoteSigned** to enable the eG agent to execute the scripts. For this, from the PowerShell command prompt, switch to the root directory, and issue the following command:

set-ExecutionPolicy remotesigned

3.9 The Citrix XenApp Dashboard

In order to ascertain how well an application is/has been performing, analysis of the performance of the **System** and **Network** layers of that application alone might not suffice. A closer look at the health of the **Application Layers** is also necessary, so as to promptly detect instantaneous operational issues with the target application, and also proactively identify persistent problems or a consistent performance degradation experienced by the application. To provide administrators with such in-depth insights into overall application performance and to enable them to accurately isolate the root-cause of any application-level slowdown, eG Enterprise offers the **Application Dashboard**. Each of the critical applications monitored by eG Enterprise is accompanied by an exclusive application dashboard. The contents of the dashboard will therefore primarily vary depending upon the application being monitored. Figure 3.38 for instance depicts the Application Dashboard of a **Citrix XenApp server**.

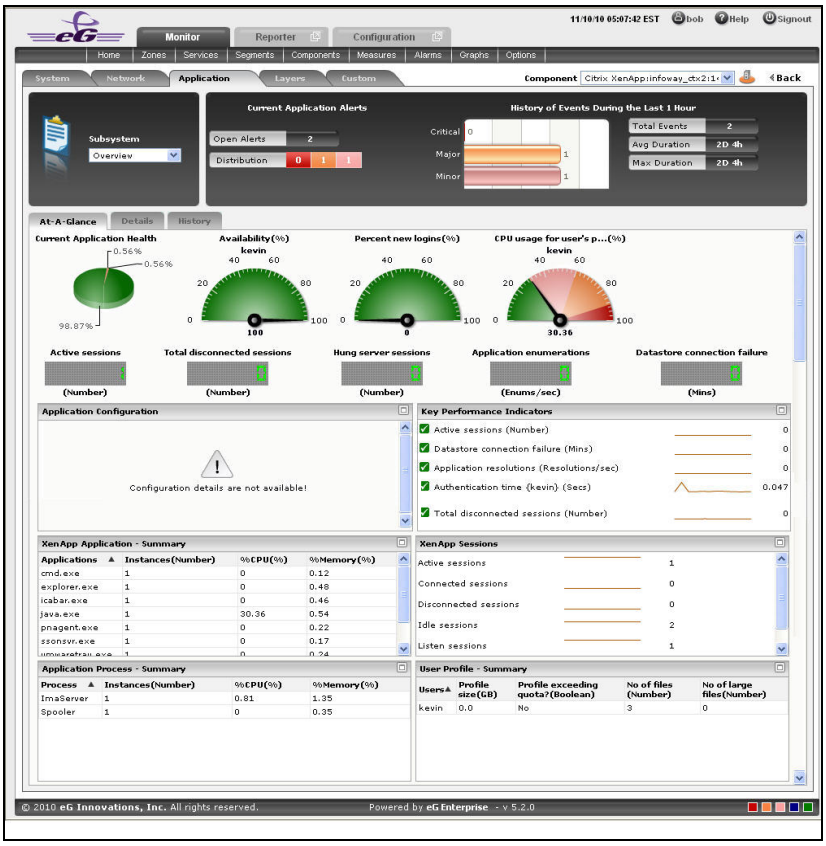


Figure 3.38: The Application Dashboard of a Citrix XenApp application

The contents of the Application dashboard are governed by the Subsystem chosen from Figure 3.38, just like that of the System and Network dashboards. By default, the **Overview** option is chosen from the **Subsystem** list. If need be, this default setting can be changed by picking a different option from the **Subsystem** list. The sections that follow will discuss each of the **Subsystems** offered by the **Citrix XenApp application dashboard** shown in Figure 3.38 above.

3.9.1 Overview

The **Overview** dashboard of a Citrix XenApp application provides an all-round view of the health of the Citrix XenApp application that is being monitored, and helps the administrators to pinpoint the problematic areas. Hence using this dashboard, you can determine the following queries in a quick and easy way.

- Has the application encountered any issue currently? If so, what is the issue and how critical is it?
- How problem-prone has the application been during the last 24 hours? Which application layer has been badly hit?

- Has the administrative staff been able to resolve all past issues? On an average, how long do the administrative personnel take to resolve an issue?
- Are all the key performance parameters of the application operating normally?
- Is the Citrix XenApp application utilizing CPU optimally or is the current CPU usage very high? Did the CPU usage increase suddenly or gradually - i.e., over a period of time?
- How many active sessions are available? What are those sessions?
- Are there any disconnected sessions? If so, when was it disconnected? What was the problem behind the disconnected session?
- How many application processes have been running? What is the CPU utilization of each of those applications? Is there any abnormal increase in CPU utilization over a period of time?
- How many users are active in the current time period? How many files are available for that particular user?

The contents of the **Overview Dashboard** have been elaborated on hereunder:

1. The **Current Application Alerts** section of Figure 3.38 reveals the number and type of issues currently affecting the performance of the Citrix XenApp application that is being monitored. To know more about the issues at hand, click on any cell against **Distribution** that represents the problem priority of interest to you; the details of the current problems of that priority will then appear as depicted by Figure 3.39.

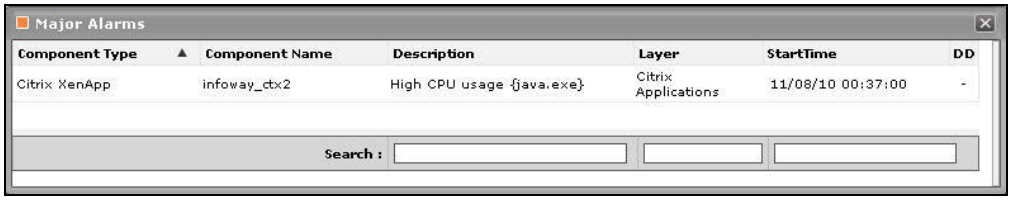


Figure 3.39: Viewing the current application alerts of a particular priority

2. If the pop-up window of Figure 3.39 reveals too many problems, you can use the **Search** text boxes that have been provided at the end of the **Description**, **Layer**, and **StartTime** columns to run quick searches on the contents of these columns, so that the alarm of your interest can be easily located. For instance, to find the alarm with a specific description, you can provide the whole/part of the alarm description in the text box at the end of the **Description** column; this will result in the automatic display of all the alarms with descriptions that contain the specified search string.
3. To zoom into the exact layer, test, and measure that reported any of the listed problems, click

on any of the alarms in the **Alarms** window of Figure 3.39. Doing so will introduce an **Alarm Details** section into the **Alarms** window (see Figure 3.40), which provides the complete information related to the problem clicked on These details include the **Site** affected by the problem for which the alarm was raised, the test that reported the problem, and the percent usage indicating the **Last Measure**.

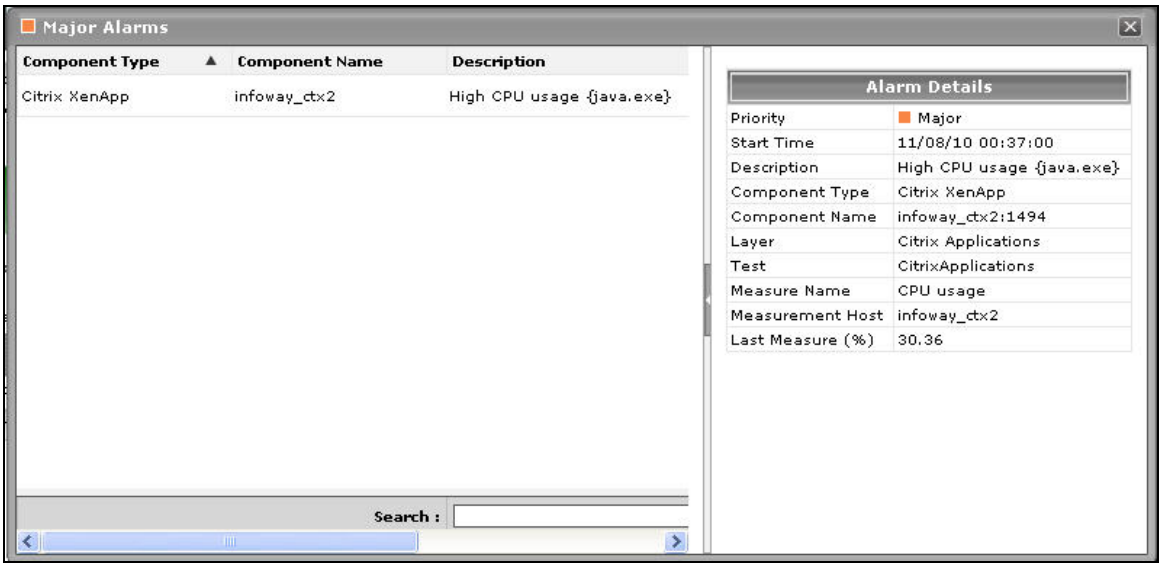


Figure 3.40: Additional alarm details

4. While the list of current issues faced by the application serves as a good indicator of the current state of the application, to know how healthy/otherwise the application has been over the time, a look at the problem history of the application is essential. Therefore, the dashboard provides the **History of Events** section; this section presents a bar chart, where every bar indicates the total number of problems along with their corresponding severity, which was experienced by the Citrix XenApp application during the last 1 hour (by default). Clicking on a bar here will lead you to Figure 3.41, which provides a detailed history of problems of that priority. Alongside the bar chart, you will also find a table displaying the average and maximum duration for problem resolution; this table helps you determine the efficiency of your administrative staff.

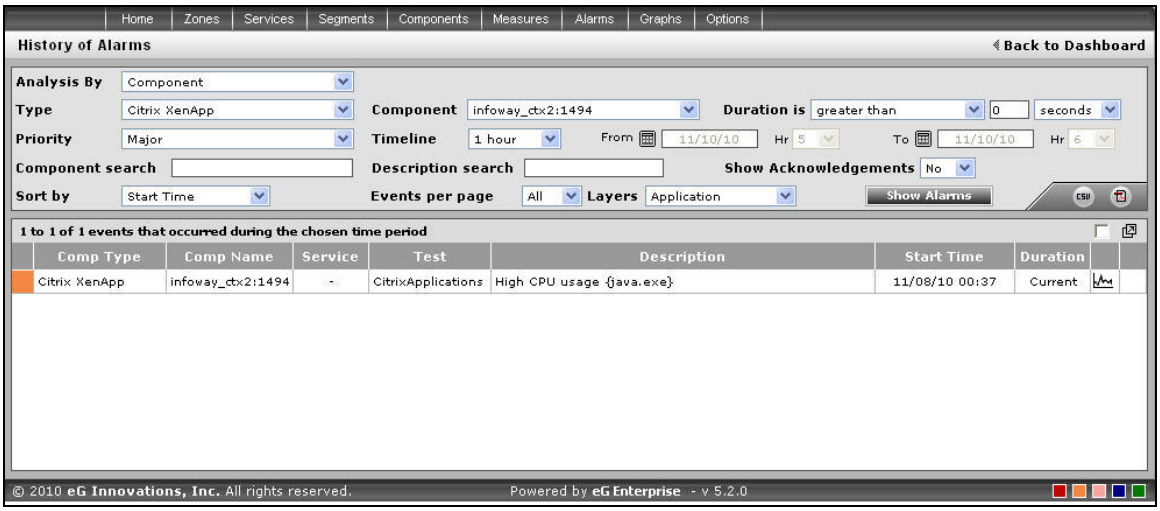


Figure 3.41: The problem history of the target application

If required, you can override the default time period of 1 hour of the event history, by following the steps below:

- Click the button at the top of the dashboard to invoke the **Dashboard Settings** window.
- Select the **Event History** option from the **Default timeline for** list.
- Set a different default timeline by selecting an option from the **Timeline** list.
- Finally, click the **Update** button.

5. In the dashboard, you will find that the **History of Events** section is followed by an **At-A-Glance** section. This section reveals the current status of some critical metrics and key components of the Citrix XenApp application at a single glance, using pie charts, digital displays and gauge charts. For instance, the **Current Application Health** pie chart indicates the current health of the application by representing the number of application-related metrics that are in various states. Clicking on a slice here will take you to Figure 3.41 that provides a detailed problem history.
6. The dial and digital graphs that follow will provide you with quick updates on the status of a pre-configured set of resource usage-related metrics pertaining to the Citrix XenApp application. If required, you can configure the dial graphs to display the threshold values of the corresponding measures along with their actual values, so that deviations can be easily detected. For this purpose, do the following:

    - Click the button at the top of the dashboard to invoke the **Dashboard Settings** window.

- Set the **Show Thresholds** flag in the window to **Yes**.
- Finally, click the **Update** button.

7. You can customize the **At-A-Glance** tab page further by overriding the default measure list for which dial/digital graphs are being displayed in that tab. To achieve this, do the following:

   - Click on the  icon at the top of the **Application Dashboard**. In the **Dashboard Settings** window that appears, select **Application** from the **Module** list, and **Overview** from the **Sub-System** list.
   - To add measures for the dial graph, pick the **Dial Graph** option from the **Add/Delete Measures for** list. Upon selection of the **Dial Graph** option, the pre-configured measures for the dial graph will appear in the **Existing Value(s)** list. Similarly, to add a measure to the digital display, pick the **Digital Graph** option from the **Add/Delete Measures for** list. In this case, the **Existing Value(s)** list box will display all those measures for which digital displays pre-exist.
   - Next, select the **Test** that reports the said measure, pick the measure of interest from the **Measures** list, provide a **Display** name for the measure, and click the **Add** button to add the chosen measure to the **Existing Value(s)** list. **Note that while configuring measures for a dial graph the 'Measures' list will display only those measures that report percentage values**.

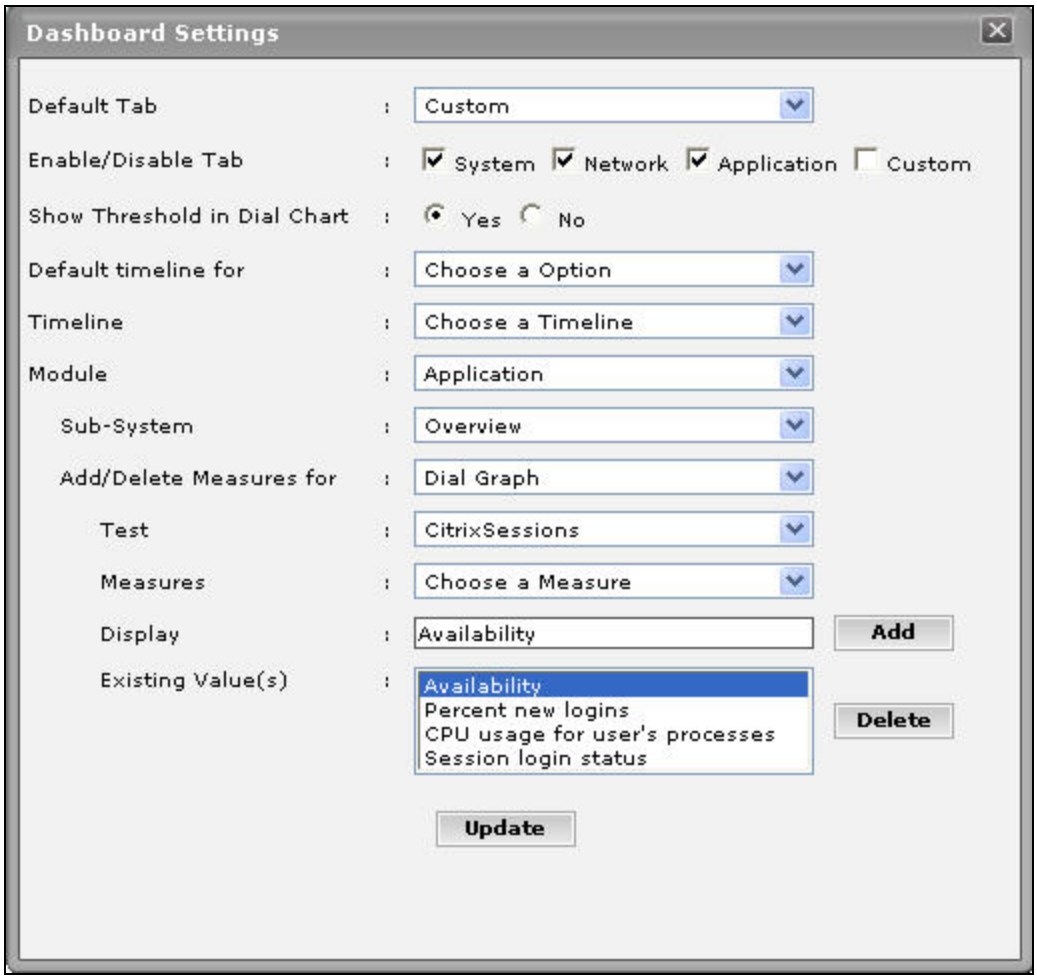


Figure 3.42: Configuring measures for the dial graph

- If you want to delete one/more measures from the dial/digital graphs, then, as soon as you choose the **Dial Graph** or **Digital Graph** option from the **Add/Delete Measures for** list, pick any of the displayed measures from the **Existing Value(s)** list, and click the **Delete** button.
- Finally, click the **Update** button to register the changes.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the **Dashboard Settings** window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

8. Clicking on a dial/digital graph will lead you to the layer model page of the Citrix XenApp

Application; this page will display the exact layer-test combination that reports the measure represented by the dial/digital graph.

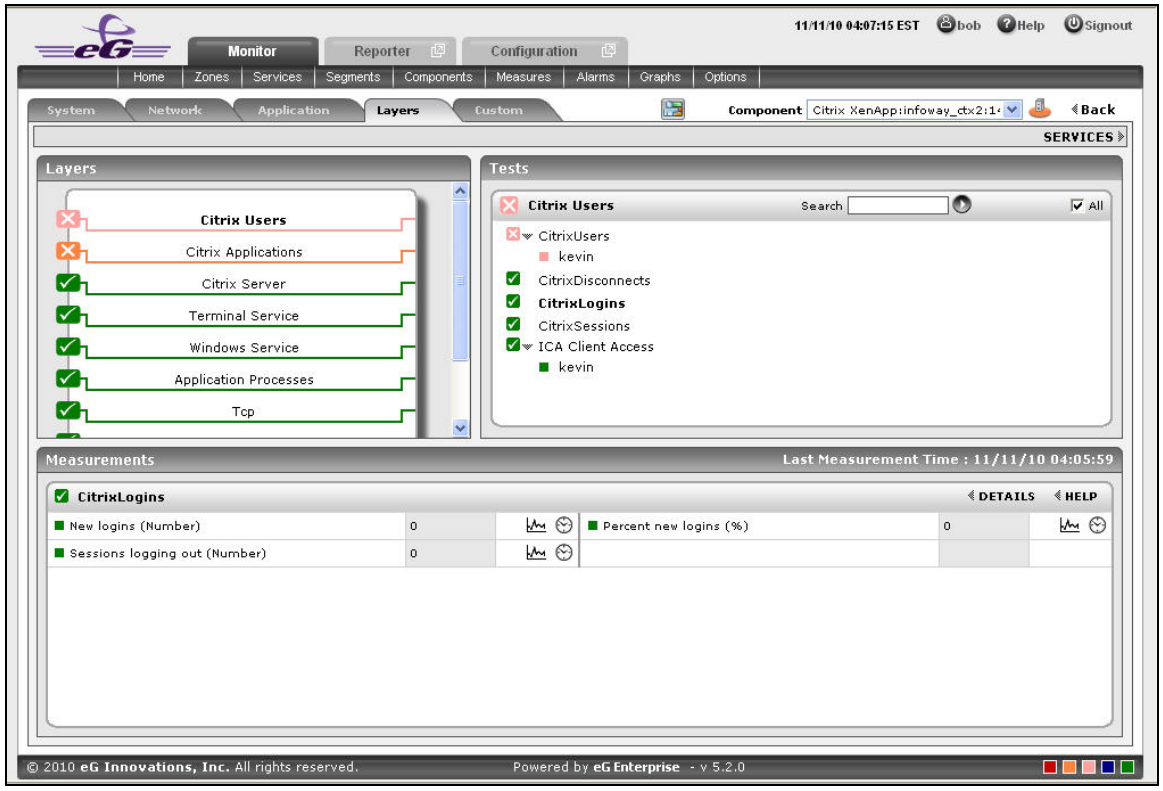


Figure 3.43: The page that appears when the dial/digital graph in the Overview dashboard of the Citrix XenApp Application is clicked

9. If your eG license enables the **Configuration Management** capability, then, an **Application Configuration** section will appear here providing the basic configuration of the application. You can configure the type of configuration data that is to be displayed in this section by following the steps below:

   - Click on the icon at the top of the **Application Dashboard**. In the **Dashboard Settings** window that appears, select **Application** from the **Module** list, and **Overview** from the **Sub-System** list.
   - To add more configuration information to this section, first, pick the **Application Configuration** option from the **Add/Delete Measures for** list. Upon selection of this option, all the configuration measures that pre-exist in the **Configuration Management** section will appear in the **Existing Value(s)** list.

- Next, select the config **Test** that reports the said measure, pick the measure of interest from the **Measures** list, provide a **Display** name for the measure, and click the **Add** button to add the chosen measure to the **Existing Value(s)** list.
- If you want to delete one/more measures from this section, then, as soon as you choose the **Application Configuration** option from the **Add/Delete Measures for** list, pick any of the displayed measures from the **Existing Value(s)** list, and click the **Delete** button.
- Finally, click the **Update** button to register the changes.

10. Next to this section, you will find a pre-configured list of **Key Performance Indicators** of the Citrix XenApp application. Besides indicating the current state of and current value reported by a default collection of critical metrics, this section also reveals 'miniature' graphs of each metric, so that you can instantly study how that measure has behaved during the last 1 hour (by default) and thus determine whether the change in state of the measure was triggered by a sudden dip in performance or a consistent one. Clicking on a measure here will lead you to Figure 3.44, which displays the layer and test that reports the measure.

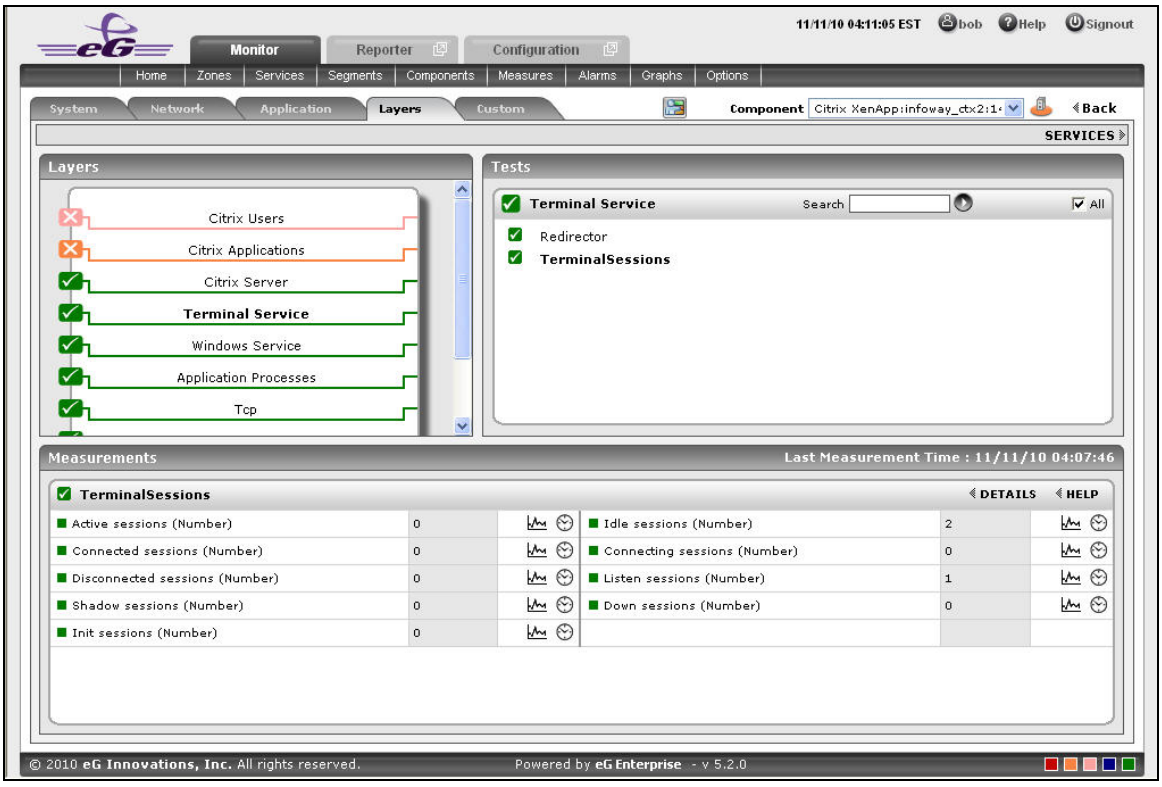


Figure 3.44: Clicking on a Key Performance Indicator

11. You can, if required, override the default measure list in the **Key Performance Indicators**

section by adding more critical measures to the list or by removing one/more existing ones from the list. For this, do the following:

- Click on the icon at the top of the **Application Dashboard**. In the **Dashboard Settings** window that appears, select **Application** from the **Module** list, and **Overview** from the **Sub-System** list.
- To add more metrics to the **Key Performance Indicators** section, first, pick the **Performance Indicator** option from the **Add/Delete Measures for** list. Upon selection of this option, all the measures that pre-exist in the **Key Performance Indicators** section will appear in the **Existing Value(s)** list.
- Next, select the **Test** that reports the said measure, pick the measure of interest from the **Measures** list, provide a **Display** name for the measure, and click the **Add** button to add the chosen measure to the **Existing Value(s)** list.
- If you want to delete one/more measures from this section, then, as soon as you choose the **Key Performance Indicators** option from the **Add/Delete Measures for** list, pick any of the displayed measures from the **Existing Value(s)** list, and click the **Delete** button.
- Finally, click the **Update** button to register the changes.

12. Clicking on a 'miniature' graph that corresponds to a key performance indicator will enlarge the graph, so that you can view and analyze the measure behaviour more clearly, and can also alter the **Timeline** and dimension (**3D**/ **2D**) of the graph, if need be.

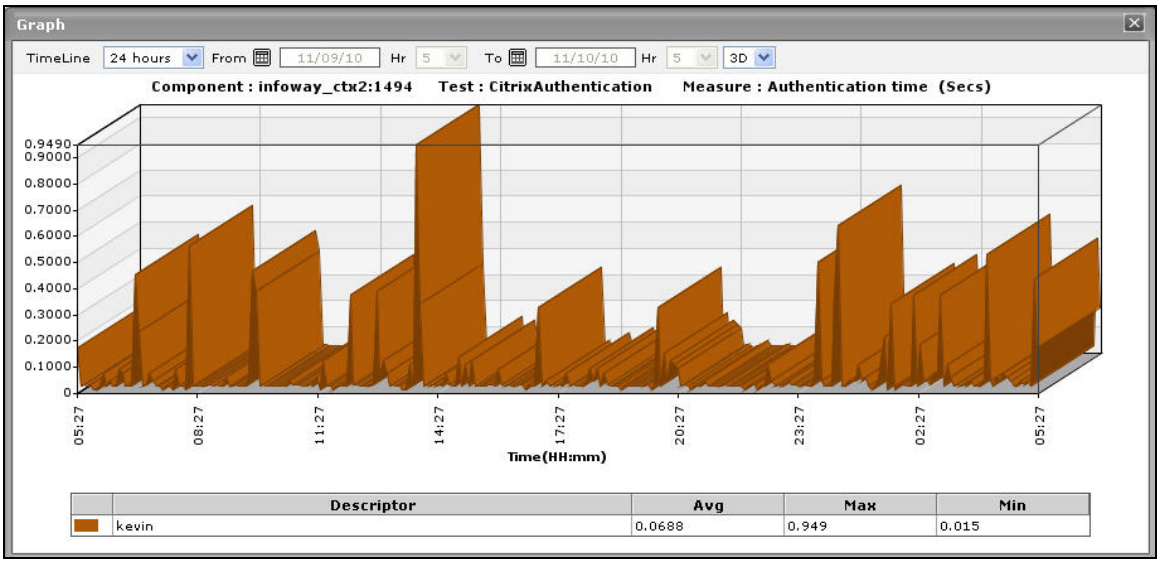


Figure 3.45: Enlarging the Key Performance Indicator graph

13. This way, the first few sections of the **At-A-Glance** tab page helps you to understand the issues that are currently affecting the application health, and when they actually originated. However, to diagnose the root-cause of these issues, you would have to take help from the remaining sections of the **At-A-Glance** tab page. For instance, the **Key Performance Indicators** section may reveal a slowdown in the Citrix server. But, to determine whether this slowdown is owing to too many instances of an application executing on the server, or due to excessive resource usage by one/more applications/OS-level processes on the server, you need to focus on the **XenApp Application - Summary** section and the **Application Process - Summary** section in the dashboard. The **XenApp Application - Summary** section lists the applications that are currently executing on the XenApp server, and for each application, reveals:

    - The percent CPU utilization of that application;
    - The percentage of memory that is utilized by that application;
    - The number of instances of that application that are currently operational

14. This section turns your attention to the most resource-hungry applications on the Citrix XenApp server.
15. The **XenApp Sessions** section provides you with a quick overview of the current session activity on the Citrix XenApp server. Session overloads, idle sessions that are unnecessarily consuming resources, and hung server sessions causing slowdowns can be instantly detected using this section. Each measure displayed here is associated with a miniature graph. By clicking on the graph, you can view an enlarged graph of that particular session-related measure for a default period of 24 hours, and infer whether any abnormal activity has taken place during the default timeline. This default timeline can be altered according to the user's desire.

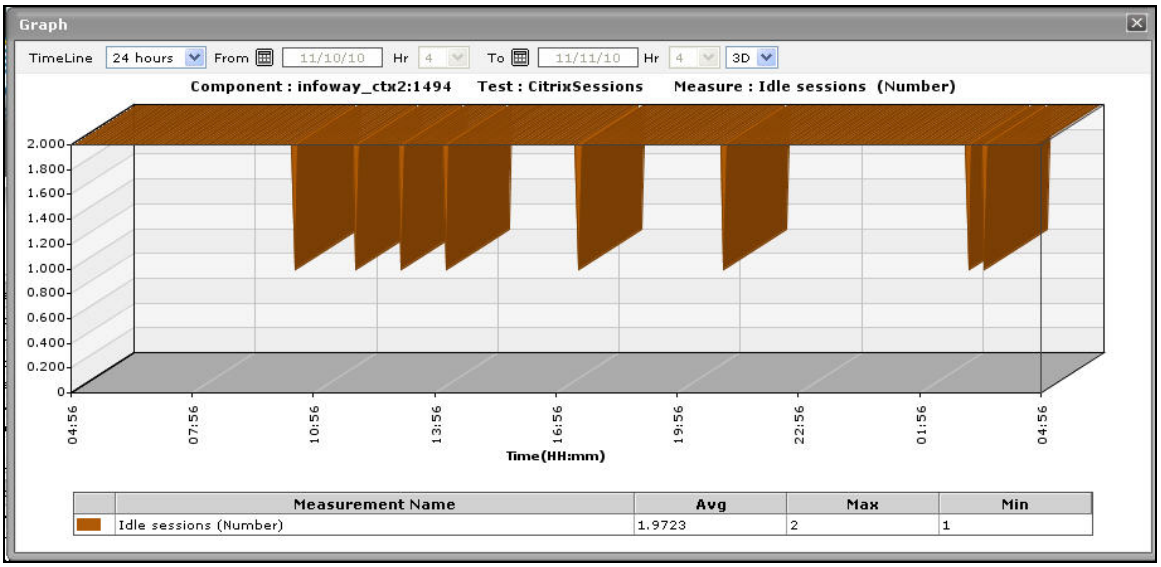


Figure 3.46: Idle sessions graph that is enlarged from the XenApp Sessions.

16. The **Application Process - Summary** section, on the other hand, traces the percent CPU usage and percent memory usage of each of the Citrix XenApp processes that are currently executing on the target host, and thus leads you to the resource-intensive processes. By default, the process list provided by this section is sorted in the alphabetical order of the process names. If need be, you can change the sort order so that the processes are arranged in, say, the descending order of values displayed in the **Instances** column - this column displays the number of instances of each process that is in execution currently. To achieve this, simply click on the column heading - **Instances**. Doing so tags the **Instances** label with a **down arrow** icon - this icon indicates that the process list is currently sorted in the descending order of the instance count. To change the sort order to 'ascending', all you need to do is just click again on the **Instances** label or the **down arrow** icon. Similarly, you can sort the process list based on any column available in the **Application Process - Summary** section.

17. While the **At-A-Glance** tab page reveals the current state of the Citrix XenApp application and the overall resource usage of the application, to perform additional diagnosis on problem conditions highlighted by the **At-A-Glance** tab page and to accurately pinpoint their root-cause, you need to switch to the **Details** tab page by clicking on it. For instance, the **At-A-Glance** tab page may that the CPU usage of an application is very high, but to know which user is utilizing that application, you will have to use the **Details** tab page.

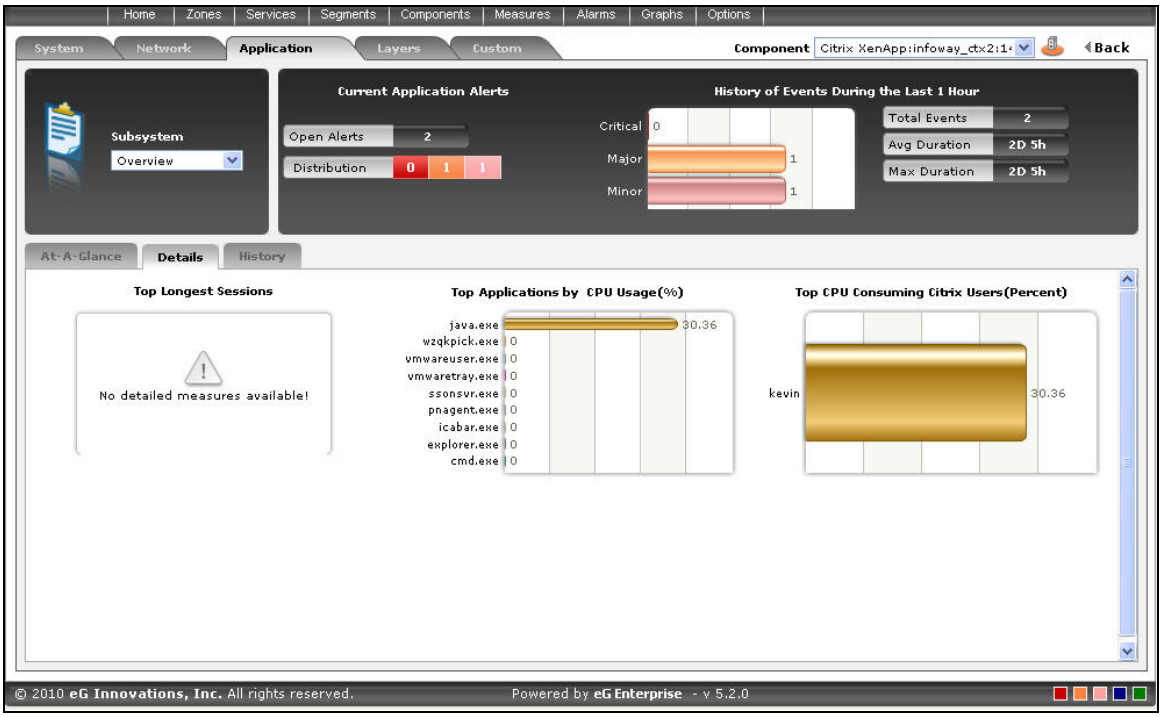


Figure 3.47: The Details tab page of the Application Overview Dashboard

18. The **Details** tab page comprises of a default set of comparison bar graphs using which you can accurately determine the following:

    - What are the longest sessions on the Citrix server?
    - What are the resource-intensive applications on the Citrix server?
    - Which user is utilizing the maximum CPU resources on the server?

19. If required, you can configure the **Details** tab page to include comparison graphs for more measures, or can even remove one/more existing graphs by removing the corresponding measures. To achieve this, do the following:

    - Click on the icon at the top of the **Application Dashboard**. In the **Dashboard Settings** window that appears, select **Application** from the **Module** list, and **Overview** from the **Sub-System** list.
    - To add measures for comparison graphs, first, pick the **Comparison Graph** option from the **Add/Delete Measures for** list. Upon selection of this option, the pre-configured measures for comparison graphs will appear in the **Existing Value(s)** list.
    - Next, select the **Test** that reports the said measure, pick the measure of interest from the

Measures list, provide a **Display** name for the measure, and click the **Add** button to add the chosen measure to the **Existing Value(s)** list.

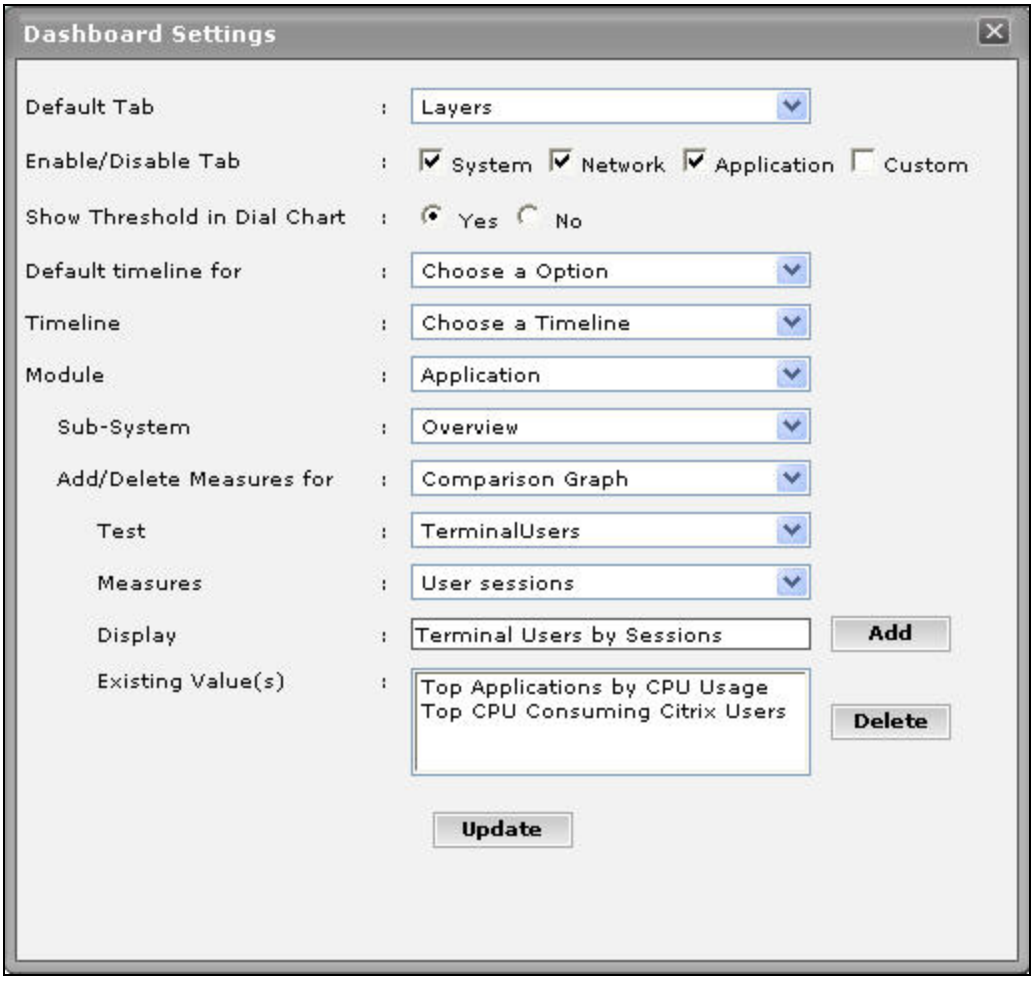


Figure 3.48: Configuring measures for the dial graph

- If you want to delete one/more measures for which comparison graphs pre-exist in the **details** tab page, then, as soon as you choose the **Comparison Graph** option from the **Add/Delete Measures for** list, pick any of the displayed measures from the **Existing Value (s)** list, and click the **Delete** button.
- Finally, click the **Update** button to register the changes.

Note:
Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the **Dashboard Settings** window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

20. By default, the comparison bar graphs list the top-10 applications and users only. To view the complete list of applications and users, simply click on the corresponding graph in Figure 3.47. This enlarges the graph as depicted by Figure 3.49.

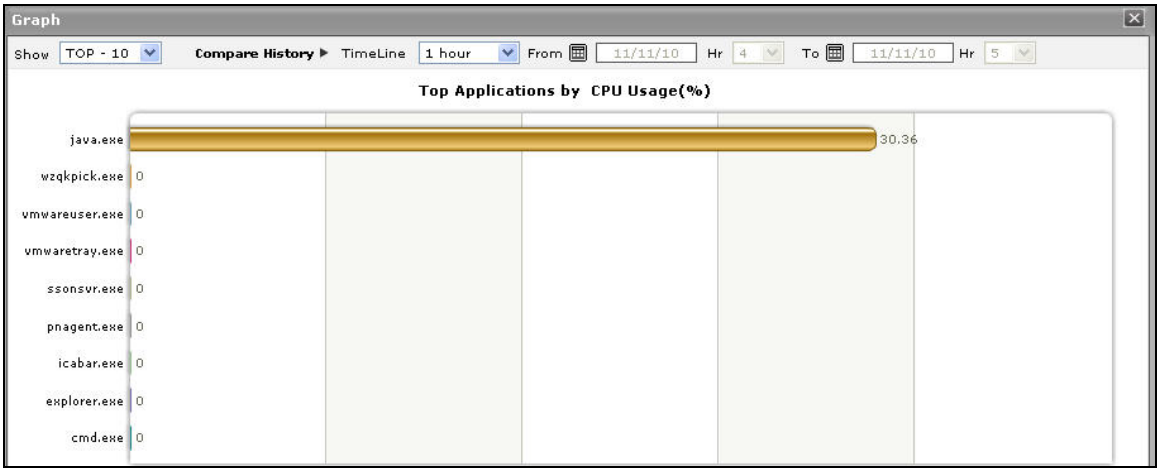


Figure 3.49: The expanded top-n graph in the Details tab page of the Application Overview Dashboard

21. Though the enlarged graph lists all the applications or users (as the case may be) by default, you can customize the enlarged graph to display the details of only a few of the best/worst-performing users and applications by picking a **TOP-N** or **LAST-N** option from the **Show** list in Figure 3.49.

22. Another default aspect of the enlarged graph is that it pertains to the current period only. Sometimes however, you might want to know what occurred during a point of time in the past; for instance, while trying to understand the reason behind a sudden spike in CPU usage on a particular day last week, you might want to first determine which application is guilty of abnormal CPU consumption on the same day. To figure this out, the enlarged graph allows you to compare the historical performance of applications or users. For this purpose, click on the **Compare History** link in Figure 1.12 and select the **TimeLine** of your choice.

23. For detailed time-of-day / trend analysis of the historical performance of a Citrix XenApp application, use the **History** tab page. By default, this tab page (see Figure 3.50) provides time-of-day graphs of critical measures extracted from the target Citrix XenApp application, using which you can understand how performance has varied during the default period of 24 hours. In the event of a problem, these graphs will help you determine whether the problem occurred suddenly or grew with time. To alter the timeline of all the graphs simultaneously, click on the **Timeline** link at the right, top corner of the **History** tab page of Figure 3.50.

24. You can even override the default timeline (of 24 hours) of the measure graphs, by following

the steps below:

- Click on the icon at the top of the **Application Dashboard**.
- In the **Dashboard Settings** window that appears, select **History Graph** from the **Default Timeline for** list.
- Then, choose a **Timeline** for the graph.
- Finally, click the **Update** button.

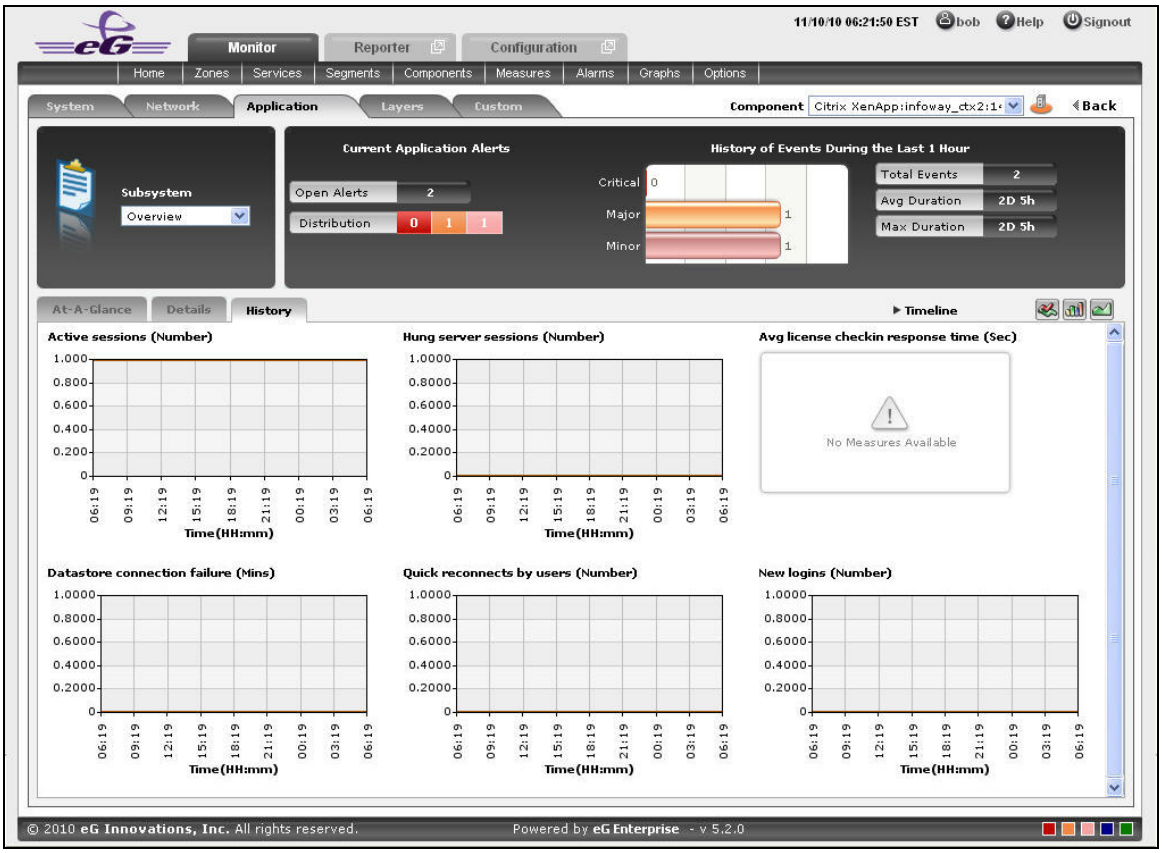


Figure 3.50: Time-of-day measure graphs displayed in the History tab page of the Application Overview Dashboard

25. You can click on any of the graphs to enlarge it, and can change the **Timeline** of that graph in the enlarged mode.

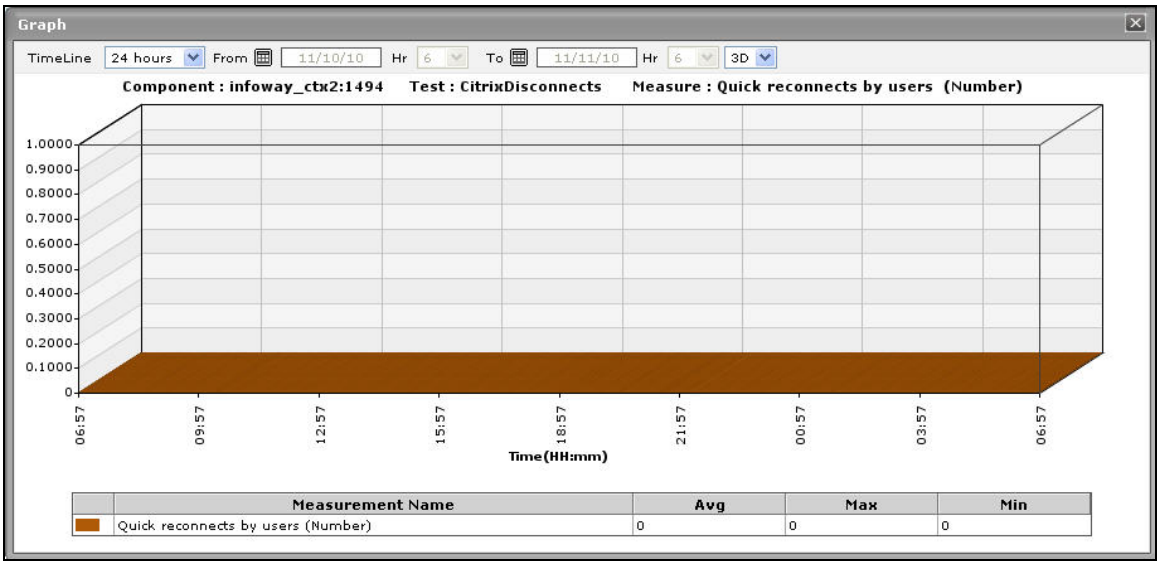


Figure 3.51: An enlarged measure graph of a Citrix XenApp Application

26. In case of tests that support descriptors, the enlarged graph will, by default, plot the values for the **Top-10** descriptors alone. To configure the graph to plot the values of more or less number of descriptors, select a different **TOP-N** / **LAST-N** option from the **Show** list in 3.9.1.

27. If you want to quickly perform service level audits on the Citrix XenApp server, then summary graphs may be more appropriate than the default measure graphs. For instance, a summary graph might come in handy if you want to determine the percentage of time during the last 24 hours the Citrix XenApp server was available. Using such a graph, you can determine whether the availability levels guaranteed by the Citrix XenApp server were met or not, and if not, how frequently did the server falter in this regard. To invoke such summary graphs, click on the icon at the right, top corner of the **History** tab page. Figure 3.52 will then appear.

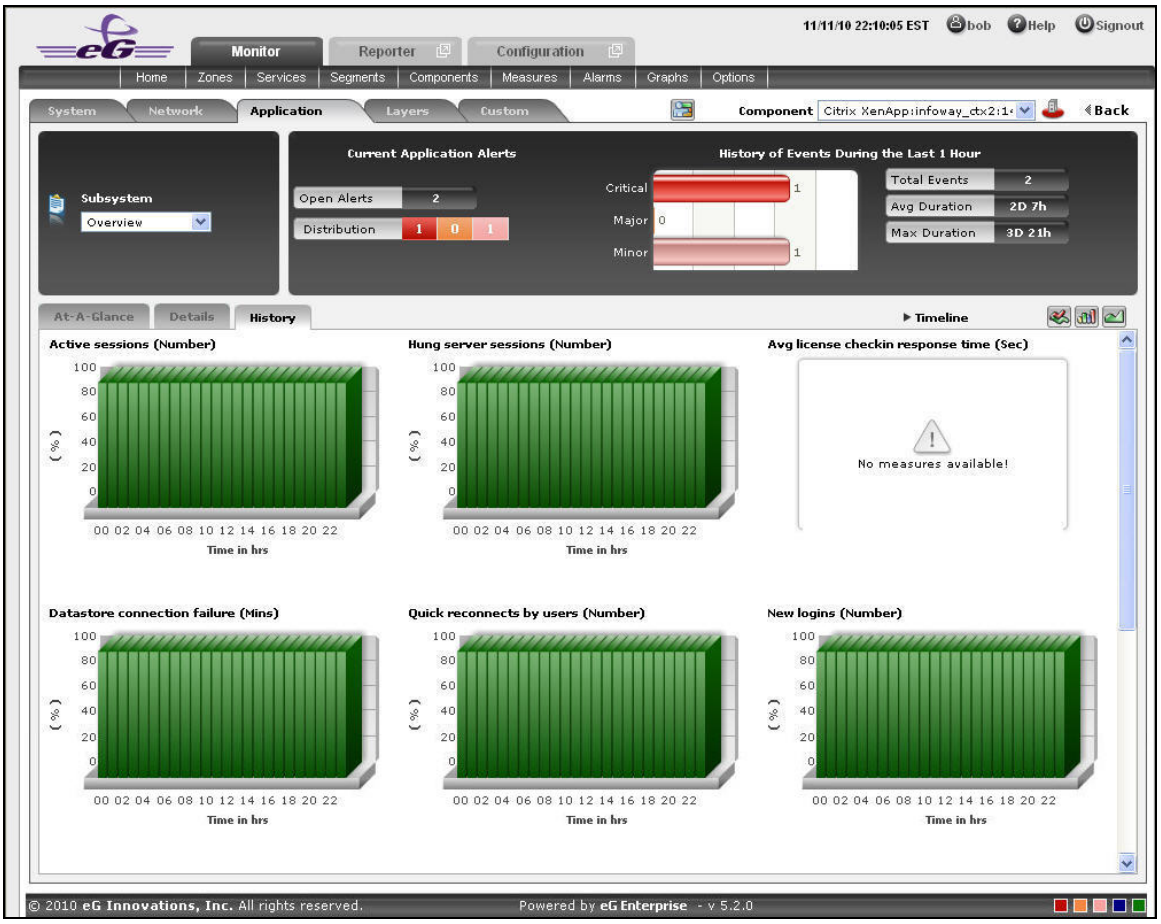


Figure 3.52: Summary graphs displayed in the History tab page of the Application Overview Dashboard

28. You can alter the timeline of all the summary graphs at one shot by clicking the **Timeline** link at the right, top corner of the **History** tab page of Figure 3.52. You can even alter the default timeline (of 24 hours) for these graphs, by following the steps given below:

    - Click on the icon at the top of the **Application Dashboard**.
    - In the **Dashboard Settings** window that appears, select **Summary Graph** from the **Default Timeline for** list.
    - Then, choose a **Timeline** for the graph.
    - Finally, click the **Update** button.

29. To change the timeline of a particular graph, click on it; this will enlarge the graph as depicted by Figure 3.53. In the enlarged mode, you can alter the **Timeline** of the graph. Also, though the graph plots hourly summary values by default, you can pick a different **Duration** for the graph in the enlarged mode, so that daily/monthly performance summaries can be analyzed.

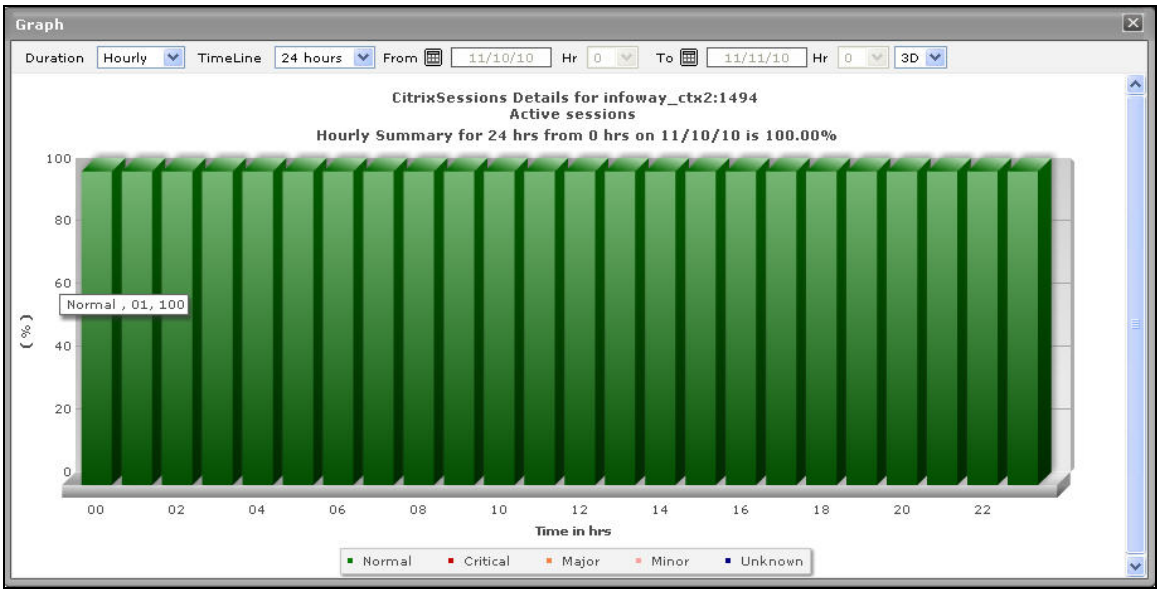


Figure 3.53: An enlarged summary graph of the Citrix XenApp Application

30. To perform effective analysis of the past trends in performance, and to accurately predict future measure behavior, click on the icon at the right, top corner of the **History** tab page. These trend graphs typically show how well and how badly a measure has performed every hour during the last 24 hours (by default). For instance, the Active Sessions trend graph will point you to when (during the last 24 hours) the number of active sessions to the Citrix server had peaked, and when it was very low. If the gap between the minimum and maximum values is marginal, you can conclude that the number of active sessions has been more or less constant during the designated period; this implies that the active session has neither increased nor decreased steeply during the said timeline. On the other hand, a wide gap between the maximum and minimum values is indicative erratic session load on the server, and may necessitate further investigation. By carefully studying the trend graph, you can even determine the points of time at which the session has behaved abnormally during the stated timeline, and this knowledge can greatly aid further diagnosis.

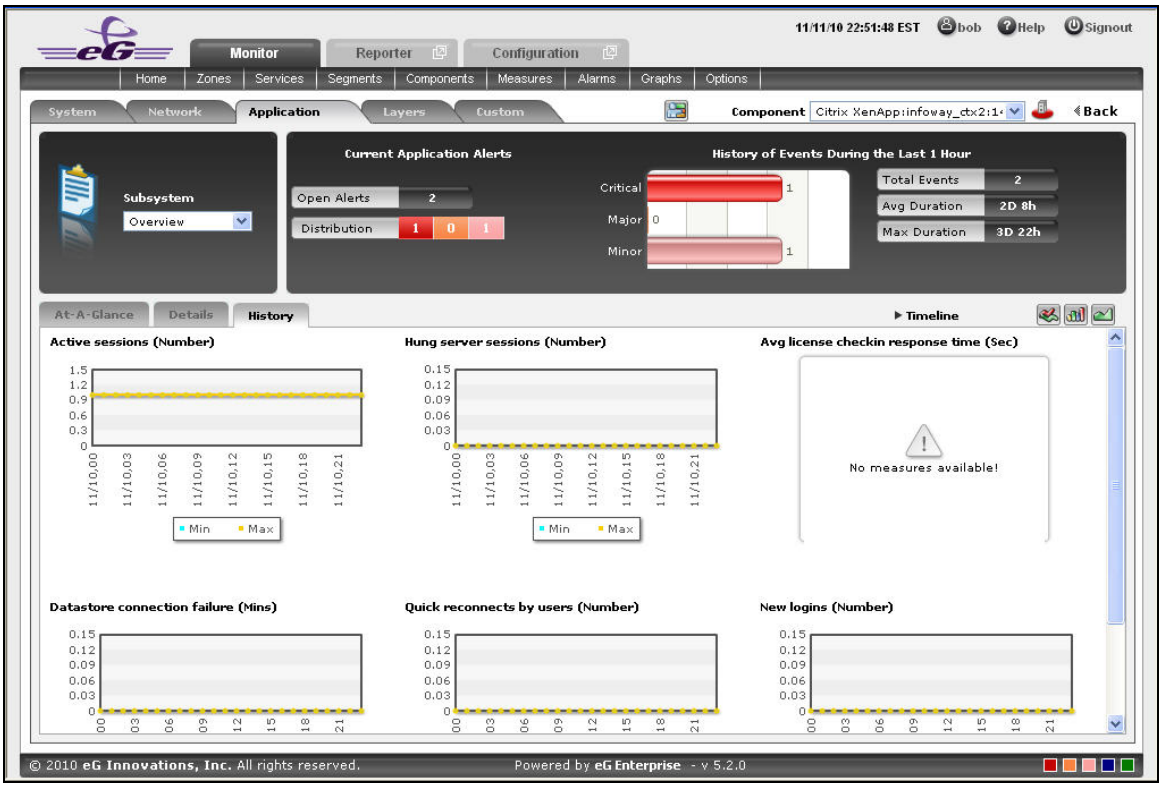


Figure 3.54: Trend graphs displayed in the History tab page of the Application Overview Dashboard

31. To analyze trends over a broader time scale, click on the **Timeline** link at the right, top corner of the **History** tab page, and edit the **Timeline** of the trend graphs. Clicking on any of the miniature graphs in this tab page will enlarge that graph, so that you can view the plotted data more clearly and even change its **Timeline**.

32. To override the default timeline (of 24 hours) of the trend graphs, do the following:

    - Click on the icon at the top of the **Application Dashboard**.
    - In the **Dashboard Settings** window that appears, select **Trend Graph** from the **Default Timeline for** list.
    - Then, choose a **Timeline** for the graph.
    - Finally, click the **Update** button.

33. Besides the timeline, you can even change the **Duration** of the trend graph in the enlarged mode. By default, **Hourly** trends are plotted in the trend graph. By picking a different option from the **Duration** list, you can ensure that **Daily** or **Monthly** trends are plotted in the graph instead.

34. Also, by default, the trend graph only plots the minimum and maximum values registered by a measure. Accordingly, the **Graph** type is set to **Min/Max** in the enlarged mode. If need be, you can change the **Graph** type to **Avg**, so that the average trend values of a measure are plotted for the given **Timeline**. For instance, if an average trend graph is plotted for the *Active Sessions* measure, then the resulting graph will enable administrators to ascertain how many sessions, on an average, were active on the Citrix server during a specified timeline; such a graph can help you assess how session load has changed during a given timeline.

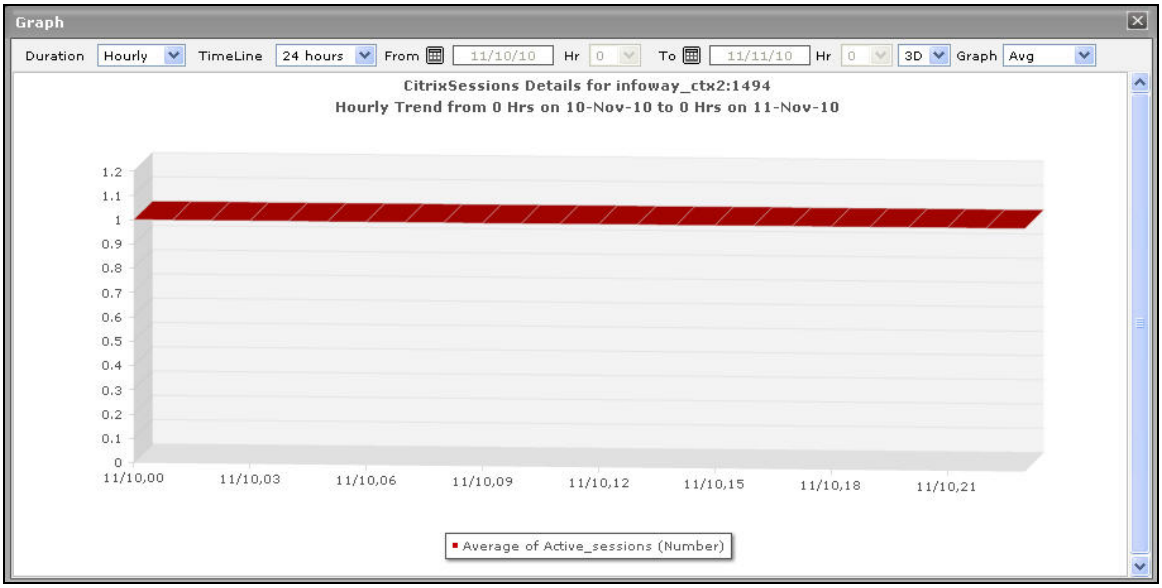


Figure 3.55: Viewing a trend graph that plots average values of a measure for a Citrix XenApp application

35. Likewise, you can also choose **Sum** as the **Graph** type to view a trend graph that plots the sum of the values of a chosen measure for a specified timeline. For instance, if you plot a 'sum of trends' graph for the measure that reports the number of active sessions of the application, then, the resulting graph will enable you to analyze, on an hourly/daily/monthly basis (depending upon the **Duration** chosen), how the level of session activity on the Citrix server has varied.

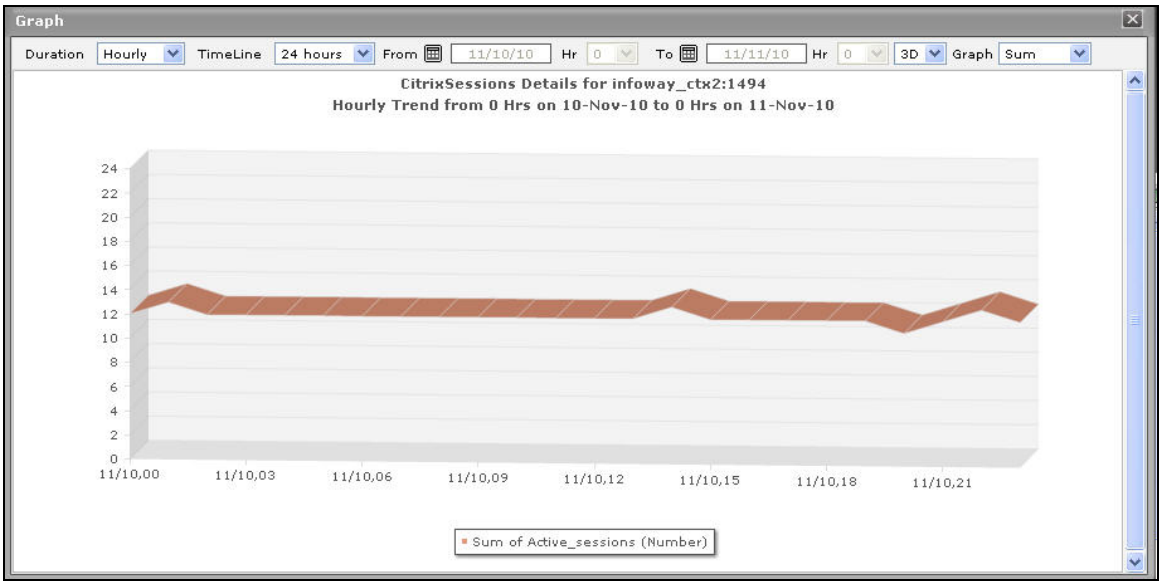


Figure 3.56: A trend graph plotting sum of trends for a Citrix XenApp application

Note:

In case of descriptor-based tests, the Summary and Trend graphs displayed in the History tab page typically plot the values for a single descriptor alone. To view the graph for another descriptor, pick a descriptor from the drop-down list made available above the corresponding summary/trend graph.

36. At any point in time, you can switch to the measure graphs by clicking on the button.

37. Typically, the **History** tab page displays measure, summary, and trend graphs for a default set of measures. If you want to add graphs for more measures to this tab page or remove one/more measures for which graphs pre-exist in this tab page, then, do the following:

    - Click the button at the top of the dashboard.
        - The **Dashboard Settings** window then appears. From the **Module** list of Figure 3.57, pick **Application**, choose **Overview** as the **Sub-System**, and then, select **History Graph** from the **Add/Delete Measures for** list.

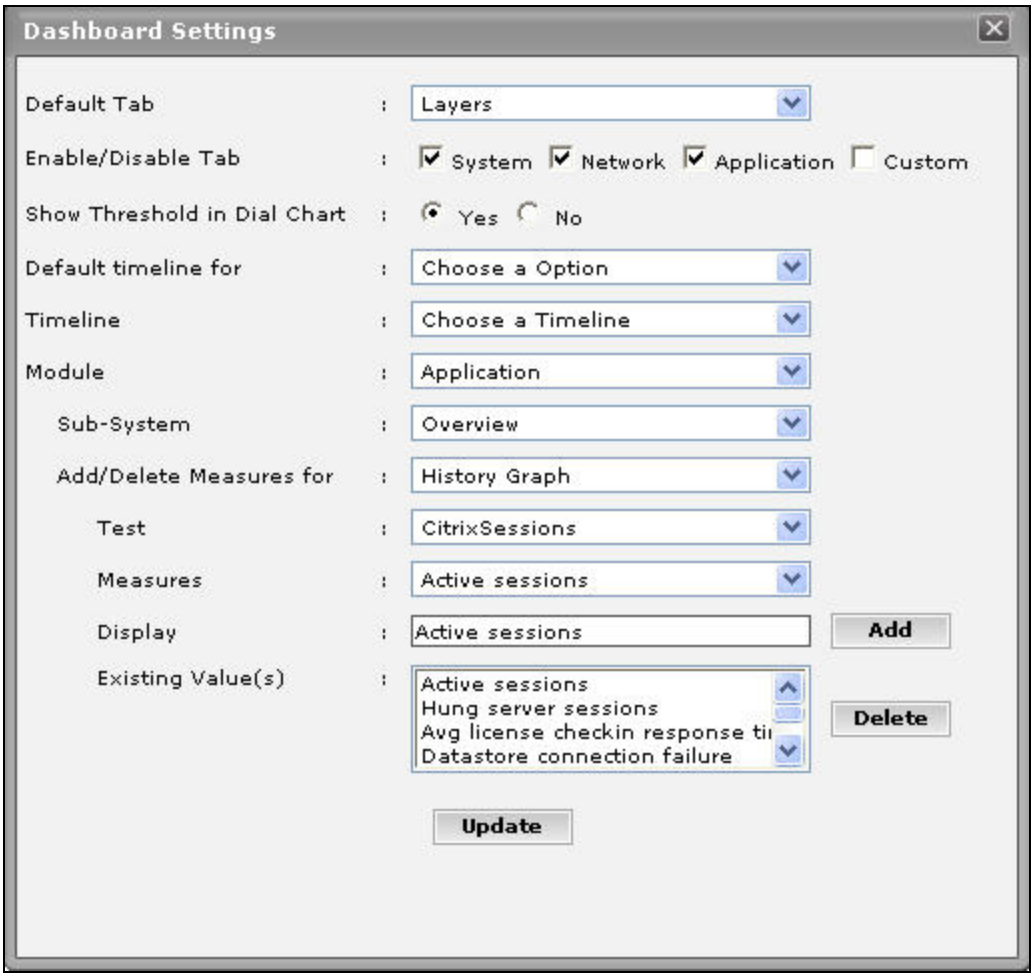


Figure 3.57: Adding a new graph to the **History** tab page

- The measures for which graphs pre-exist in the **History** tab page will be automatically displayed in the **Existing Value(s)** list. To delete a measure, and in effect, its corresponding graph as well, select the measure from the **Existing Value(s)** list, click the **Delete** button, and then click the **Update** button.
- To add a new graph, first, pick the **Test** that reports the measure for which a graph is to be generated.
- Next, select the **Measure** of interest.
- Provide a **Display** name for the measure. Then, click the **Add** button to add the measure to the **Existing Values(s)** list. Finally, click the **Update** button.
- This will add a new measure, summary, and trend graph for the chosen measure, to the **History** tab page.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the Dashboard Settings window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

3.9.1.1 CitrixServer

To periodically assess the availability of a Citrix server, quickly measure the load-handling capacity of the server, and promptly detect aberrations in the internal operations of the server, select the **CitrixServer** option from the **Subsystem** list.

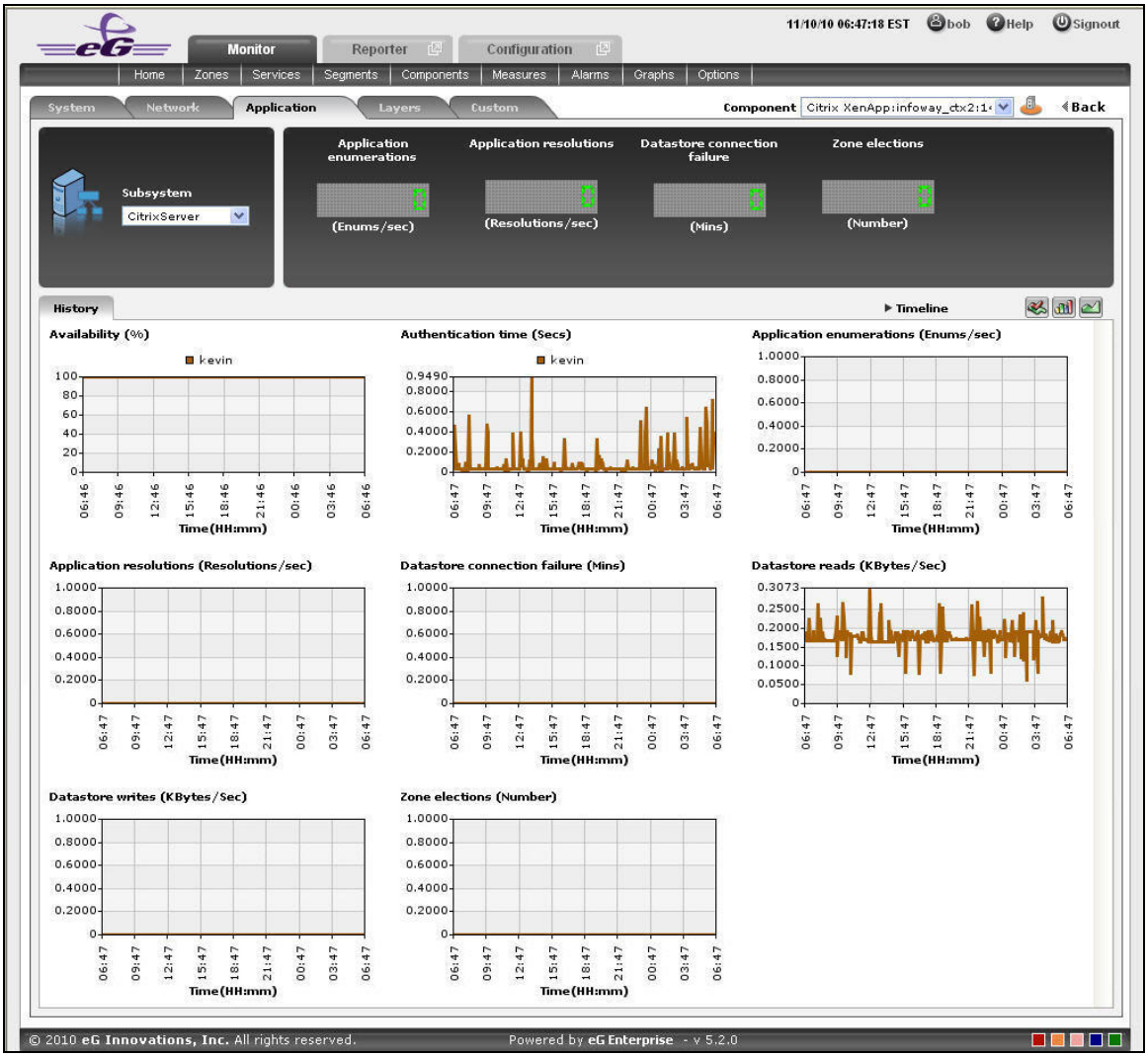


Figure 3.58: The CitrixServer Subsystem

The contents of the **CitrixServer** subsystem that then appears (see Figure 3.58) are as follows:

1. The dashboard begins with digital displays that report the current values of pre-configured metrics; typically, critical server-related metrics can be configured for display here. Using these displays, you can quickly visualize the overall health of the server.
2. The **History** tab page that follows the **Digital display** section offers measure graphs of pre-configured metrics, which help analyze the performance of the Citrix server over time. By quickly cross-correlating and time-correlating across these metrics, you can rapidly identify the root-cause of many performance issues.
3. By default, these historical graphs track the time-of-day variations in the performance of the Citrix server during the last 24 hours. You can override this default timeline by following the steps discussed below:
   - Click on the  icon at the top of the **Application Dashboard**.
   - In the **Dashboard Settings** window that appears, select **History Graph** from the **Default Timeline for** list.
   - Then, choose a **Timeline** for the graph.
   - Finally, click the **Update** button.
4. To change the timeline of all the measure graphs at one shot, just click on the **Timeline** link at the right, top corner of the **History** tab page. To alter the timeline for a single graph, just click on that graph - this will enlarge the graph. You can change the **Timeline** of the graph in the enlarged mode.

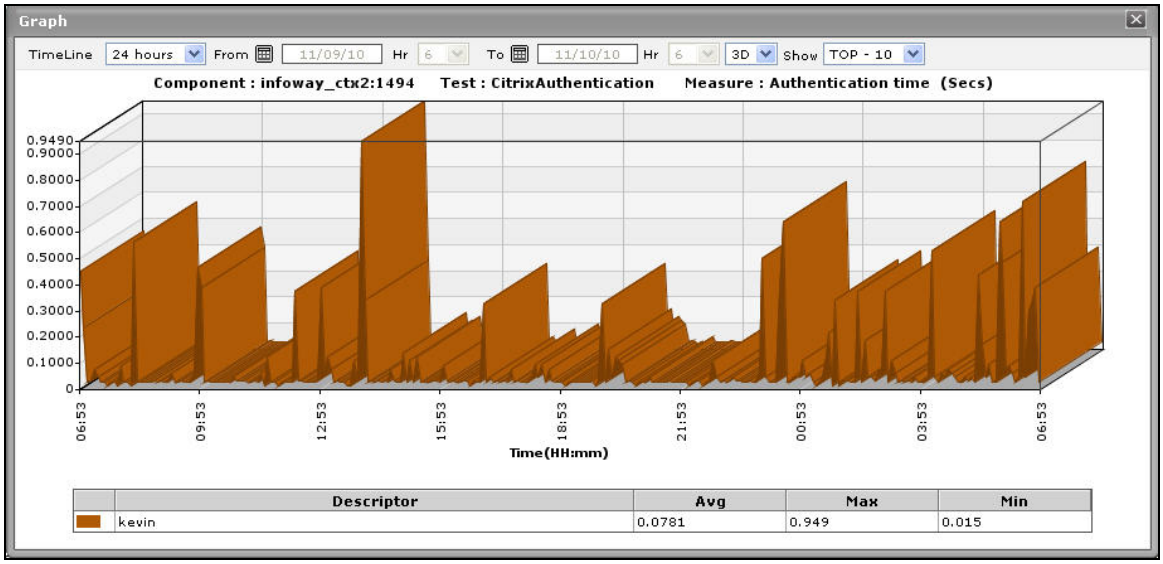


Figure 3.59: An enlarged measure graph in the History tab page of the CitrixServer dashboard

5. In case of graphs that plot values for multiple descriptors, you can also change the number of descriptors for which the graph should plot values. By default, the enlarged graph reveals the variations in the performance of the **TOP-10** descriptors. If need be, you can pick a different **TOP-N** or **LAST-N** option from the **Show** list in the enlarged graph.
6. Instead of these measure graphs, you can, if required, view summary graphs of the server-related measures in the **History** tab page. For this, click on the  icon at the right, top corner of the **History** tab page. Summary graphs help you figure out the percentage of time during the last 24 hours (by default) the quality of service delivered by the Citrix XenApp server was compromised. While monitoring mission-critical applications that are governed by rigid service level agreements, summary graphs will help you determine whether the guaranteed availability of the server was met or not, and if not, how often was the server not available.
7. You can override the default timeline (of 24 hours) of the summary graphs by following the steps discussed below:
   - Click on the  icon at the top of the **Application Dashboard**.
   - In the **Dashboard Settings** window that appears, select **Summary Graph** from the **Default Timeline for** list.
   - Then, choose a **Timeline** for the graph.
   - Finally, click the **Update** button.

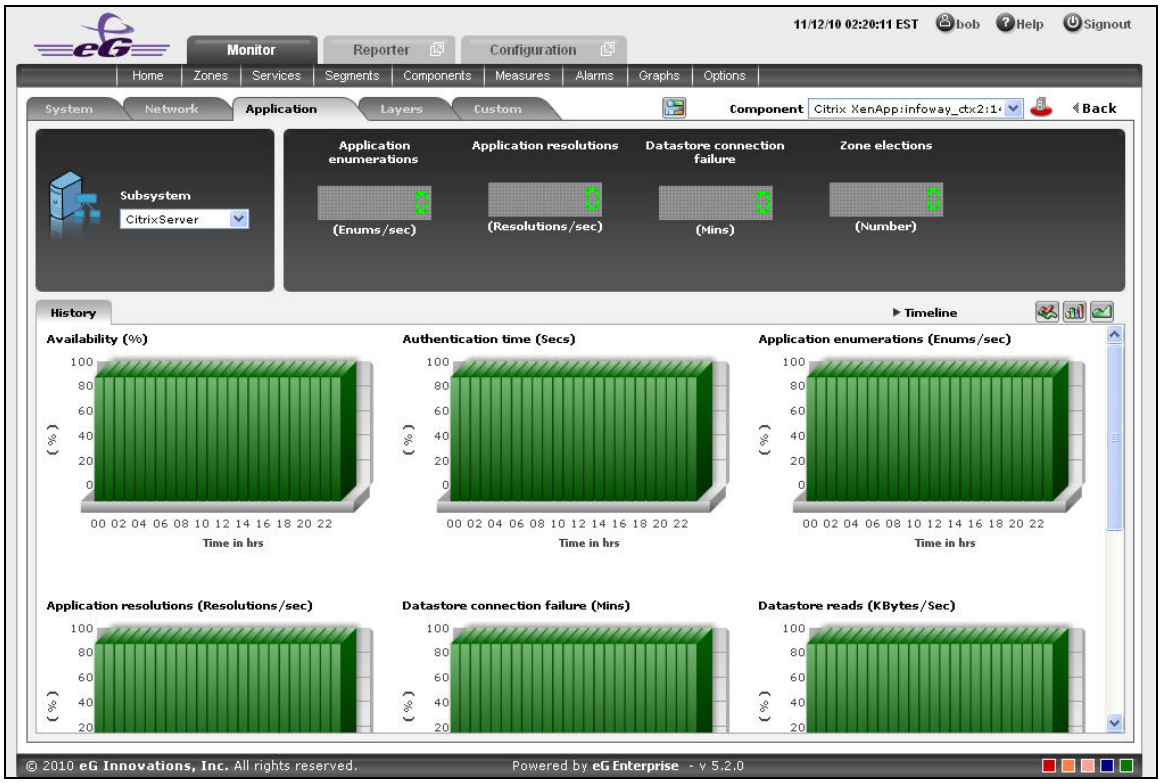


Figure 3.60: Summary graphs displayed in the History tab page of the CitrixServer Dashboard

8. Here again, you can change the **Timeline** of all the summary graphs by clicking on the **Timeline** link in 3.9.1, or click on a graph, enlarge it, and change its **Timeline** in the enlarged mode. Also, though the graph plots hourly summary values by default, you can pick a different **Duration** for the graph in the enlarged mode, so that daily/monthly performance summaries can be analyzed.
9. You can click on the icon at the right, top corner of the **History** tab page to view trend graphs of the metrics. By default, these trend graphs plot the maximum and minimum health state values for every hour of the last 24 hours (by default). The default timeline of 24 hours can be overridden by following the steps discussed below:
   - Click on the icon at the top of the **Application Dashboard**.
   - In the **Dashboard Settings** window that appears, select **Trend Graph** from the **Default Timeline for** list.
   - Then, choose a **Timeline** for the graph.
   - Finally, click the **Update** button.
10. Using these trend graphs, you can understand the variations in the overall health of the Citrix

XenApp server during the last 24 hours (by default), deduce the future health trends, and accordingly recommend changes to the application.

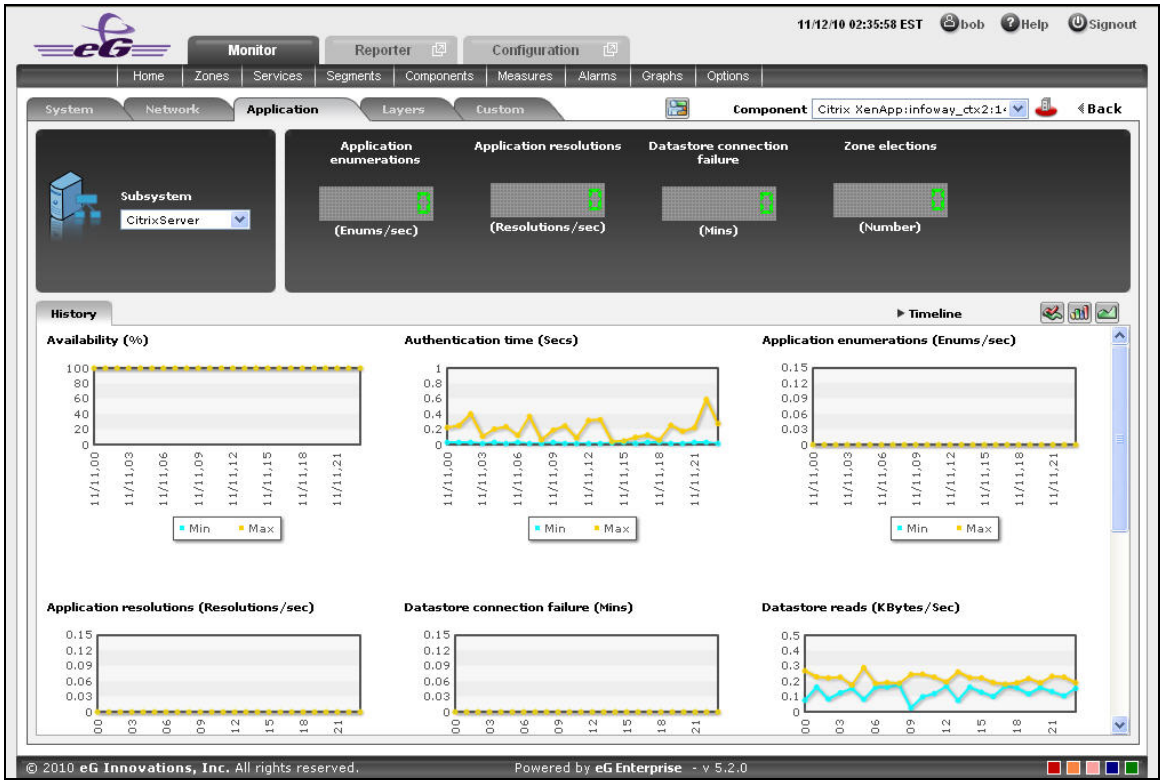


Figure 3.61: Trend graphs displayed in the History tab page of the CitrixServer Dashboard

11. Here again, you can change the **Timeline** of all the trend graphs by clicking on the **Timeline** link in 3.9.1, or click on a graph, enlarge it, and change its **Timeline** in the enlarged mode. Also, though the graph plots hourly trend values by default, you can pick a different **Duration** for the graph in the enlarged mode, so that daily/monthly performance trends can be analyzed. IThe timeline of this graph can be altered at runtime by

12. Also, by default, the trend graph only plots the minimum and maximum values registered by a measure. Accordingly, the **Graph** type is set to **Min/Max** in the enlarged mode. If need be, you can change the **Graph** type to **Avg**, so that the average trend values of a measure are plotted for the given **Timeline**. Such a graph will enable you to assess whether the memory resources were utilized effectively or not, over time.

13. Likewise, you can also choose **Sum** as the **Graph** type to view a trend graph that plots the sum of the values of a chosen measure for a specified timeline. For instance, a 'sum of trends' Availability will enable you to analyze, on an hourly/daily/monthly basis (depending upon the **Duration** chosen), whether the server was available during the specified timeline.

Note:

In case of descriptor-based tests, the **Summary** and **Trend** graphs displayed in the **History** tab page typically plot the values for a single descriptor alone. To view the graph for another descriptor, pick a descriptor from the drop-down list made available above the corresponding summary/trend graph.

14. At any point in time, you can switch to the measure graphs by clicking on the  button.

15. Typically, the **History** tab page displays measure, summary, and trend graphs for a default set of measures. If you want to add graphs for more measures to this tab page or remove one/more measures for which graphs pre-exist in this tab page, then, do the following:

- Click the  button at the top of the dashboard.
- The **Dashboard Settings** window then appears. From the **Module** list of Figure 3.57, pick **Application**, choose **CitrixServer** as the **Sub-System**, and then, select **History Graph** from the **Add/Delete Measures for** list.
- The measures for which graphs pre-exist in the **History** tab page will be automatically displayed in the **Existing Value(s)** list. To delete a measure, and in effect, its corresponding graph as well, select the measure from the **Existing Value(s)** list, click the **Delete** button, and then click the **Update** button.
- To add a new graph, first, pick the **Test** that reports the measure for which a graph is to be generated.
- Next, select the **Measure** of interest.
- Provide a **Display** name for the measure. Then, click the **Add** button to add the measure to the **Existing Values(s)** list. Finally, click the **Update** button.
- This will add a new measure, summary, and trend graph for the chosen measure to the **History** tab page.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the Dashboard Settings window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the  button will not appear.

3.9.1.2

3.9.1.3 CitrixSessions

If you require an integrated dashboard for analyzing the present/past performance and problem information pertaining to the sessions that are executed on the Citrix XenApp application, select the **CitrixSessions** option from the **Subsystem** list. This option helps you to efficiently and accurately diagnose the root-cause of the session-related abnormalities. Using this single, central dashboard, you can ascertain the following quickly and easily:

- Are all the sessions active on this particular application?
- How long has a particular session been in an idle state? What is the exact time period of the idle session?
- Are there any disconnected sessions?
- Has the application been unavailable during a particular session?

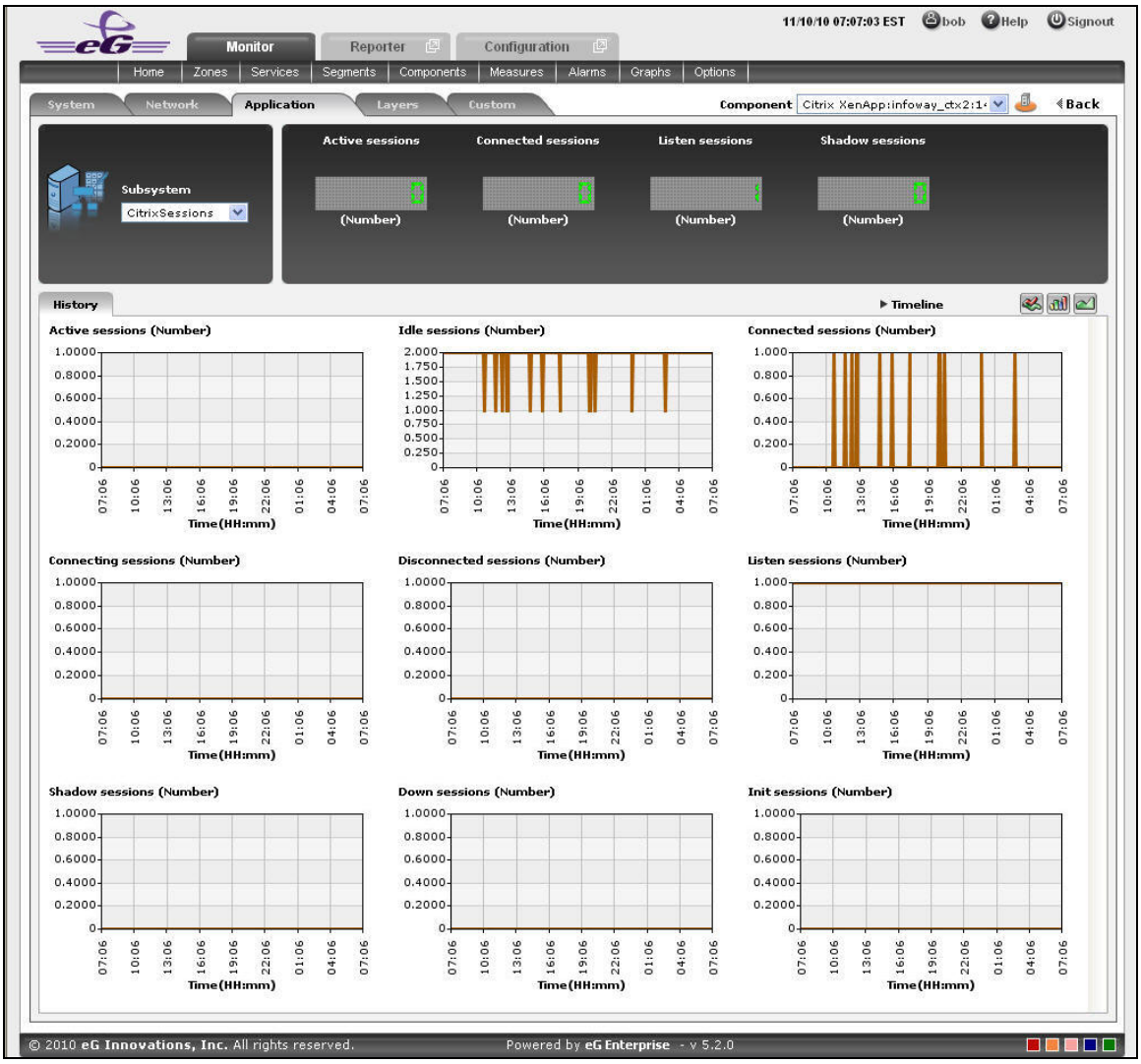


Figure 3.62: The CitrixSessions Dashboard

The contents of this dashboard are discussed hereunder:

1. The **Digital display** section, displays the session activity in numbers. For instance the number of active session will be displayed in this section which can be viewed at a single glance. Clicking on a Digital display will lead you to Figure 3.62, which displays the layer and test that reports the measure.

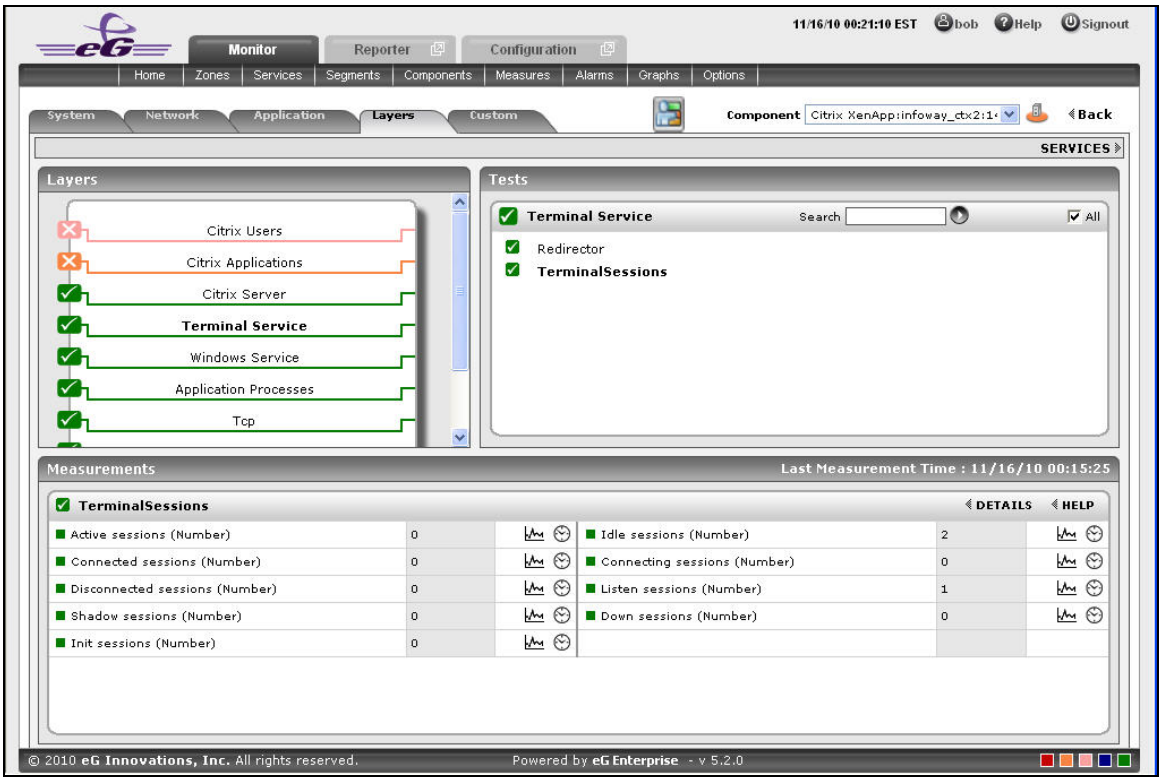


Figure 3.63: Clicking on a digital display in the CitrixSessions dashboard

2. For historically analyzing the session activity of the Citrix XenApp application, click on the **History** tab page. This tab page displays time-of-day graphs for all the thread-related measures for default duration of 24 hours. You can override this default timeline (of 24 hours) by following the steps below:

   - Click on the icon at the top of the **Application Dashboard**.
   - In the **Dashboard Settings** window that appears, select **History Graph** from the **Default Timeline for** list.
   - Then, choose a **Timeline** for the graph.
   - Finally, click the **Update** button.

3. Say, you suddenly notice that the session state has been idle for a while; in such a case, you can use these measure graphs to figure out when during the last 24 hours the session has been idle. If required, you can even look beyond the last 24 hours - i.e., you can find out whether the anomaly originated much earlier. For this, you just need to click on the graph of interest to you. This will enlarge the graph; in the enlarged mode, you can alter the graph **TIMELINE**, so that the performance of that measure can be analyzed over a broader time window. In this mode,

you can even change the graph dimension from **3D** to **2D,** or vice-versa.

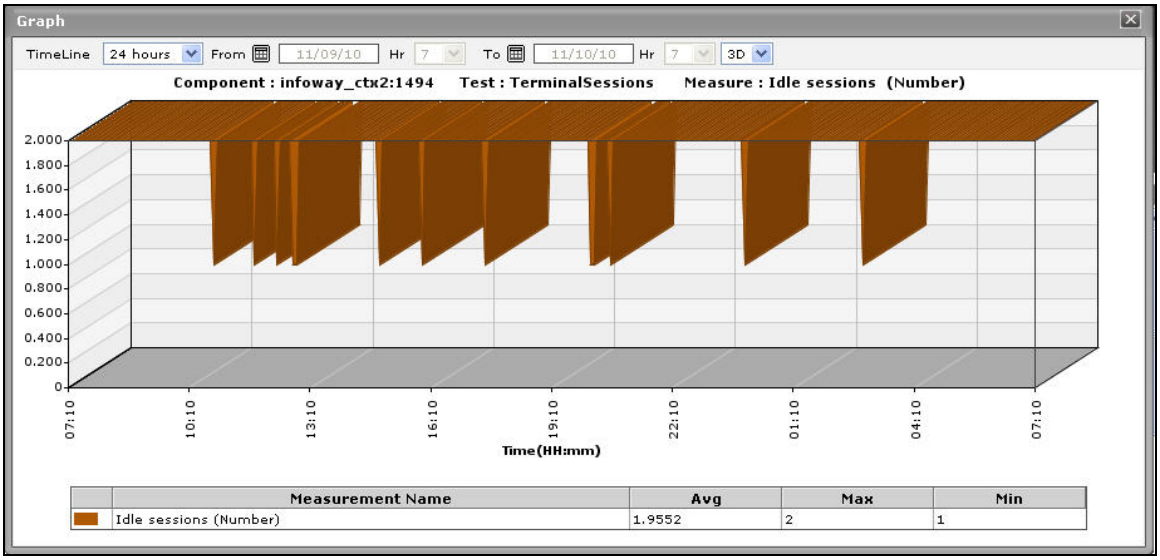


Figure 3.64: An enlarged measure graph in the History tab page of the Citrix Session dashboard

4. To view summary graphs on Idle sessions state instead of the default measure graphs, just click on the icon at the right, top corner of the **History** tab page. Figure 3.65 will then appear. The summary graphs of Figure 3.65 reveal the percentage of time during the last 24 hours (by default) the Citrix XenApp application has been idle. These graphs will therefore be useful to figure out the type of issues (whether critical/major/minor) the application was experiencing. These graphs also help to determine whether the assured service levels were delivered or not.
5. The default duration (of 24 hours) of the summary graphs can be overridden by following the procedure discussed below:
   - Click on the icon at the top of the **Application Dashboard**.
   - In the **Dashboard Settings** window that appears, select **Summary Graph** from the **Default Timeline for** list.
   - Then, choose a **Timeline** for the graph.
   - Finally, click the **Update** button.

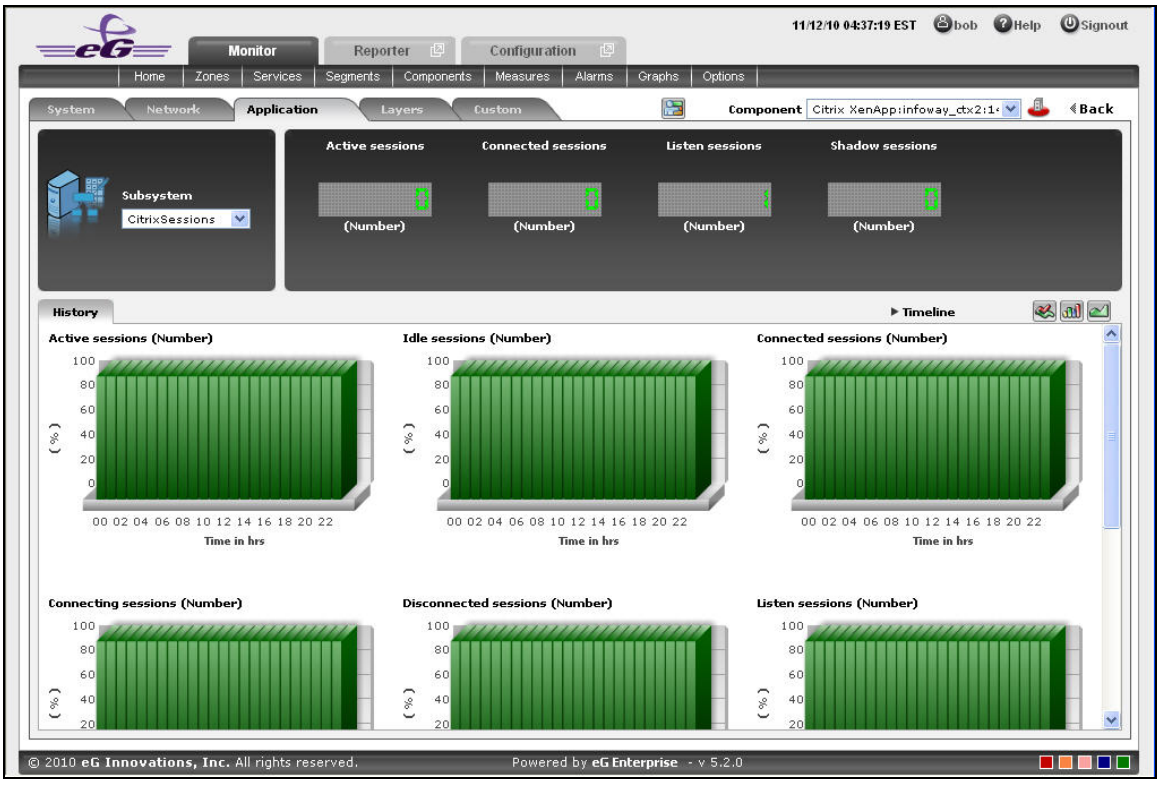


Figure 3.65: Summary graphs displayed in the History tab page of the CitrixSessions Dashboard

6. Use the **Timeline** link at the right, top corner of the tab page to change the timeline of all the summary graphs at one shot. For altering the timeline of a single graph, click on it; this will enlarge the graph. In the enlarged mode, you can change the **Timeline** of the summary graph and modify the dimension (3D/2D) of the graph. Also, by default, hourly summaries are plotted in the summary graph; you can configure these graphs to plot daily/monthly summaries instead by picking the relevant option from the **Duration** list in the enlarged mode.

7. If you want to view the past trends of various sessions, click on the icon at the right, top corner of the **History** tab page. Figure 3.66 will then appear. Using the trend graphs displayed in Figure 3.66, you can better assess the current sessions of your application and can accordingly plan its future availability. By default, these trend graphs plot the maximum and minimum values registered by every session related measure during every hour for the last 24 hours. From this data, you can clearly figure out when during the last 24 hours the application performance has peaked and when it has been below-normal.

8. The default duration (of 24 hours) of the trend graphs can be overridden by following the procedure discussed below:

- Click on the icon at the top of the **Application Dashboard**.
- In the **Dashboard Settings** window that appears, select **Trend Graph** from the **Default Timeline for** list.
- Then, choose a **Timeline** for the graph.
- Finally, click the **Update** button.

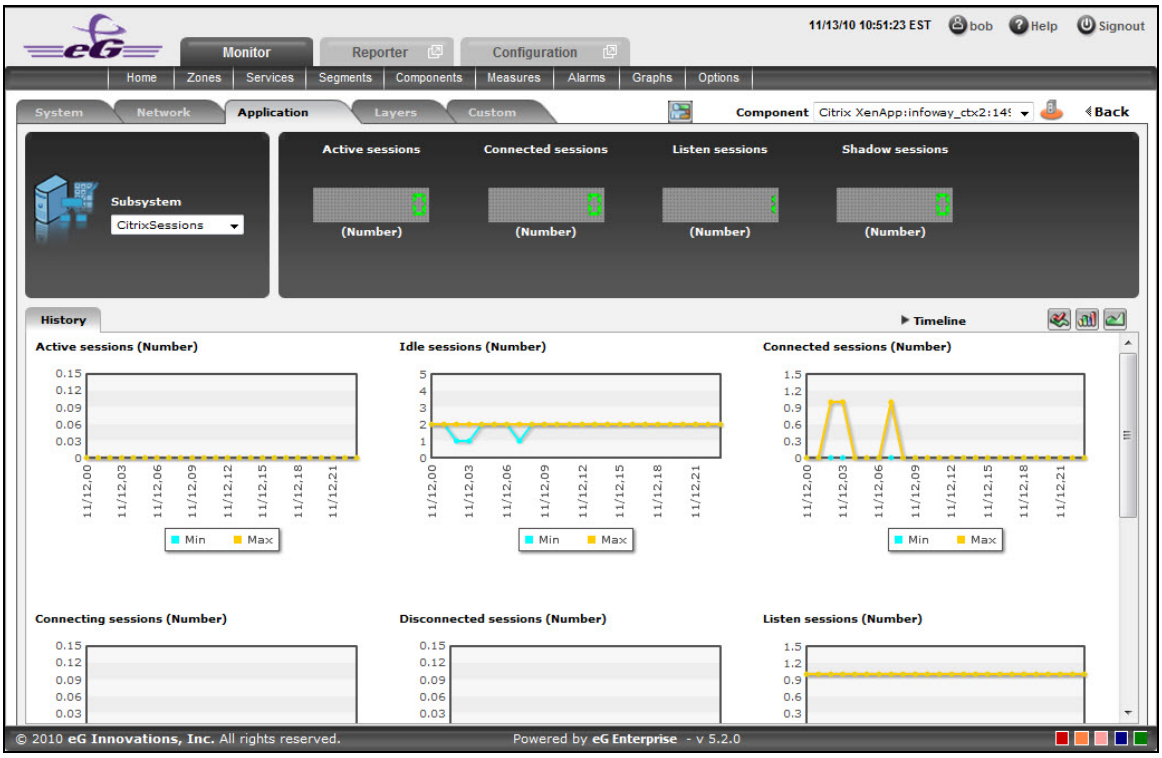


Figure 3.66: Trend graphs displayed in the History tab page of the CitrixSessions Dashboard

9. Use the **Timeline** link at the right, top corner of the tab page to change the timeline of all the trend graphs at one shot. For altering the timeline of a single graph, click on it; this will enlarge the graph. In the enlarged mode, you can change the **Timeline** of the trend graph and modify the dimension (3D/2D) of the graph. Also, by default, hourly trends are plotted in the trend graph; you can configure these graphs to plot daily/monthly trend values instead by picking the relevant option from the **Duration** list in the enlarged mode. Moreover, by default, the trend graphs plot only the minimum and maximum values registered by a measure during the specified timeline - this graph will enable you to isolate those times at which performance of that measure had peaked and the times it had fared poorly. For instance, using the default trend graph for the Idle sessions measure, you can clearly identify when too many sessions were idle and when the number of Idle sessions were minimum. If need be, you can select the **Avg**

option from the **Graph type** list in the enlarged mode to make sure that the trend graph plots the average trend values for the specified timeline - in the case of the above example, such a graph will help you understand how the number of Idle sessions has varied during the set timeline. Alternatively, you can select the **Sum** option from the **Graph type** list to have the trend graph plot the sum of trends for the specified timeline.

Note:

In case of descriptor-based tests, the **Summary** and **Trend** graphs displayed in the **History** tab page typically plot the values for a single descriptor alone. To view the graph for another descriptor, pick a descriptor from the drop-down list made available above the corresponding summary/trend graph.

10. At any point in time, you can switch to the measure graphs by clicking on the  button.

11. Typically, the **History** tab page displays measure, summary, and trend graphs for a default set of measures. If you want to add graphs for more measures to this tab page or remove one/more measures for which graphs pre-exist in this tab page, then, do the following:

    - Click the  button at the top of the dashboard.
    - The **Dashboard Settings** window then appears. From the **Module** list of Figure 3.57, pick **Application**, choose **CitrixSessions** as the **Sub-System**, and then, select **History Graph** from the **Add/Delete Measures for** list.
    - The measures for which graphs pre-exist in the **History** tab page will be automatically displayed in the **Existing Value(s)** list. To delete a measure, and in effect, its corresponding graph as well, select the measure from the **Existing Value(s)** list, click the **Delete** button, and then click the **Update** button.
    - To add a new graph, first, pick the **Test** that reports the measure for which a graph is to be generated.
    - Next, select the **Measure** of interest.
    - Provide a **Display** name for the measure. Then, click the **Add** button to add the measure to the **Existing Values(s)** list. Finally, click the **Update** button.
    - This will add a new measure, summary, and trend graph for the chosen measure to the **History** tab page.

    Note:

    Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards

using the Dashboard Settings window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

3.9.1.4 CitrixApplications

Select the **CitrixApplications** option from the **Subsystem** list to know how efficiently the applications are used by the Citrix XenApp. Upon selection of this **Subsystem** Figure 3.67 will appear.

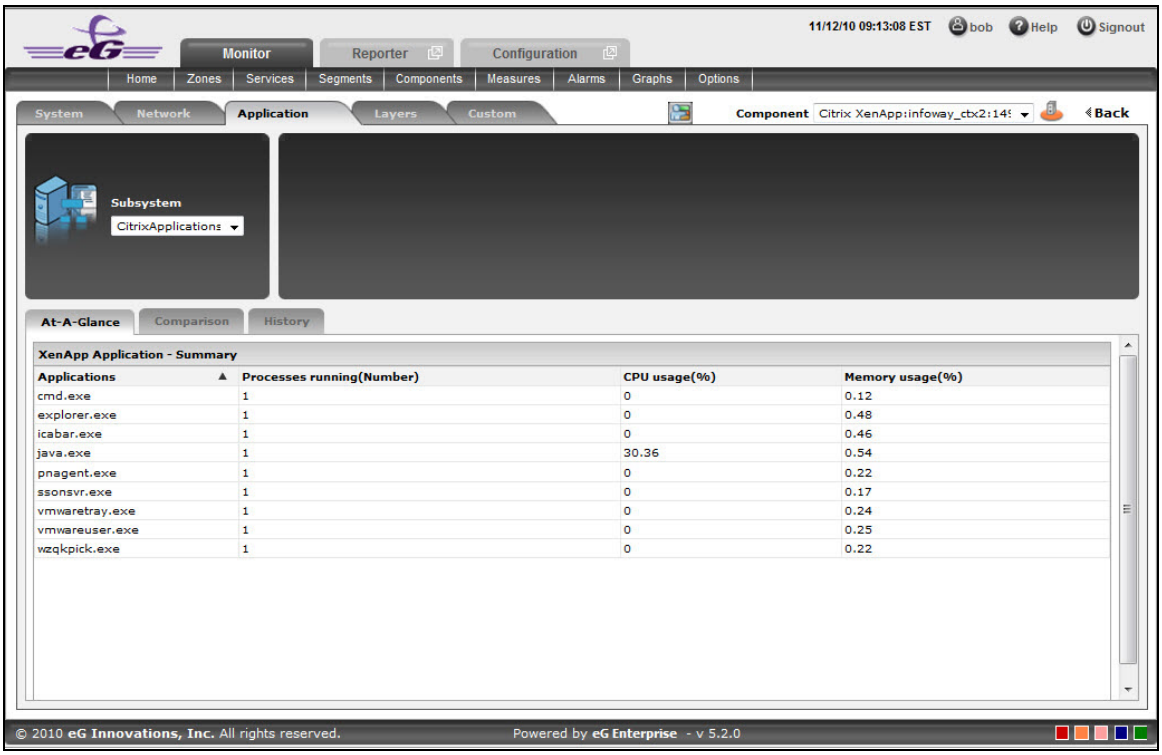


Figure 3.67: The CitrixApplications Dashboard

The contents of this dashboard are as follows:

1. The **At-A-Glance** tab page (see 3.9.1) contains a **XenApp Application-Summary** section which provides an insight view of the **Applications** that are available for the Citrix XenApp. The Applications can either be sorted in alphabetical order or can be sorted according to their current health status such as **Processes running**, **CPU Usage** and **Memory usage**.
2. As shown in 3.9.1, the **Comparison** tab page that follows the **At-A-Glance** tab page provides a series of top-10 charts, using which you can quickly isolate those Applications that are leading the lot in the following default performance areas: Instances, amount of CPU used, amount of

memory used. This default list of performance areas (i.e., measures) for top-n chart generation can be overridden by following the steps discussed below:

- Click on the  icon at the top of the **Application Dashboard**. In the **Dashboard Settings** window that appears, select **Application** from the **Module** list, and **CitrixApplications** from the **Sub-System** list.
- To add new measures for which top-n graphs are to be displayed in the **Comparison** tab page, first, pick the **Comparison Graph** option from the **Add/Delete Measures for** list. Upon selection of this option, the pre-configured measures for comparison graphs will appear in the **Existing Value(s)** list.
- Next, select the **Test** that reports the said measure, pick the measure of interest from the **Measures** list, provide a **Display** name for the measure, and click the **Add** button to add the chosen measure to the **Existing Value(s)** list.
- If you want to delete one/more measures for which comparison graphs pre-exist in the **Comparison** tab page, then, as soon as you choose the **Comparison Graph** option from the **Add/Delete Measures for** list, pick any of the displayed measures from the **Existing Value (s)** list, and click the **Delete** button.
- Finally, click the **Update** button to register the changes.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the Dashboard Settings window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the  button will not appear.

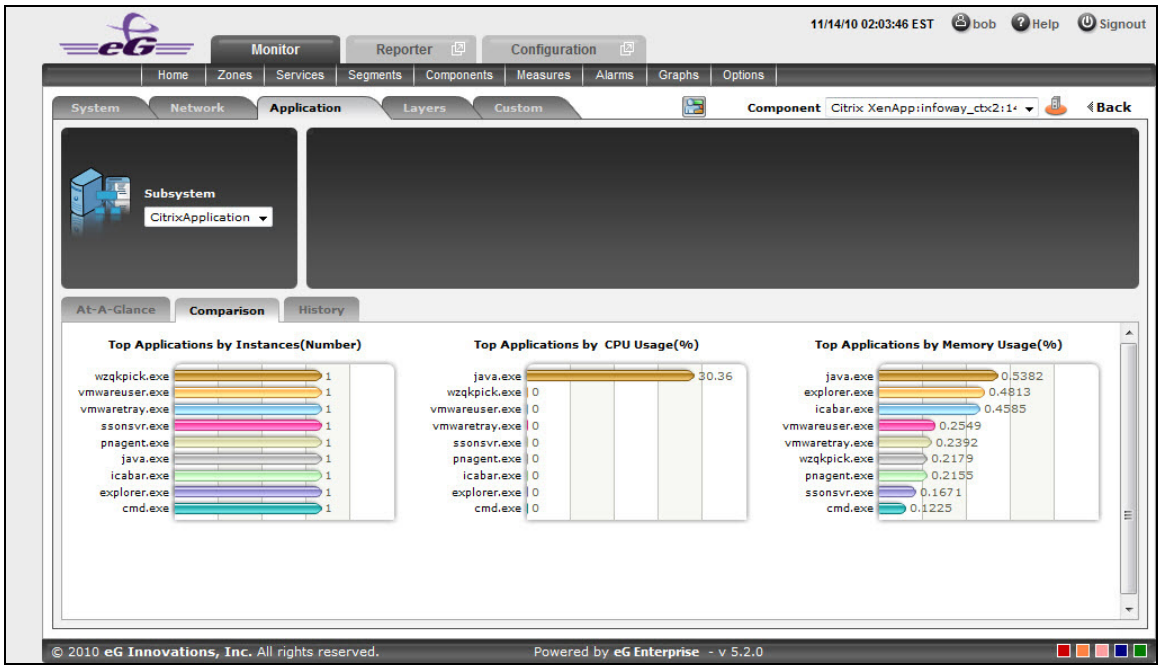


Figure 3.68: The Comparison tab page of a CitrixApplication dashboard

3. If an application slowdown can be attributed to the lack of adequate CPU or Memory resources, then these top-10 bar charts can aid you in swiftly nailing the exact application that could be serving as the source of this CPU or memory contention.

4. Typically, these bar charts depict the current usage data. Sometimes however, you might want to detect which Application was over-utilizing any resource at some point of time in the past. In such a case, you will have to click on the corresponding graph in the **Comparison** tab page to enlarge it. In the enlarged mode, you can click on the **Compare History** link, so that you can alter the graph **Timeline**, and view which application was being fully utilized during the specified timeline.

5. The **History** tab page in Figure 3.69 below, by default, provides a series of measure graphs that reveal how the Application has been performing over the default duration of the last 24 hours. The CPU and Memory utilization as well as the number of Processes that are running currently can be identified. The default duration of 24 hours can be overridden using the procedure discussed below:

   - Click on the icon at the top of the **Application Dashboard**.
   - In the **Dashboard Settings** window that appears, select **History Graph** from the **Default Timeline for** list.

- Then, choose a **Timeline** for the graph.
- Finally, click the **Update** button.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the Dashboard Settings window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

Figure 3.69: The History tab page of CitrixApplication dashboard

6. If need be, you can even alter the timeline of all these measure graphs so that you can analyze performance across days and weeks; for this, simply click the **Timeline** link at the right, top corner of the **History** tab page and change the timeline for the graphs using the calendar that pops out. To change the timeline of a single graph alone, simply click on that graph to enlarge it, and then modify the **Timeline** of the graph in the enlarged mode. In the enlarged mode, you can even change the dimension of the measure graph (3d / 2d). Figure 3.70 shows an enlarged measure graph.

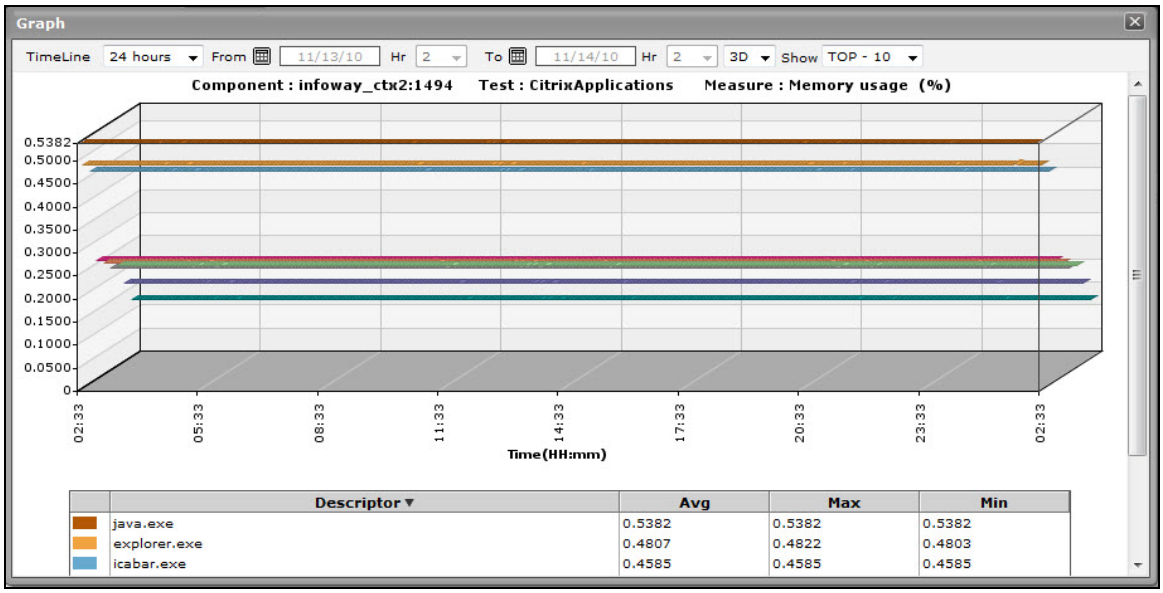


Figure 3.70: An enlarged measure graph in the History tab page of the CitrixApplications dashboard

7. To determine the service level achievements / slippages of the Citrix Application, you need to view summary graphs of the measures and not the default measure graphs. For this, just click on the icon at the right, top corner of the **History** tab page.

8. Besides revealing the efficiency of your administrative staff in recognizing bottlenecks and mitigating them, these summary graphs also indicate whether the CitrixApplication has been able to maintain the assured performance levels during the default duration of 24 hours.

9. To override this default duration, follow the steps below:

   - Click on the icon at the top of the **Application Dashboard**.
   - In the **Dashboard Settings** window that appears, select **Summary Graph** from the **Default Timeline for** list.
   - Then, choose a **Timeline** for the graph.
   - Finally, click the **Update** button.

10. In case of the summary graphs too, you can change the **Timeline** of all graphs by clicking on the **Timeline** link at the right, top corner of the **History** tab page. To alter the timeline of a single graph, here again, you will have to click on that graph, enlarge it, and modify the timeline. Also, by default, hourly summaries are plotted in the summary graph; you can configure these graphs to plot daily/monthly summaries instead by picking the relevant option from the **Duration** list in the enlarged mode.

11. To analyze past trends in the loading/unloading of classes, click on the icon at the right, top corner of the **History** tab page.

12. These trend graphs, by default, plot the minimum and maximum values that every measure registered during each hour of the last 24 hours (by default). Using such graphs, you can accurately point to the time during which the performance of the Application was at peak, and the times at which there was a lull. By carefully observing these past trends, you can effectively analyze the performance of the application, predict future performances accordingly, and suggest measures to enhance the efficiency. Here again, you can change the timeline of all graphs using the **Timeline** link in Figure 3.73, or just a particular graph by clicking on it and enlarging it.

13. For changing the default duration (of 24 hours) of the trend graphs, do the following:

    - Click on the icon at the top of the **Application Dashboard**.
    - In the **Dashboard Settings** window that appears, select **Trend Graph** from the **Default Timeline for** list.
    - Then, choose a **Timeline** for the graph.
    - Finally, click the **Update** button.

14. In addition, when a trend graph is enlarged, it is not just the **Timeline** that you can modify. The **Duration** of the graph can also be altered. By default, trend graphs reveal only the hourly trends in performance. By picking the relevant option from the **Duration** list, you can ensure that the trend graph in question plots daily/monthly trend values instead. Also, in the enlarged mode, the **Graph type** can also be modified. Since the default **Graph type** is **Min/Max**, the trend graph, by default, reveals the minimum and maximum values registered by a measure. If need be, you can select the **Avg** or **Sum** option from the **Graph type** list to plot average trend values of a measure or sum of trends (as the case may be) in the graph.

    Note:

    In case of descriptor-based tests, the **Summary** and **Trend** graphs displayed in the **History** tab page typically plot the values for a single descriptor alone. To view the graph for another descriptor, pick a descriptor from the drop-down list made available above the corresponding summary/trend graph.

15. At any point in time, you can switch to the measure graphs by clicking on the button.

16. Typically, the **History** tab page displays measure, summary, and trend graphs for a default set of measures. If you want to add graphs for more measures to this tab page or remove one/more measures for which graphs pre-exist in this tab page, then, do the following:

- Click the button at the top of the dashboard.
- The **Dashboard Settings** window then appears. From the **Module** list of Figure 3.57, pick **Application**, choose **CitrixApplicaations** as the **Sub-System**, and then, select **History Graph** from the **Add/Delete Measures for** list.
- The measures for which graphs pre-exist in the **History** tab page will be automatically displayed in the **Existing Value(s)** list. To delete a measure, and in effect, its corresponding graph as well, select the measure from the **Existing Value(s)** list, click the **Delete** button, and then click the **Update** button.
- To add a new graph, first, pick the **Test** that reports the measure for which a graph is to be generated.
- Next, select the **Measure** of interest.
- Provide a **Display** name for the measure. Then, click the **Add** button to add the measure to the **Existing Values(s)** list. Finally, click the **Update** button.
- This will add a new measure, summary, and trend graph for the chosen measure to the **History** tab page.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the Dashboard Settings window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

3.9.1.5 CitrixUsers

Select the **CitrixUsers** option from the **Subsystem** list to know how many Users are currently accessing the Citrix XenApp application. Upon selection of this **Subsystem** Figure 3.71 will appear.

Figure 3.71: The CitrixUsers Dashboard

The contents of this dashboard are as follows:

1. A dial chart for **Percent new logins** and digital displays for various user sessions provide an insight view of the user login information at a single glance Clicking on a dial chart / digital display will lead you to the corresponding layer and test that reports the measure.
2. The **At-A-Glance** tab page (see Figure 3.71) contains a **Citrix Users** left panel which lists out the number of users who are currently active for this session. A context-sensitive right panel provides an insight view of the user information that is available for the Citrix XenApp. The user's processes information can be viewed at a single glance with the help of dial charts and cylindrical charts.

3. The **Citrix Sessions – Summary** (see Figure 3.71) in the right panel indicates the user session information such as Login time, Idle Time and Total Connected Time, at a single glance.

4. The **Citrix User Application Summary** (see Figure 3.71) lists the number of Applications that are currently used by the user. The applications can be sorted either in alphabetical order or in accordance with the application specific information that is available next to each application name.

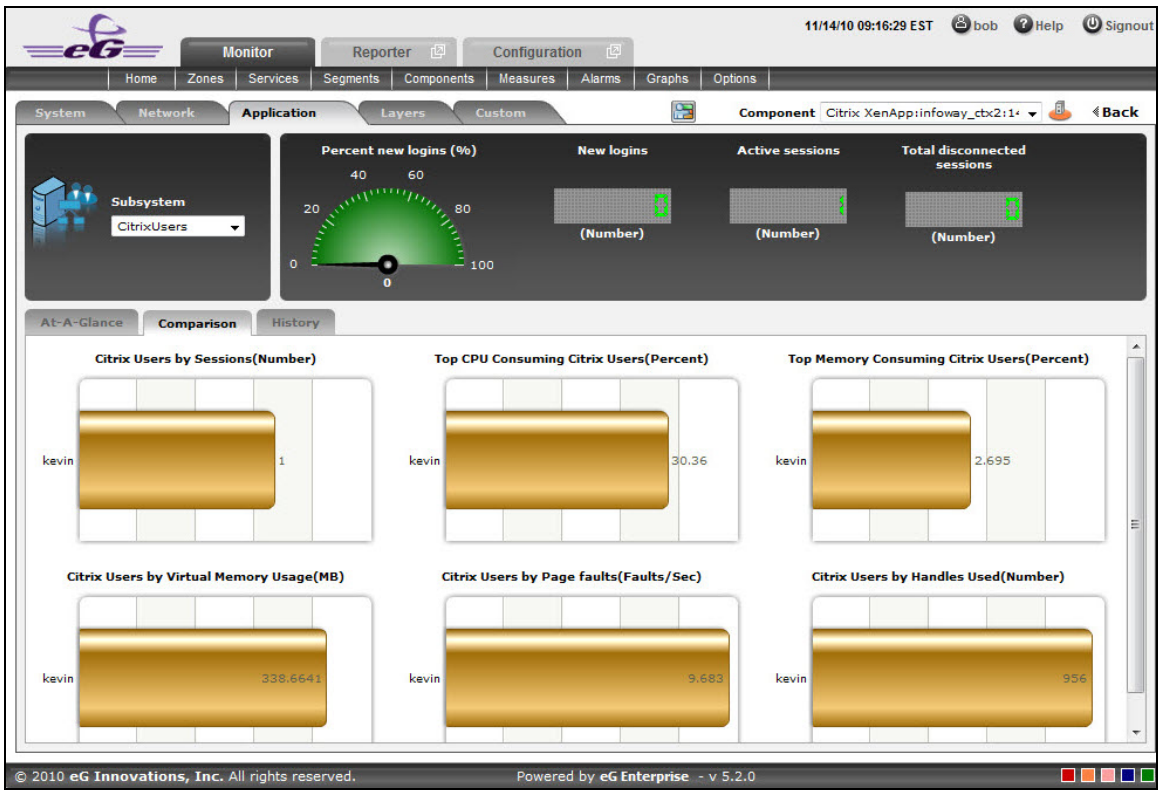


Figure 3.72: The Comparison tab page of CitrixUsers dashboard

5. As shown in Figure 3.72, the **Comparison** tab page that follows the **At-A-Glance** tab page provides a series of top-10 charts, using which you can quickly isolate the Users who are currently active for this session. These graphs provide an insight view of various session related activities that are performed for each user login. This default list of performance areas (i.e., measures) for top-n chart generation can be overridden by following the steps discussed below:

   - Click on the icon at the top of the **Application Dashboard**. In the **Dashboard Settings** window that appears, select **Application** from the **Module** list, and **CitrixUsers** from the **Sub-System** list.

- To add new measures for which top-n graphs are to be displayed in the **Comparison** tab page, first, pick the **Comparison Graph** option from the **Add/Delete Measures for** list. Upon selection of this option, the pre-configured measures for comparison graphs will appear in the **Existing Value(s)** list.
- Next, select the **Test** that reports the said measure, pick the measure of interest from the **Measures** list, provide a **Display** name for the measure, and click the **Add** button to add the chosen measure to the **Existing Value(s)** list.
- If you want to delete one/more measures for which comparison graphs pre-exist in the **Comparison** tab page, then, as soon as you choose the **Comparison Graph** option from the **Add/Delete Measures for** list, pick any of the displayed measures from the **Existing Value (s)** list, and click the **Delete** button.
- Finally, click the **Update** button to register the changes.

  Note:

  Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the Dashboard Settings window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the  button will not appear.

6. If an application slowdown can be attributed to the lack of adequate resources, then these top-10 bar charts can aid you in swiftly nailing the exact resource location that could be serving as the source of this resource contention.
7. Typically, these bar charts depict the current usage data. Sometimes however, you might want to detect which Application was over-utilizing the resources at some point of time in the past. In such a case, you will have to click on the corresponding graph in the **Comparison** tab page to enlarge it. In the enlarged mode, you can click on the **Compare History** link, so that you can alter the graph **Timeline**, and view which user was the leading memory consumer during the specified timeline.
8. The **History** tab page below, by default, provides a series of measure graphs that reveal how the Application has been performing over the default duration of the last 24 hours. The CPU and Memory utilization as well as the number of Processes that are running currently can be identified. The default duration of 24 hours can be overridden using the procedure discussed below:

- Click on the icon at the top of the **Application Dashboard**.
- In the **Dashboard Settings** window that appears, select **History Graph** from the **Default Timeline for** list.
- Then, choose a **Timeline** for the graph.
- Finally, click the **Update** button.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the Dashboard Settings window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

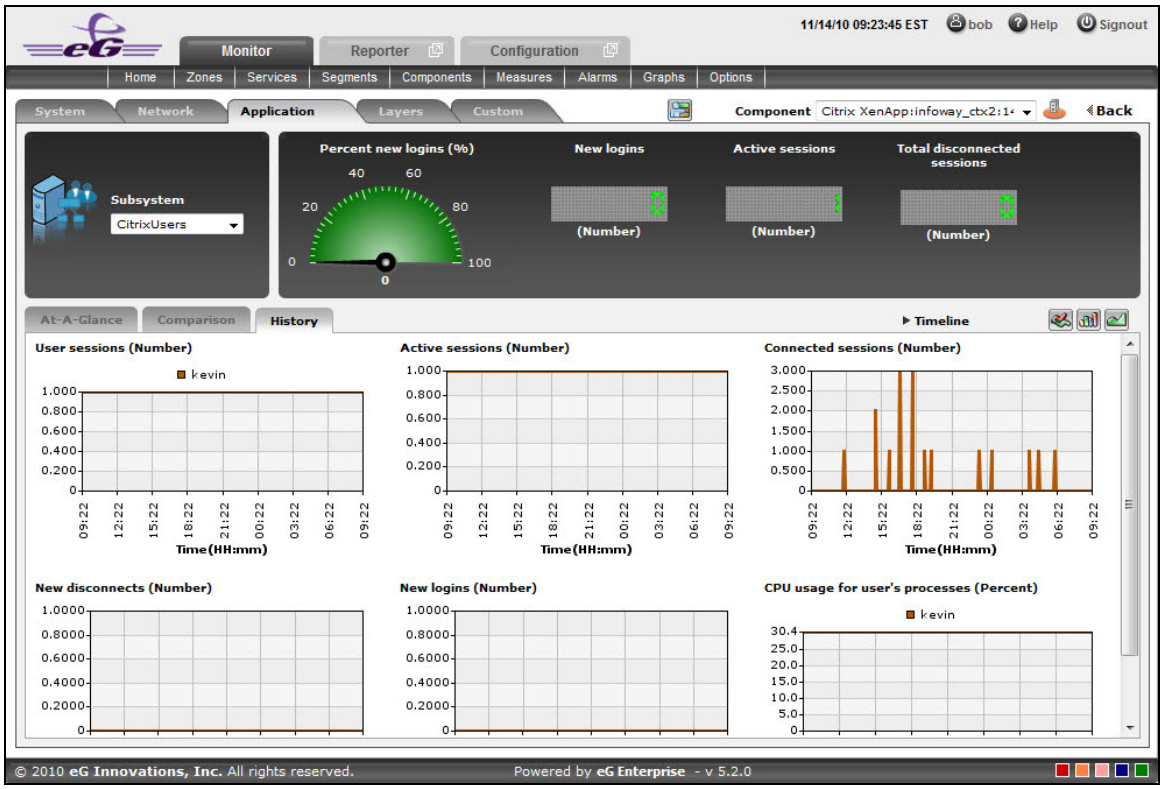


Figure 3.73: The History tab page of CitrixUsers dashboard

9. If need be, you can even alter the timeline of all these measure graphs so that you can analyze performance across days and weeks; for this, simply click the **Timeline** link at the right, top corner of the **History** tab page and change the timeline for the graphs using the calendar that pops out. To change the timeline of a single graph alone, simply click on that graph to enlarge it,

and then modify the **Timeline** of the graph in the enlarged mode. In the enlarged mode, you can even change the dimension of the measure graph (**3D** / **2D**).

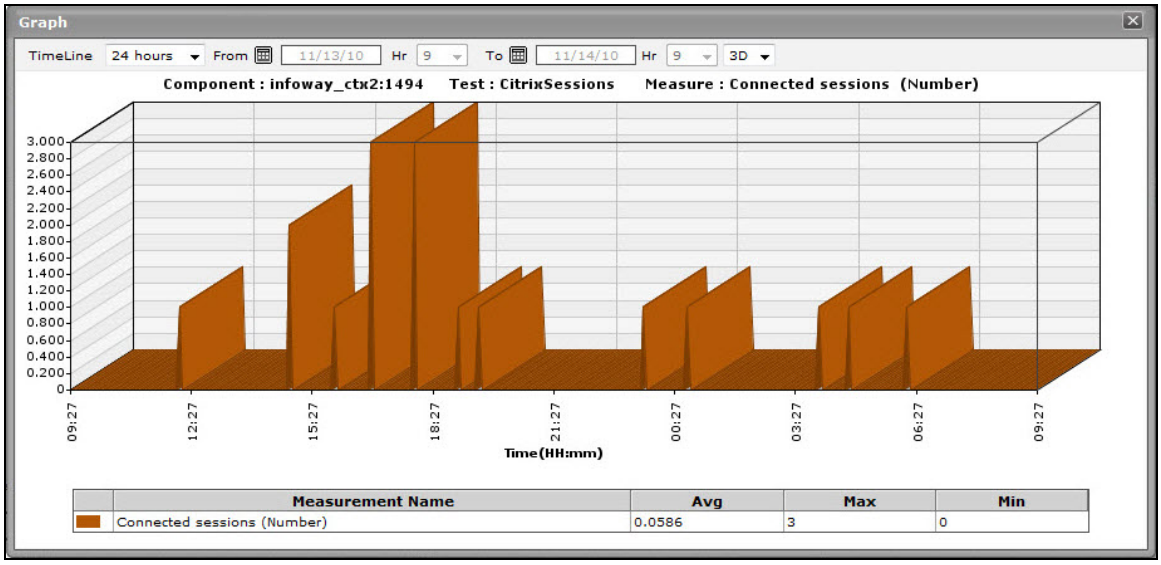


Figure 3.74: An enlarged measure graph in the History tab page of the CitrixUsers dashboard

10. To determine the service level achievements / slippages of the CitrixUsers, you need to view summary graphs of the measures and not the default measure graphs. For this, just click on the icon at the right, top corner of the **History** tab page.

11. Besides revealing the efficiency of your administrative staff in recognizing bottlenecks and mitigating them, these summary graphs also indicate whether the CitrixUsers are able to acquire the assured performance levels during the default duration of 24 hours.

12. To override this default duration, follow the steps below:

    - Click on the icon at the top of the **Application Dashboard**.
    - In the **Dashboard Settings** window that appears, select **Summary Graph** from the **Default Timeline for** list.
    - Then, choose a **Timeline** for the graph.
    - Finally, click the **Update** button.

13. In case of the summary graphs too, you can change the **Timeline** of all graphs by clicking on the **Timeline** link at the right, top corner of the **History** tab page. To alter the timeline of a single graph, here again, you will have to click on that graph, enlarge it, and modify the timeline. Also, by default, hourly summaries are plotted in the summary graph; you can configure these graphs to

plot daily/monthly summaries instead by picking the relevant option from the **Duration** list in the enlarged mode.

14. To analyze past trends in the loading/unloading of classes, click on the icon at the right, top corner of the **History** tab page.
15. These trend graphs, by default, plot the minimum and maximum values that every measure registered during each hour of the last 24 hours (by default). Using such graphs, you can accurately point to the time windows in which the performance of the Application was at peak, and the times at which there was a lull. By carefully observing these past trends, you can effectively analyze the performance of the application, predict future performances accordingly, and suggest measures to enhance the efficiency. Here again, you can change the timeline of all graphs using the **Timeline** link in Figure 3.73, or just a particular graph by clicking on it and enlarging it.
16. For changing the default duration (of 24 hours) of the trend graphs, do the following:
    - Click on the icon at the top of the **Application Dashboard**.
    - In the **Dashboard Settings** window that appears, select **Trend Graph** from the **Default Timeline for** list.
    - Then, choose a **Timeline** for the graph.
    - Finally, click the **Update** button.

    Note:

    In case of descriptor-based tests, the Summary and Trend graphs displayed in the History tab page typically plot the values for a single descriptor alone. To view the graph for another descriptor, pick a descriptor from the drop-down list made available above the corresponding summary/trend graph.
17. At any point in time, you can switch to the measure graphs by clicking on the button.
18. Typically, the **History** tab page displays measure, summary, and trend graphs for a default set of measures. If you want to add graphs for more measures to this tab page or remove one/more measures for which graphs pre-exist in this tab page, then, do the following:
    - Click the button at the top of the dashboard.
    - The **Dashboard Settings** window then appears. From the **Module** list of Figure 3.57, pick **Application**, choose **CitrixUsers** as the **Sub-System**, and then, select **History Graph** from the **Add/Delete Measures for** list.

- The measures for which graphs pre-exist in the **History** tab page will be automatically displayed in the **Existing Value(s)** list. To delete a measure, and in effect, its corresponding graph as well, select the measure from the **Existing Value(s)** list, click the **Delete** button, and then click the **Update** button.
- To add a new graph, first, pick the **Test** that reports the measure for which a graph is to be generated.
- Next, select the **Measure** of interest.
- Provide a **Display** name for the measure. Then, click the **Add** button to add the measure to the **Existing Values(s)** list. Finally, click the **Update** button.
- This will add a new measure, summary, and trend graph for the chosen measure to the **History** tab page.

3.9.1.6 TerminalServices

To investigate issues relating to the terminal services of the Citrix XenApp application, select **TerminalServices** as the **Subsystem**. Figure 3.75 will then appear.

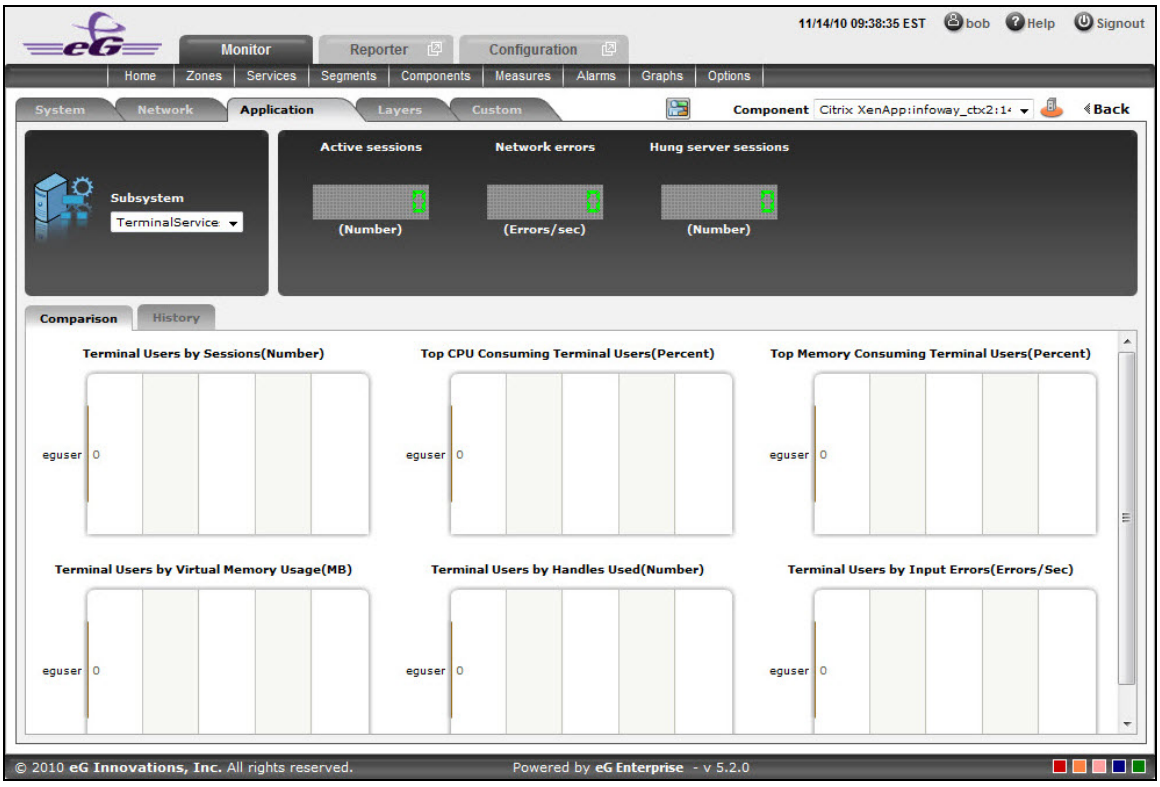


Figure 3.75: The TerminalServices Dashboard

The contents of the TerminalServices dashboard are as follows:

1. The digital graphs section indicates the number of Active sessions, Network errors and Hung server sessions at a single glance. Clicking on a digital graph will lead you to the layer model page of the Citrix XenApp Application; this page will display the exact layer-test combination (see Figure 3.76) that reports the measure represented by the digital graph.

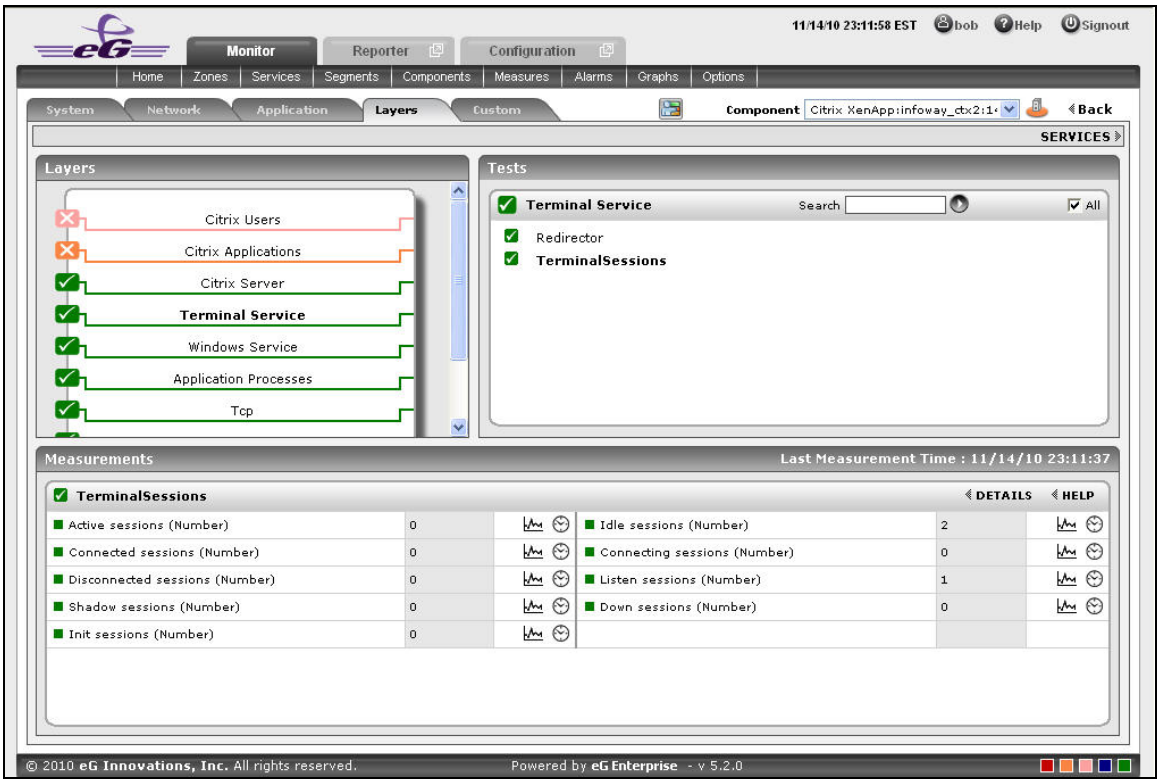


Figure 3.76: The page that appears when the digital graph in the TerminalServices dashboard of the Citrix XenApp Application is clicked

2. The **Comparison** tab page (see 3.9.1) provides a series of graphs for the Terminal Users activity. These graphs provide an insight view of various session related activities that are performed for each Terminal User. This default list of performance areas (i.e., measures) for top-n chart generation can be overridden by following the steps discussed below:

   - Click on the icon at the top of the **Application Dashboard**. In the **Dashboard Settings** window that appears, select **Application** from the **Module** list, and **TerminalServices** from the **Sub-System** list.
   - To add new measures for which top-n graphs are to be displayed in the **Comparison** tab page, first, pick the **Comparison Graph** option from the **Add/Delete Measures for** list.

Upon selection of this option, the pre-configured measures for comparison graphs will appear in the **Existing Value(s)** list.

- Next, select the **Test** that reports the said measure, pick the measure of interest from the **Measures** list, provide a **Display** name for the measure, and click the **Add** button to add the chosen measure to the **Existing Value(s)** list.
- If you want to delete one/more measures for which comparison graphs pre-exist in the **Comparison** tab page, then, as soon as you choose the **Comparison Graph** option from the **Add/Delete Measures for** list, pick any of the displayed measures from the **Existing Value(s)** list, and click the **Delete** button.
- Finally, click the **Update** button to register the changes.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the **Dashboard Settings** window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the  button will not appear.

3. If an application slowdown can be attributed to the lack of adequate resources, then these top-10 bar charts can aid you in swiftly nailing the exact resource location that could be serving as the source of this resource contention.
4. Typically, these bar charts depict the current usage data. Sometimes however, you might want to detect which Application was over-utilizing the resources at some point of time in the past. In such a case, you will have to click on the corresponding graph in the **Comparison** tab page to enlarge it. In the enlarged mode, you can click on the **Compare History** link, so that you can alter the graph **Timeline**, and view which memory pool was the leading memory consumer during the specified timeline.
5. The **History** tab page depicted below, by default, displays time-of-day graphs revealing the user's processes statistics for a default period of 24 hours. If the eG agent reports about a particular session which is down, these graphs will help determine when exactly in the last 24 hours the anomaly has occurred. This default duration of 24 hours can be overridden using the following steps:

- Click on the  icon at the top of the **Application Dashboard**.
- In the **Dashboard Settings** window that appears, select **History Graph** from the **Default Timeline for** list.

- Then, choose a **Timeline** for the graph.
- Finally, click the **Update** button.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the **Dashboard Settings** window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

Figure 3.77: The History tab page of a TerminalServices dashboard

6. A careful study of this graph over time periods longer than 24 hours, can reveal intermittent breaks (if any) in various measures of the user's processes. To ensure that all graphs plot values for longer time periods, click on the **Timeline** link at the right, top corner of the **History** tab page, and then change the timeline using the calendar that pops out. To modify the timeline for a particular graph alone, click on the graph to enlarge it, and alter the timeline in the enlarged mode. Besides the timeline, you can even change the graph dimension (**3D** / **2D**) in the enlarged mode. Figure 1.41 shows an enlarged graph of a measure that is represented in the

History tab page.

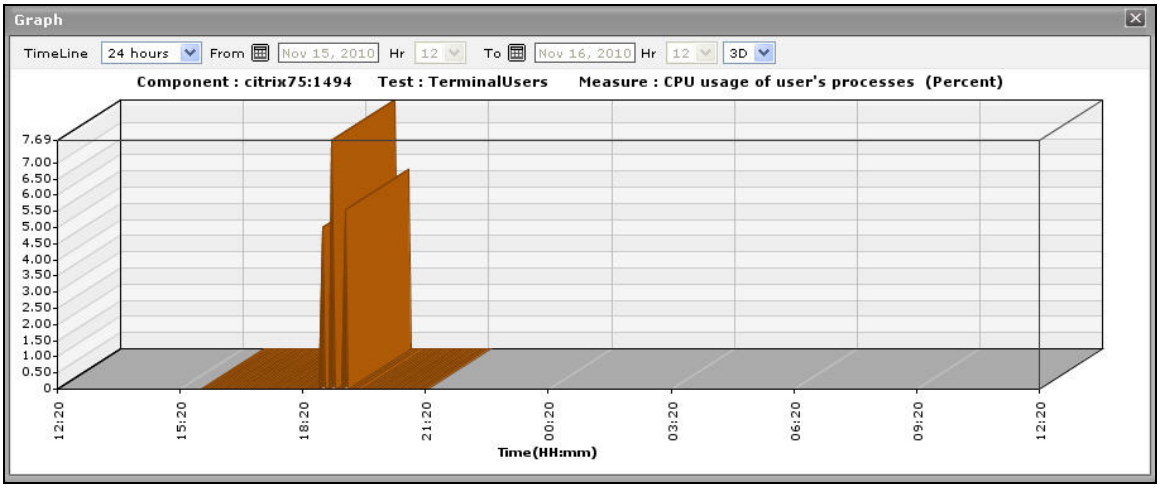


Figure 3.78: The enlarged graph of a measure in the TerminalServices dashboard

7. Sometimes, you might have to periodically determine the percentage of time for which certain critical Citrix XenApp applications have been running, so that you know whether/not the application has been able to maintain the desired service levels. To run such checks, summary graphs of the user's processes measures are useful. To view summary graphs in the **History** tab page, click on the icon at the right, top corner of the **History** tab page.

8. These summary graphs reveal the percentage of time during the last 24 hours (by default) the Citrix XenApp has experienced issues related to the terminal service. To override this default timeline, do the following:

   - Click on the icon at the top of the **Application Dashboard**.
   - In the **Dashboard Settings** window that appears, select **Summary Graph** from the **Default Timeline for** list.
   - Then, choose a **Timeline** for the graph.
   - Finally, click the **Update** button.

9. To perform the summary analysis over a broader time window, click on the **Timeline** link at the right, top corner of the **History** tab page and change the timeline; this will alter the timeline for all the graphs. To change the timeline of a particular graph alone, click on the graph to enlarge it, and then alter its timeline. Also, by default, hourly summaries are plotted in the summary graph; you can configure these graphs to plot daily/monthly summaries instead by picking the relevant option from the **Duration** list in the enlarged mode. Here again, the graph dimension (**3D** / **2D**) can be altered.

10. Similarly, you can analyze the TerminalServices trends by viewing trend graphs in the **History** tab page. For this, click on the icon at the right, top corner of the tab page.

11. These trend graphs, by default, plot the minimum and maximum values registered by every uptime-related measure during every hour for the last 24 hours. Using these graphs, you can ascertain when during the last 24 hours uptime was very high, and when it was low. The default duration of 24 hours can be overridden using the procedure discussed below:

    - Click on the icon at the top of the **Application Dashboard**.
    - In the **Dashboard Settings** window that appears, select **Summary Graph** from the **Default Timeline for** list.
    - Then, choose a **Timeline** for the graph.
    - Finally, click the **Update** button.

12. To perform trend analysis over a longer time span, click on the **Timeline** link at the right, top corner of the **History** tab page and change the timeline; this will alter the timeline for all the graphs. To change the timeline of a particular graph alone, click on the graph to enlarge it, and then alter its timeline. In addition to the timeline, the graph dimension (**3D** / **2D**), the graph **Duration**, and the **Graph type** can also be changed in the enlarged mode. By default, the graph **Duration** is **Hourly**, indicating that trend graphs plot hourly trend values by default. To ensure that these graphs plot the daily/monthly trend values instead, select the relevant option from the **Duration** list. Similarly, as already mentioned, trend graphs plot only the minimum and maximum values registered by a measure during the specified timeline. Accordingly, the **Graph type** is set to **Min/Max** by default in the enlarged mode. If you want the trend graph to plot the average trend values instead, set the **Graph type** to **Avg**. On the other hand, to configure the trend graph to plot the sum of trends set the **Graph type** to **Sum**.

    Note:

    In case of descriptor-based tests, the **Summary** and **Trend** graphs displayed in the **History** tab page typically plot the values for a single descriptor alone. To view the graph for another descriptor, pick a descriptor from the drop-down list made available above the corresponding summary/trend graph.

13. At any point in time, you can switch to the measure graphs by clicking on the button.

14. Typically, the **History** tab page displays measure, summary, and trend graphs for a default set of measures. If you want to add graphs for more measures to this tab page or remove one/more measures for which graphs pre-exist in this tab page, then, do the following:

- Click the button at the top of the dashboard.
- The **Dashboard Settings** window then appears. From the **Module** list of Figure 1.20, pick **Application**, choose **TerminalServices** as the **Sub-System**, and then, select **History Graph** from the **Add/Delete Measures for** list.
- The measures for which graphs pre-exist in the **History** tab page will be automatically displayed in the **Existing Value(s)** list. To delete a measure, and in effect, its corresponding graph as well, select the measure from the **Existing Value(s)** list, click the **Delete** button, and then click the **Update** button.
- To add a new graph, first, pick the **Test** that reports the measure for which a graph is to be generated.
- Next, select the **Measure** of interest.
- Provide a **Display** name for the measure. Then, click the **Add** button to add the measure to the **Existing Values(s)** list. Finally, click the **Update** button.
- This will add a new measure, summary, and trend graph for the chosen measure to the **History** tab page.

Note:

Only users with **Admin** or **Supermonitor** privileges can enable/disable the system, network, and application dashboards, or can customize the contents of such dashboards using the **Dashboard Settings** window. Therefore, whenever a user without **Admin** or **Supermonitor** privileges logs into the monitoring console, the button will not appear.

Chapter 4: Administering the eG Manager to Monitor the Citrix XenApp v7 (or above)

Follow the procedure explained below for managing the Citrix XenApp servers using the eG administrative interface.

1. Log into the eG administrative interface.
2. eG Enterprise is capable of automatically discovering the Citrix XenApp server. If a Citrix XenApp server is already discovered, then directly proceed towards managing it using the **COMPONENTS - MANAGE/UNMANAGE** page (Infrastructure -> Components -> Manage/Unmanage). However, if it is yet to be discovered, then run discovery (Infrastructure -> Components -> Discovery) to get it discovered or add the Citrix XenApp server manually using the **COMPONENTS** page (Infrastructure -> Components -> Add/Modify). Remember that components manually added are managed automatically. Discovered components, however, are managed using the **COMPONENTS - MANAGE/UNMANAGE** page.

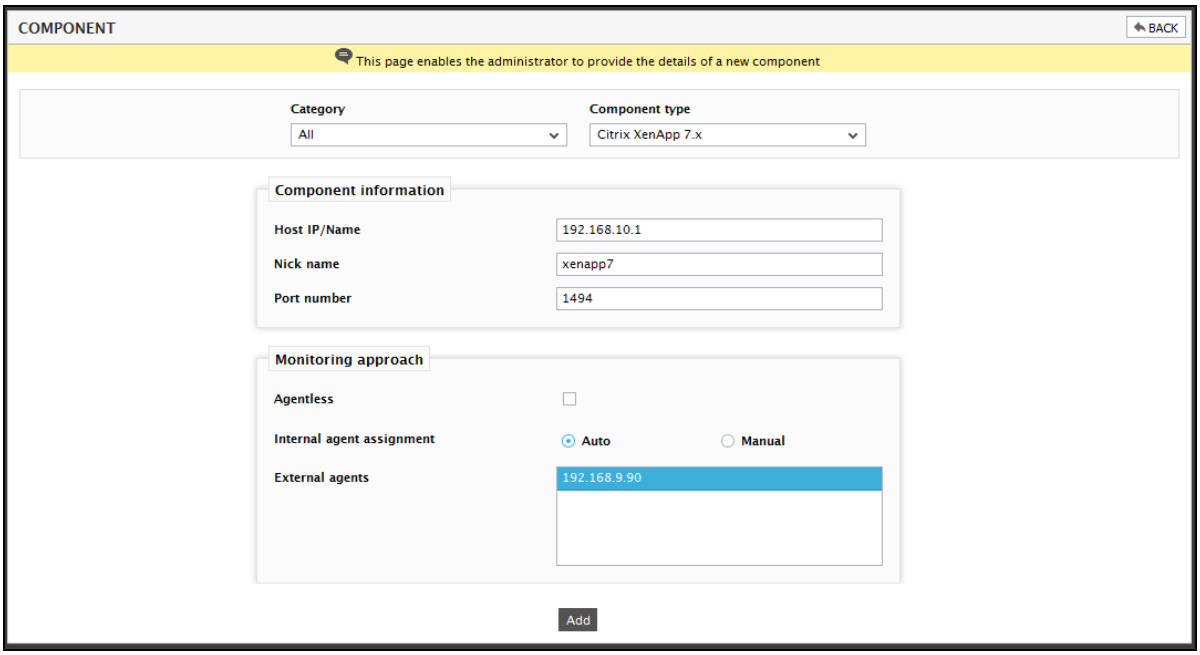


Figure 4.1: Adding a Citrix XenApp server 7.x

2. Specify the **Host IP** and the **Nick name** of the Citrix XenApp server in Figure 4.1. Then click the **Add** button to register the changes.

3. The tests pertaining to the Citrix XenApp 7.x server will be configured automatically.
4. Finally, signout of the eG administrative interface.

Chapter 5: Monitoring Citrix XenApp Servers v7 (and above)

Citrix XenDesktop 7.x represents the merging of the XenApp and XenDesktop technologies into one cohesive package that's built on the same back-end components. Previously, XenApp servers were running on the Citrix Independent Management Architecture. Citrix XenDesktop 7.x however is built on the Citrix FlexCast Management Architecture. This architecture is made up out of Delivery Controllers and Agents. XenDesktop 7.x supports two types of Delivery Agents: one for Windows Server OS machines and one for Windows Desktop OS machines. As shown in the diagram below, both Delivery Agents communicate with the same set of Delivery Controllers and share the common management infrastructure in XenDesktop 7.x. This infrastructure consists of the following core components:

- **Receiver** provides users with self-service access to published resources.
- **StoreFront** authenticates users to site(s) hosting resources and manages stores of desktops and applications that users access.
- **Studio** is a single management console that enables you to configure and manage your deployment. Studio provides various wizards to guide you through the process of setting up an environment, creating workloads to host applications and desktops, and assigning applications and desktops to users.
- **Delivery Controller** distributes applications and desktops, manages user access, and optimizes connections to applications. Each site will have one or more delivery controllers.
- **Server OS Machines** are the “XenApp” replacement – these are VMs or physical machines based on the Windows Server operating system used for delivering applications or hosted shared desktops to users.
- **Desktop OS Machines** are the “XenDesktop” replacement – these are VMs or physical machines based on the Windows Desktop operating system used for delivering personalized desktops to users, or applications from desktop operating systems.

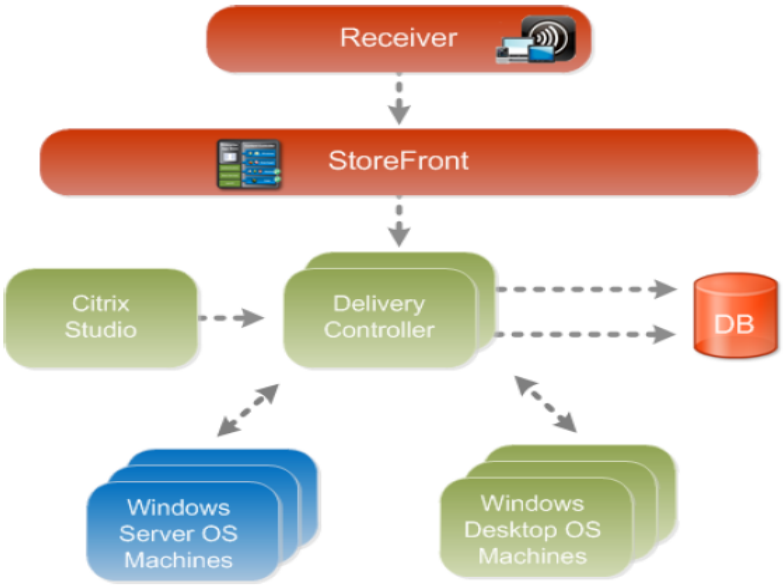


Figure 5.1: The Citrix XenDesktop 7 architecture

Since these components closely co-ordinate with each other to deliver desktops and applications to end-users, a problem with any of these core components – say, the unavailability of StoreFront to authorize user logins, the failure of the broker service, performance bottlenecks with the hypervisor, resource-intensive user sessions to the Server OS machines, snags in the internal operations of the Desktop OS machines – can significantly impact the user experience with Citrix XenDesktop 7.x. Therefore, to ensure a high-quality user experience with the application/desktop delivery service, administrators should closely monitor each component of the XenDesktop 7.x infrastructure, proactively capture performance dips, and accurately isolate where the root-cause of the problem lies – is it with StoreFront? Is it with the delivery controller? Is it with the Server OS machines? Is it with the virtualized platform? Or is it with the Desktop OS machines? This is where eG Enterprise helps!

The eG Enterprise Suite performs **end-to-end monitoring of the Citrix XenDesktop 7.x infrastructure!** Dedicated, web-based monitoring models are offered by eG for each component in the XenDesktop 7.x infrastructure. While the *Citrix StoreFront* model focuses on the health of StoreFront and promptly captures issues in user authentication, the *Citrix XenDesktop Broker 7.x* component monitors the Delivery Controller (or the XenDesktop broker) and reports how well it manages the delivery agents and brokers connections to the Server OS and Desktop OS machines. Moreover the *Citrix XenApp 7.x* model that eG Enterprise provides zooms into the overall performance and problems related to the Server OS machines (that typically run Citrix XenApp 7) and helps isolate pain-points. Also, to monitor the resources allocated to and the resource usage of hypervisors and the Desktop OS machines operating on them, eG Enterprise offers a specialized

monitoring model per hypervisor (such as Citrix XenServer, VMware vSphere, Microsoft Hyper—V, etc.).

Detailed service topology maps in eG represent how these heterogeneous models interact with each other and how dependencies flow.

In the event of a slowdown, eG's patented virtualization-aware root-cause analysis engine analyzes these dependencies, auto-correlates the performance results captured from the different monitoring models in the light of these dependencies, and accurately diagnoses the source of the slowdown. Proactive email/SMS/web-based alerts are then promptly sent out to administrators to alert them to the potential slowdown and what is causing it. This way, eG Enterprise emerges as the ideal solution for monitoring Citrix XenDesktop 7. x

This section deep dives into the *Citrix XenApp 7.x* monitoring model that eG Enterprise offers.

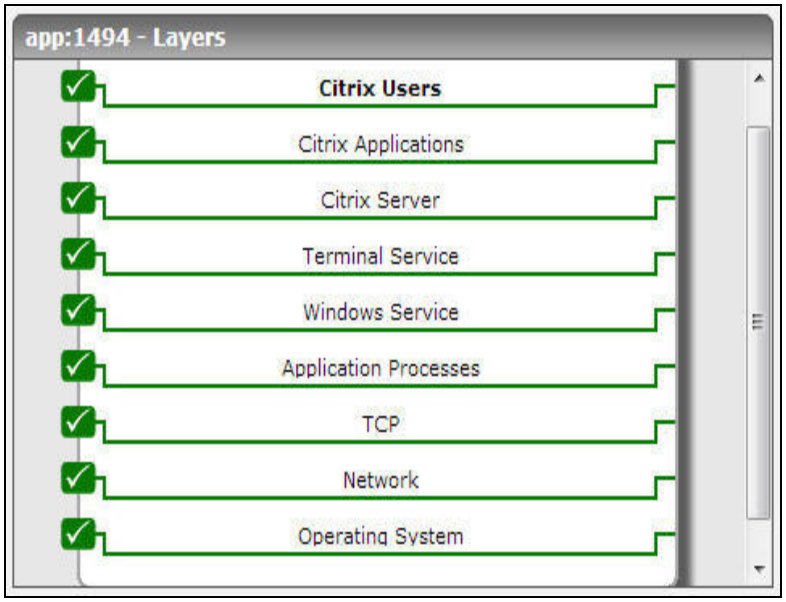


Figure 5.2: The layer model of the Citrix XenApp server 7.x

Each layer of Figure 5.2 above is mapped to a series of tests that periodically check on the availability, responsiveness, and overall performance of the XenApp server, and report a wealth of performance information related to the server. Using the metrics so reported, administrators can find quick and accurate answers to the following performance queries:

Server Monitoring	• Is the Citrix XenApp server available to service user requests? • Are there sporadic disconnects from the Citrix XenApp server? • At what times do peak usage of the servers happen and is the server

	capacity adequate?
User Monitoring	• What is the average response time that critical users are seeing when connecting to Citrix XenApp? • How many users are logged in to each Citrix XenApp in the Citrix farm? • What is the resource usage (CPU and memory) for each user?
Operating System Monitoring	• What is the average CPU and memory usage on all the servers in the farm? • Is any unusual memory scanning/paging activity happening on the systems? • Are the critical Citrix XenApp server processes processes up? What is their resource consumption?
Published Applications Monitoring	• What are the published applications on the server? • Who is using each application? • What is the resource usage for each published application?

The **Operating System**, **Network**, **TCP** and **Windows Service** layers of the *Citrix XenApp* are similar to that of a *Windows* server model. Since these tests have been dealt with in the *Monitoring Unix and Windows Servers* document, the following section focuses on the **Application Processes** layer.

5.1 The Application Processes Layer

This layer tracks the TCP ports and reports the availability and responsiveness of each port. Besides, this layer depicts the states of the different processes that must be executing for the application service to be available. Since the Processes and Windows Processes tests mapped to this layer are detailed in the *Monitoring Unix and Windows* document, let us now discuss the Port Checks test in detail.

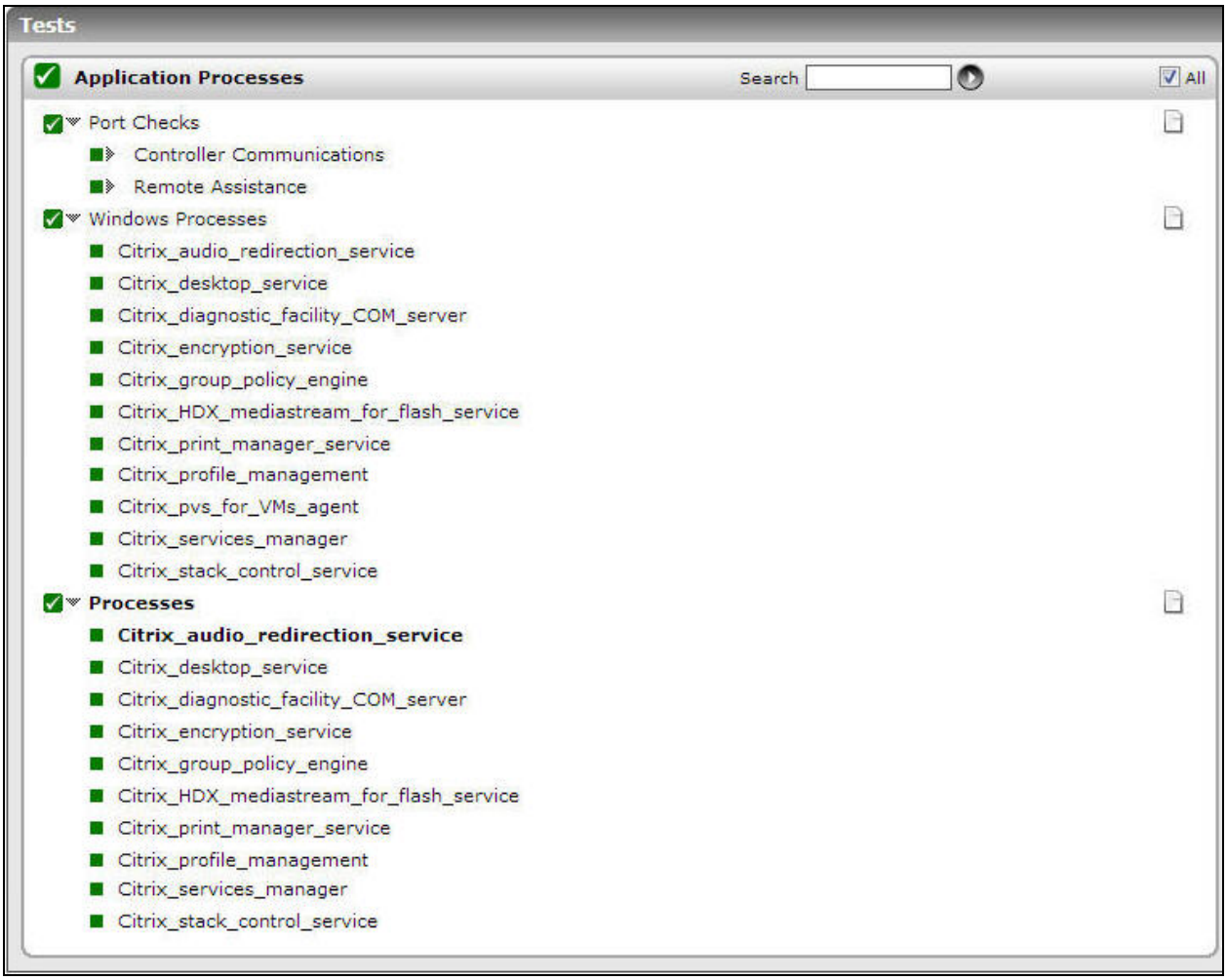


Figure 5.3: The tests mapped to the Application Processes layer

5.1.1 Port Checks Test

This test primarily checks whether the critical TCP ports on the Citrix XenApp server are up/down, and reports the responsiveness of each configured port to client requests. However, these checks might not be adequate at all times; you could have a case where the Citrix XenApp server port is up but the server is still not responding. When a connection is made to the Citrix XenApp server, it will typically send a message "ICA" to the client. This check connects to the port and then validates the response from the Citrix XenApp server to see if the ICA stream is being received by the client. Hence, this test additionally reports the ICA connection availability.

Target of the test : A Citrix XenApp server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each port that is to be 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 specified HOST listens to. By default, this is 1494.
4. **TARGETPORTS** – Specify either a comma-separated list of port numbers that are to be tested (eg., 1494,1495,1496), or a comma-separated list of *port name:port number* pairs that are to be tested (eg., ica:1494,smtp:25,mssql:1433). In the latter case, the port name will be displayed in the monitor interface. Alternatively, this parameter can take a comma-separated list of *port name:IP address:port number* pairs that are to be tested, so as to enable the test to try and connect to Tcp ports on multiple IP addresses. For example, *mysql:192.168.0.102:1433*,*egwebsite:209.15.165.127:80*.
5. **TIMEOUT** - Here, specify the maximum duration (in seconds) for which the test will wait for a response from the server. The default **TIMEOUT** period is 60 seconds.
6. **ISPASSIVE** - If the value chosen is **YES**, then the server under consideration is a passive server in a cluster. No alerts will be generated if the server is not running. Measures will be reported as "Not applicable" by the agent if the server is not up.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
TCP connection availability:	Indicates whether the TCP connection is available or not.	Percent	An availability problem can be caused by different factors – e.g., the server process may not be up, a network problem may exist, or there could be a configuration problem with the DNS server.
Response time:	Indicates the time taken by the server to respond to a request.	Secs	An increase in response time can be caused by several factors such as a server bottleneck, a configuration problem with the DNS server, a network problem, etc.
ICA connection availability:	Indicates whether ICA connection is available	Percent	While the value 100 for this measure indicates that the ICA

Measurement	Description	Measurement Unit	Interpretation
	or not.		stream is being received by the client, the value 0 indicates that it is not.

5.2 The Remote Desktop Services Layer

In most environments, the Citrix XenApp 7 (or above) server functions in conjunction with a Terminal server. To enable the administrators of XenDesktop 7 environment to monitor the movement and resource usage of the Terminal clients on the Citrix XenApp server, the eG Enterprise system has introduced the **Remote Desktop Services** layer. The tests mapped to this layer are the same as those mapped to the **Remote Desktop Services** layer of a Microsoft RDS server. These tests hence, have already been dealt with elaborately in the *Monitoring Microsoft RDS Servers* document. So, let us proceed to look at the The Citrix Server Layer layer.

5.3 The Citrix Server Layer

Citrix XenApp server-related performance parameters are monitored by the tests mapped to the **Citrix Server** layer. This includes:

- Profile size
- User login and profile loading process
- User profile management

Since there tests are already discussed in the , let us now proceed to discuss the **Citrix Applications** layer.

5.3.1 Citrix MCS Storage Driver Test

Machine Creation Services (MCS) Storage Optimization (MCSIO), is a new feature within MCS provisioning and was introduced in XenApp and XenDesktop 7.9.

MCSIO reduces I/O load through a two-tier caching system. An in-memory cache, known as the “temporary memory cache,” is used as the first storage tier. If the in-memory cache fills up, subsequent writes will be cached using an additional disk attached to the provisioned machine as the second tier - this is known as the “temporary disk cache.” To achieve this, MCSIO provisioned machines have an additional MCSIO driver to intercept and manage IO operations.

For improved I/O performance, both the storage tiers should be adequately sized, so that the likelihood of writes directly reaching the system disk reduces considerably. If the caches are not sized right, then they may soon run out of space for writes, causing the driver to direct writes to the system disk. This in turn will reduce cache hits, increase direct disk accesses, and thus, significantly degrade I/O performance. To avoid this, administrators should continuously monitor the I/O load on the MCSIO driver, understand how the driver uses the in-memory and disk cache for managing these I/O operations, and make sure that the caches are sized right to support these operations. This is where the Citrix MCS Storage Driver test helps!

This test tracks the I/O requests to the driver and reports the rate at which the driver reads from or writes into each of the caches and he system disks in order to process these requests. This way, the test reveals whether/not the caches are doing a good job of preventing direct disk accesses. Additionally, the test also closely monitors how the memory in the in-memory cache and the disk space in the cached disk is utilized, and proactively alerts administrators to any potential resource crunch in the caches. This way, the test provides useful sizing pointers to administrators.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 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 used by the Citrix server

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Cache memory data reads:	Indicates the rate at which data is read from the in-memory cache.	KB/Sec	
Cache memory data writes:	Indicates the rate at which data is written to the in-memory cache.	KB/Sec	A steady drop in the value of this measure could indicate a writing bottleneck. One of the reasons for this could be the lack of enough

Measurement	Description	Measurement Unit	Interpretation
			memory. Check the value of the Cache memory utilization measure to determine whether/not there is a memory crunch.
Cache memory data reads and writes:	Indicates the rate at which data is written to and read from the in-memory cache.	KB/Sec	A steady drop in the value of this measure is indicative of an I/O bottleneck.
Cache memory reads:	Indicates the rate at which read operations are performed on the in-memory cache.	Reads/Sec	
Cache memory writes:	Indicates the rate at which write operations are performed on the in-memory cache.	Writes/Sec	
Cache memory IOPS:	Indicates the rate at which read and write operations are performed on the in-memory cache.	Operations/Sec	
Cache memory target size:	Indicates the amount of memory that the MCS storage driver will aim to use.	MB	
Cache memory used:	Indicates the amount of memory that the driver has actually used.	MB	
Cache memory free:	Indicates the amount of memory that is unused.	MB	
Cache memory utilization:	Indicates the percentage of memory	Percent	A value close to 100% is a cause for concern as it indicates that the

Measurement	Description	Measurement Unit	Interpretation
	that the driver has utilized.		in- memory cache is rapidly running out of memory. You may want to allocate more RAM to the cache to make sure that the writes do not spill over to the cache disk.
Cache disk data reads:	Indicates the rate at which data is read from the cache disk.	KB/Sec	
Cache disk data writes:	Indicates the rate at which data is written to the cache disk.	KB/Sec	A steady drop in the value of this measure could indicate a writing bottleneck. One of the reasons for this could be the lack of enough disk space in the cache disk. Check the value of the Cache disk utilization measure to determine whether/not there is a space crunch.
Cache disk data reads and writes:	Indicates the rate at which data is written to and read from the cache disk.	KB/Sec	A steady drop in the value of this measure is indicative of an I/O bottleneck.
Cache disk reads:	Indicates the rate at which read operations are performed on the cache disk.	Reads/Sec	
Cache disk writes:	Indicates the rate at which write operations are performed on the cache disk.	Writes/Sec	
Cache disk IOPS:	Indicates the rate at which read and write operations are performed on the cache disk.	Operations/Sec	

Measurement	Description	Measurement Unit	Interpretation
Cache disk size:	Indicates the current size of the cache disk.	MB	
Cache disk used:	Indicates the amount of space that the driver has used in the cache disk.	MB	
Cache disk free:	Indicates the amount of space that is unused in the cache disk.	MB	
Cache disk utilization:	Indicates the percentage of space used in the cache disk.	Percent	A value close to 100% is a cause for concern as it indicates that the cache disk is running out of disk space. You may want to expand the capacity of the cache disk to make sure that the writes do not spill over to the system disks.
Storage driver read requests:	Indicates the number of read requests that were received by the MCS storage driver since boot.	Number	
Storage driver write requests:	Indicates the number of write requests that were received by the MCS storage driver since boot.	Number	
Storage driver read and write requests:	Indicates the number of read and write requests that were received by the MCS storage driver since boot.	Number	This is a good indicator of the I/O load on the driver.
System disk data reads:	Indicates the rate at which data is read from the system disks.	KB/Sec	

Measurement	Description	Measurement Unit	Interpretation
System disk data writes:	Indicates the rate at which data is written to the system disks.	KB/Sec	
System disk data reads and writes:	Indicates the rate at which data is written to and read from the system disks.	KB/Sec	
System disk reads:	Indicates the rate at which reads are performed from the system disks.	Reads/Sec	
System disk writes:	Indicates the rate at which writes are performed into the system disks.	Writes/Sec	
System disk IOPS:	Indicates the rate at which I/O operations are performed on system disks.	Operations/Sec	A zero value is desired for this measure.

5.3.2 Citrix Universal Printing Load Balancer Performance Test

This test reports statistics pertaining to the different applications executing on a Citrix XenApp server and their usage by Citrix clients.

Note:

This test will report metrics only if the XenApp server being monitored uses the .Net framework v3.0 (or above).

Target of the test : Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each application that is monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** – The host for which the test is to be configured.
3. **PORT** – The port number at which the specified **HOST** listens to. By default, this is 1494.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Active printer connections	Indicates the number of printer connections that were active on this Citrix Universal Print server instance.	Number	This measure is a good indicator of the load to the printers in the Citrix environment.
Printer connections created	Indicates the number of printer connections that were created on this Citrix Universal Print server instance.	Number	
Printer connections deleted	Indicates the number of printer connections that were deleted on this Citrix Universal Print server instance.	Number	

5.3.3 Citrix Server Input Delay Test

Poor application performance is one of the most difficult problems to diagnose by the administrators. Traditionally, diagnosis was done by collecting CPU, memory, disk I/O and a few other metrics. The data collected from traditional metrics were not sufficient to figure out the root cause of poor performance of the applications since the variations measured by the metrics were large. In virtual environments where multiple users accessed an application from remote at the same time, users faced difficulties in accessing the application whenever there was an increase in the count of users. The more the users are accessing the application, the higher was the CPU usage of the systems in the environment and the higher was the user input delays i.e., the users were forced to wait for a longer duration to interact with the application. The user input delay is measured by how long any user input (such as mouse or keyboard usage) stays in the queue before it is picked up by a process.

To aid in figuring out the real reason behind such poor show put up by the applications from the perspective of user experience, it is necessary to measure the user input flows or rather user input delays while the applications were accessed.

This test captures such user input delays at the Citrix XenApp server level and reports the same to the administrators. With the help of this test, administrators can determine the maximum and average time taken by the applications to respond to the user input across all XenApp sessions, and thus figure out if there is any time delay in responses from applications.

Note:

This test will report measures only on Windows 2019 (and above).

Target of the test : A Citrix XenApp 7.x Server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the target Citrix XenApp 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 used by the Citrix XenApp server. By default, this is set to 1494.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Maximum input delay for sessions	Indicates the maximum amount of time lag detected between the user's input through any input device (e.g., mouse, keyboard) and the time at which the application responds to the input.	Seconds	Ideally, the values of these measures should be 0 or very low. To know exactly which user/application experiences the maximum input lag, you can refer to the **Citrix Users in Sessions** and **Citrix Applications** tests.
Average input delay for sessions	Indicates the average amount of time lag detected between the user's input through any	Seconds	

Measurement	Description	Measurement Unit	Interpretation
	input device (e.g., mouse, keyboard) and the time at which the application detected the input.		

5.3.4 Citrix Session Recording Agent Test

Citrix Session Recording Agent is a Windows service installed on a Citrix XenApp server. The recording agent is responsible for capturing and recording the on-screen activities and transferring the content to the Session Recording Server using the Microsoft Message Queuing (MSMQ) service. Administrators may wish to know the workload handled by the recording agent, so that they can ensure that the server is sized with adequate processing power for handling the load. The **Citrix Session Recording Agent** test helps administrators determine the workload of the Session Recording Agent.

Target of the test : A Citrix XenApp server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for Citrix XenApp server being monitored

Configuration 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 Citrix XenApp server.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Active session recording count	Indicates the number of sessions that are currently being recorded by the recording agent on the XenApp Server OS VDA machine.	Number	This measures is a good indicator of current workload of the recording agent.

Measurement	Description	Measurement Unit	Interpretation
Data read from session recording driver	Indicates the rate at which the Session Recording Agent reads data (in KB) from kernel components responsible for acquiring session data.	KB/sec	

5.4 The Citrix Applications Layer

Using the tests mapped to this layer, the resource usage per application executing on the Citrix XenApp server can be measured.

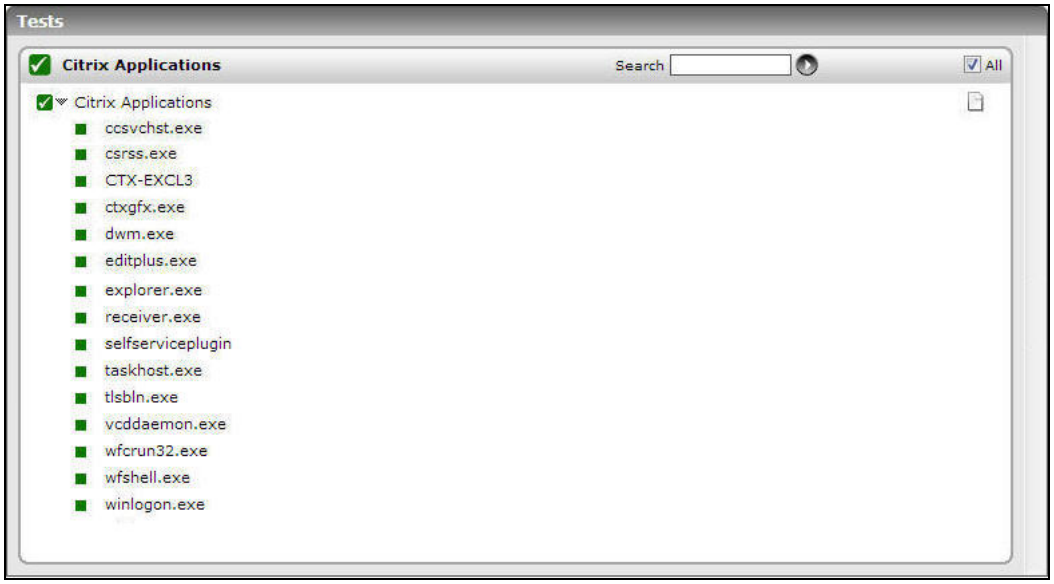


Figure 5.4: Tests associated with the Citrix Applications layer

5.4.1 Citrix Applications Test

This test reports statistics pertaining to the different applications executing on a Citrix XenApp server and their usage by Citrix clients.

Note:

This test will report metrics only if the XenApp server being monitored uses the .Net framework v3.5 (or above).

Target of the test : Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each application that is 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 number at which the specified **HOST** listens to. By default, this is 1494.
4. **SHOW PUBLISHED APPS** – Using this flag, you can indicate whether the test should monitor published applications alone or all applications running on the server. By default, this flag is set to **No**, indicating that all applications will be monitored by default. To monitor only published applications, you need to set this flag to **Yes**. **However, prior to changing the flag status to 'Yes', you need to make sure that a 'Citrix XenDesktop Broker' componexnt is also managed by the eG Enterprise system and is reporting metrics**.
5. **SHOW PUBLISHED DESKTOPS** – By default, this flag is set to **No**. If this flag is set to **Yes**, then the detailed diagnosis of this test will list the resource-intensive processes/applications accessed by a user along with the exact published desktop that has been used by the user to access the application. **Note that, in the detailed diagnosis, the 'host name' of the monitored server will be displayed as the 'published desktop name'**.
6. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who accessed an application on the server. This way, administrators will be able to quickly determine which user logged into the server from which domain. If you want the detailed diagnosis to display only the username of these users, set this flag to **No**.
7. **ENABLE BROWSER MONITORING** - By default, this flag is set to **No**, indicating that the eG agent does not monitor browser activity on the XenApp server. If this flag is set to **Yes**, then, whenever one/more IE (Internet Explorer) browser instances on the XenApp server are accessed, the detailed diagnosis of the *Instances currently running* measure will additionally reveal the URL being accessed via each IE instance and the resources consumed by every URL. Armed with this information, administrators can identify the web sites that are responsible for excessive resource usage by an IE instance.
8. **COMBINED IE REPORT** - **This flag is applicable only if the SHOW PUBLISHED APPS flag is set to Yes**.

   Typically, the detailed diagnosis of this test lists the applications that a user accessed on the XenApp server and the resource usage of each application. To fetch these details, the

eG agent takes the published name (eg., Word) of each application that is open on the XenApp server, determines the underlying process and process arguments that drive the application (eg., winword.exe is the process that drives Word), and tries to find an exact match for this process name and arguments in the task manager. If an exact match is found, then, by default, the underlying process name of the application is displayed in the detailed diagnosis. The resource usage metrics that correspond to that process name in the task manager are also displayed as part of detailed diagnosis.

For published applications that open in the Internet Explorer (IE) browser on XenApp, this process name matching algorithm may not work. This is because, if multiple applications on XenApp are opened using IE, each such application will open only in a different tab page of the IE browser. As a result, though the underlying process names will be different for each of these applications, in the task manager, the process names for all these applications will only be 'iexplore.exe'. Because of the name mismatch (between XenApp and the task manager), the eG agent will wrongly determine that no instance of an application is running, and will exclude that application name from the detailed diagnostics. To avoid this, with the **SHOW PUBLISHED APPS** flag set to **Yes**, set this flag to **Yes**.

If this is done, then, the eG agent will be able to capture every application or application instance that is opened in a different tab page of an IE browser, despite the process name mismatch. Also, the eG agent will be able to collect detailed metrics of such applications and display them in the Detailed Diagnosis page against the process name 'Internet Explorer'.

9. **EXCLUDE** - By default, this parameter is set to *none*. This means that the test will monitor all the applications that are launched on the XenApp server, by default. If you want the test to disregard certain applications when monitoring, then provide a comma-separated list of process names that correspond to the applications you want to ignore, in the **EXCLUDE** text box. For instance, your specification can be: winword. *exe,js.exe,taskmgr.exe* . Your specification can include wild card patterns as well. For example: **win*,js*,*task*

10. **SHOW ALL DESKTOP PROCESSES** - Using this flag, you can indicate whether the test should report top resource-intensive processes alone or all processes running in the background when the user accesses an application. By default, this flag is set to **No**, indicating that this test will report only top three resource-intensive processes e.g. CPU, Memory and IO Reads processes from the desktop OS processes. This helps the administrator optimize the database. To report all the processes, you need to set this flag to **Yes**.

11. **SHOW ONLY ACTIVE APPS** – Using this flag, you can indicate whether the test should monitor all applications or applications that are currently active on the server. By default, this flag is set to **Yes**, indicating that only the currently active applications will be monitored by the eG agent. To

monitor all applications, you need to set this 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 made by the test

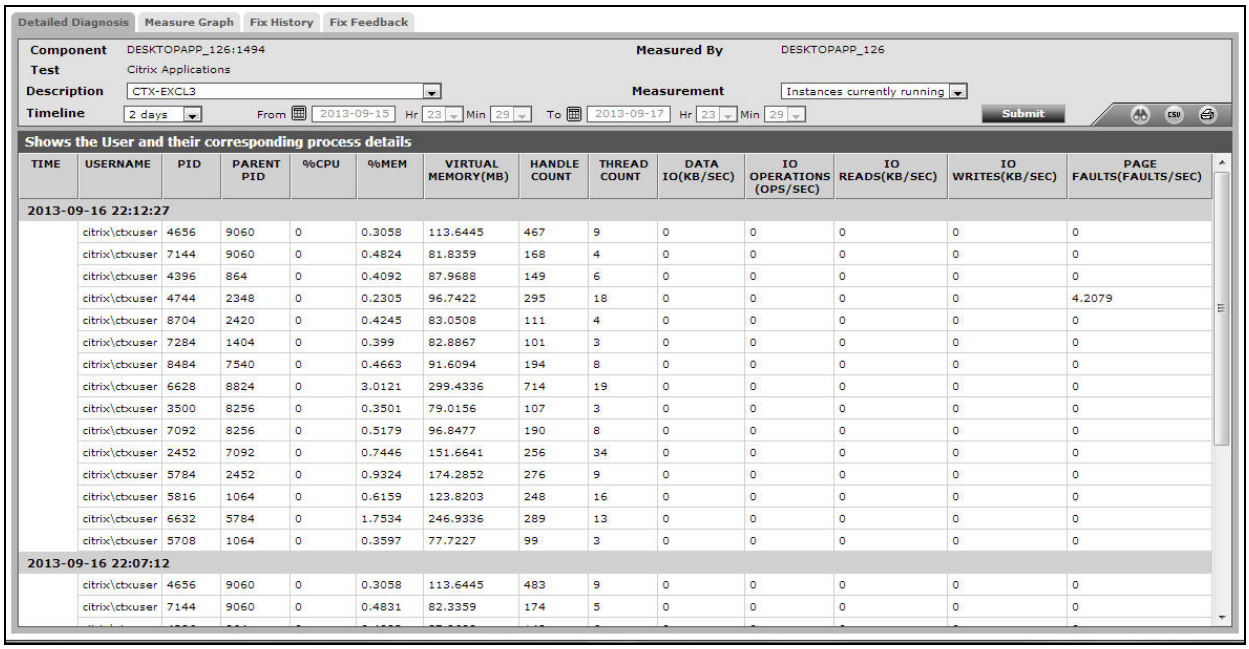
Measurement	Description	Measurement Unit	Interpretation
Instances currently running:	Number of instances of the published application currently executing on this Citrix XenApp server.	Number	This value indicates if too many or too few instances corresponding to an application are executing on the host. Use the Detailed diagnosis of this measure to identify all the users executing this application and comparing the users will help you to identify which user is utilizing the maximum memory, CPU etc
CPU usage:	Indicates the percentage of CPU used by the published application.	Percent	A very high value could indicate that the specified application is consuming excessive CPU resources.
Memory usage:	This value represents the ratio of the resident set size of the memory	Percent	A sudden increase in memory utilization for an application may be indicative of memory leaks in the application.

Measurement	Description	Measurement Unit	Interpretation
	utilized by the application to the physical memory of the host system, expressed as a percentage.		
Handle count:	Indicates the number of handles opened by this application.	Number	An increasing trend in this measure is indicative of a memory leak in the application.
Number of threads:	Indicates the number of threads that are used by the application.	Number	
I/O data rate:	Indicates the rate at which this application is reading and writing bytes in I/O operations.	KBytes/Sec	This value counts all I/O activity generated by each instance of the application and includes file, network and device I/Os.
I/O data operations:	Indicates the rate at which this application is issuing read and write data to file, network and device I/O operations.	Operations/Sec	
I/O read data rate:	Indicates the rate at which this application is reading data from file, network and device I/O operations.	KBytes/Sec	
I/O write data rate:	Indicates the rate at which this application is writing data to file, network and device I/O operations.	KBytes/Sec	
Page fault rate:	Indicates the total rate at which page faults are occurring for the threads of all matching applications.	Faults/Sec	This measure is a good indicator of the load on the application. A page fault occurs when a thread refers to a virtual memory page that is not in its working set in main memory. This may not cause the page to be

Measurement	Description	Measurement Unit	Interpretation
			fetched from disk if it is on the standby list and hence already in main memory, or if it is in use by another application with whom the page is shared.
Working set memory	Indicates the current size of the working set of this application.	MB	The Working Set is the set of memory pages touched recently by the threads in a process/application. If free memory in the server is above a threshold, pages are left in the Working Set of an application even if they are not in use. When free memory falls below a threshold, pages are trimmed from Working Sets. If they are needed they will then be soft-faulted back into the Working Set before leaving main memory. Comparing the working set across applications indicates which application is taking up excessive memory.
Input delay for processes - max	Indicates the maximum amount of time lag detected between the user's input through any input device (e.g., mouse, keyboard) and the time at which this application detected the input.	Seconds	Poor application performance is one of the most difficult problems to diagnose by the administrators. Traditionally, diagnosis was done by collecting CPU, memory, disk I/O and a few other metrics. The data collected from traditional metrics were not sufficient to figure out the root cause of poor performance of the applications since the variations measured by the metrics were large. In virtual environments where multiple users accessed an application from remote at the same time, users faced difficulties in accessing the application whenever there was an increase in the count of

Measurement	Description	Measurement Unit	Interpretation
			users. The more the users are accessing the application, the higher was the CPU usage of the systems in the environment and the higher was the user input delays i.e., the users were forced to wait for a longer duration to interact with the application. The user input delay is measured by how long any user input (such as mouse or keyboard usage) stays in the queue before it is picked up by a process. These two measures capture such user input delays at the application/process level. These insights enable administrators to accurately identify which application/process is responding slowly to user requests. These measures will be reported only on Windows 2019 (and above). Ideally, the values of these measures should be 0 or very low.
Input delay for processes - avg	Indicates the average amount of time lag detected between the user's input through any input device (e.g., mouse, keyboard) and the time at which this application detected the input.	Seconds	

The detailed diagnosis of the *Instances currently running* measure, if enabled, lists the user sessions that are currently open, the process ids of the processes being executed by each of the users, and the CPU and memory utilization (in %) of each of these processes. Additionally, this detailed diagnosis helps you in identifying the handles that are opened, the thread count, the read/write operations as well as the I/O operations for each application. This information enables the Citrix administrator to identify the processes with a high CPU/memory utilization. In the event of a server overload, the Citrix administrator might decide to terminate these processes (see Figure 5.5).



TIME	USERNAME	PID	PARENT PID	%CPU	%MEM	VIRTUAL MEMORY(MB)	HANDLE COUNT	THREAD COUNT	DATA IO(KB/SEC)	IO OPERATIONS (OPS/SEC)	IO READS(KB/SEC)	IO WRITES(KB/SEC)	PAGE FAULTS(FAULTS/SEC)
2013-09-16 22:12:27													
	citrix\ctxuser	4656	9060	0	0.3058	113.6445	467	9	0	0	0	0	0
	citrix\ctxuser	7144	9060	0	0.4824	81.8359	168	4	0	0	0	0	0
	citrix\ctxuser	4396	864	0	0.4092	87.9688	149	6	0	0	0	0	0
	citrix\ctxuser	4744	2348	0	0.2305	96.7422	295	18	0	0	0	0	4.2079
	citrix\ctxuser	8704	2420	0	0.4245	83.0508	111	4	0	0	0	0	0
	citrix\ctxuser	7284	1404	0	0.399	82.8867	101	3	0	0	0	0	0
	citrix\ctxuser	8484	7540	0	0.4663	91.6094	194	8	0	0	0	0	0
	citrix\ctxuser	6628	8824	0	3.0121	299.4336	714	19	0	0	0	0	0
	citrix\ctxuser	3500	8256	0	0.3501	79.0156	107	3	0	0	0	0	0
	citrix\ctxuser	7092	8256	0	0.5179	96.8477	190	8	0	0	0	0	0
	citrix\ctxuser	2452	7092	0	0.7446	151.6641	256	34	0	0	0	0	0
	citrix\ctxuser	5784	2452	0	0.9324	174.2852	276	9	0	0	0	0	0
	citrix\ctxuser	5816	1064	0	0.6159	123.8203	248	16	0	0	0	0	0
	citrix\ctxuser	6632	5784	0	1.7534	246.9336	289	13	0	0	0	0	0
	citrix\ctxuser	5708	1064	0	0.3597	77.7227	99	3	0	0	0	0	0
2013-09-16 22:07:12													
	citrix\ctxuser	4656	9060	0	0.3058	113.6445	483	9	0	0	0	0	0
	citrix\ctxuser	7144	9060	0	0.4831	82.3359	174	5	0	0	0	0	0

Figure 5.5: The detailed diagnosis for the Instances currently running measure

5.4.2 Citrix Application Launches Test

To know which published applications on the XeAnApp server are currently accessed by users and how many instances of each application have been launched presently, use the **Citrix Application Launches** test. Detailed diagnostics, if enabled, reveal the users accessing the published applications and the thin clients from which the users are connecting to the XenApp server.

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 : Citrix XenApp 7

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every 'published application' on the XenApp server 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** - Refers to the port used by the Citrix server
4. **REPORT BY DOMAIN NAME** - By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who logged into the Citrix server. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.
5. **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.
6. **DETAILED DIAGNOSIS** -To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

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

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

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New application launches:	Represents the number of instances of this published application that were launched since the last measurement period.	Number	Use the detailed diagnosis of this measure to know which users are currently accessing the application and the clients from which the users are connecting.

5.4.3 Application Launches Test

When a user complains that it is taking too long to launch applications on Citrix, 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 Launches** test provides these valuable insights to the administrators. This test auto-discovers all the applications that are currently launched on the Citrix server, 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, select the **Enable/Disable** option from the **Tests** menu of the **Agents** tile, select **Component type** as Citrix XenApp, pick this test from the **DISABLED TESTS** list, click the **<** button, and click **Update** to save the changes.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every application (published and non-published) on the XenApp server 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** - Refers to the port used by the Citrix server
4. **REPORT BY DOMAIN NAME** - By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who logged into the Citrix server. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.
5. **SERVER VERSION** - By default, this parameter is set to XA6 for this test. **Do not change this default setting**.
6. **EXCLUDE** – By default, this parameter is set to *none*. This means that the test will monitor all

the applications that are launched on the XenApp server, by default. If you want the test to disregard certain applications when monitoring, then provide a comma-separated list of process names that correspond to the applications you want to ignore, in the **EXCLUDE** text box. For instance, your specification can be: *winword.exe,js.exe,taskmgr.exe* . Your specification can include wild card patterns as well. For example: **win*,js*,*task*

7. **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.
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 made by the test

Measurement	Description	Measurement Unit	Interpretation
Application launches:	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.
Launch duration :	Indicates the average time taken by this application to launch.	Secs	Compare the value of this measure across applications to know which application took the longest time to launch. User experience with this application will naturally be poor.

Measurement	Description	Measurement Unit	Interpretation
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's experience with this delay in launching, use the detailed diagnosis of the Application launches measure.

5.4.4 Application File Status Test

This test reports whether configured files are available or not, and if available, reports the size of the individual files.

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 : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every configured file path

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured
3. **FILENAME** - Provide a comma-separated list of the full path of the files that are to be monitored. For instance, on a Unix host, your specification can be: *opt/usr/alert.log,opt/tmp/error.log*. On a Windows host, your specification can be: *C:\eGurkha\agent\logs\agentout.log,C:\eGurkha\agent\logs\agenterr.log*.

   Also, if you want to monitor files with names that include a date, then your **FILENAME** specification should indicate the date format used for naming the files. For instance, to monitor all files that are named error, but which end with dates that are of the format DDMMYY, your

FILENAME specification should be: *C:\logs\errorDDMMYY*. As per this specification, files such as *error21082015, error22082015,* and *error24082015* will be monitored.

Your **FILENAME** specification can include file names with dates and without dates – for eg., *C:\eGurkha\agent\logs\agentout.log,C:\eGurkha\agent\logs\agenterr.log, C:\logs\errorDDMMYY,C:\errorlogs\MMDDYYYY_error*

If you wish to monitor the latest file in a folder that consists of too many files with the same extension, say for example *.log*, then your **FILENAME** specification should be : *C:\Temp*.log.*

Note:

Wildcard characters are not supported while entering the full path of the files in the **FILENAME** text box. So, provide the exact path of the files in the same.

4. **DATE PATTERN** – Using this test, you can also monitor all files with names that include a date. If your **FILENAME** specification above includes files with dates, then set the **DATE PATTERN** flag to **Yes**. If this is done, then the test will look for date patterns in your **FILENAME** specification. If your **FILENAME** specification does not include data patterns, set this flag to **No**. If this is done, then the test will not look for date patterns in your **FILENAME** specification. Say that you include a file name that embeds a date pattern in your **FILENAME** specification – eg., *C:\logs\errorMMDDYYYY* - and set the **DATE PATTERN** flag to **No**. In this case, the test will disregard the date pattern *MMDDYYYY*, and will instead search for a file with the name, *errorMMDDYYYY*.
5. **MAX AGE IN HOURS** - Specify the time duration in hours beyond which this test should report whether/not the file configured against the **FILENAME** is updated. By default, this parameter is set to *none*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
File availability	Indicates whether this file is currently available or not.	Percent	This measure reports the value 100, if the file is available in the configured path. If the files is not available, a value of 0 is reported.
File size	Indicates the current size of this file.	MB	This measure reports the size of a file only if the File availability measure returns a value of 100 for that file - i.e., only when the file is available.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>File growth during the last measurement period</td><td>Indicates the increase in the size of this file since the last measurement period.</td><td>KB</td><td>A consistent increase in the value of this measure indicates that the file size is increasing steadily.</td></tr>
<tr><td>Is the file modified?</td><td>Indicates whether/not this file was modified.</td><td></td><td>If the size of the file increased from the last measurement period, then this measure will report the value Yes. If there is no change in file size since the last measurement period, then this measure will report the value No.

The numeric values that correspond to these measure values are as follows:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Yes</td><td>100</td></tr><tr><td>No</td><td>0</td></tr></table>
Note:

By default, this test reports the Measure Values listed in the table above to indicate whether/not a file has grown. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Is the file not updated above configured hours?</td><td>Indicates whether this file is not updated since the time duration specified against the MAX AGE IN HOURS parameter.</td><td></td><td>If this file is not updated beyond the specified time duration, then this measure will report a value Yes. Otherwise, this measure will report the value No.

The numeric values that correspond to these measure values are as follows:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Yes</td><td>1</td></tr><tr><td>No</td><td>0</td></tr></table></td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			Note: By default, this test reports the **Measure Value**s listed in the table above to indicate whether his file was not updated beyond the specified time duration. In the graph of this measure however, the same is represented using the numeric equivalents only.

5.5 The Citrix Users layer

To accurately assess the individual user experience on the Citrix XenApp server, use the tests mapped to the **Citrix Users** layer.

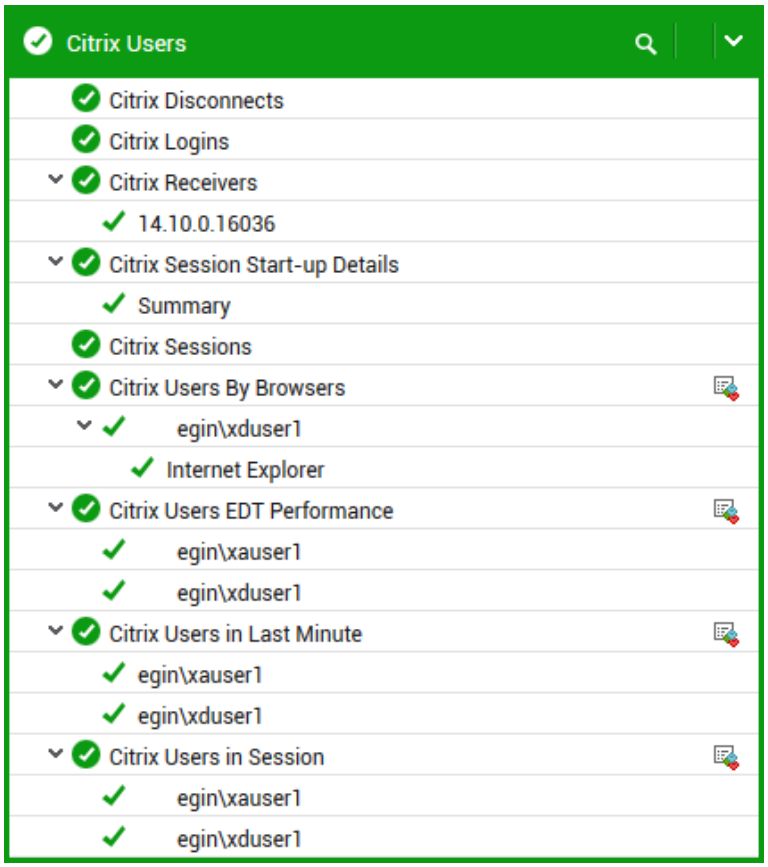


Figure 5.6: The tests associated with the Citrix Users layer

5.5.1 Citrix App Layering Test

Citrix App Layering lets you package any Windows application as a virtual disk, layer and deliver it, installation-free, to pooled desktops and session hosts. With App Layering, you can:

- Install and manage a single copy of your Windows operating system and a single copy of each of your applications in layers. A layer is simply a container for the file system objects and registry entries unique to that layer.
- Select any combination of layers to create layered Images that are deployable as desktops or session hosts.
- Deploy the layered images to virtual machine desktops and session hosts, making the applications available to users.

Citrix App Layering enables IT administrators to deliver applications that look, act and feel as if they are installed locally in the VM/Golden Image, but these applications are actually stored as separate manageable objects in their own virtual disks. With Citrix App Layering, any application can be separated from the Windows OS. As a result, IT administrators will only have a single OS Layer to manage regardless of the number of machine configurations (pools, silos, delivery groups). This simplifies the environment while reducing management time/complexity and the costs associated with OS and app management. Application layers can be attached to the virtual machine in one of two ways:

1. App Layers can be combined with an OS Layer, in a process called image publishing, and pushed to existing provisioning systems such as Citrix Provisioning Services, Citrix Machine Creation Services, or VMware View Composer;
2. App Layers can be attached to a VM at user login based on user AD group membership and app assignments. Each user can also receive a unique “Personalization Layer." This Personalization Layer will contain unique information for that user that will include things like local Windows profiles, application settings, files and folders created by the user and even user-installed applications.

When the App Layers are attached to a VM at the time of user login, the App Layers specific to a user should be fully attached without adding a significant delay to the logon process. If, for any reason, the App Layering fails or takes more time during the logon process, then the logon process may eventually fail or take longer time to complete. This in turn will impact the productivity of the users and overall user experience with the XenApp server. Therefore, it is imperative that administrators keep track of the time taken for attaching the App Layers. This can be easily done using the **Citrix App Layering** test.

This test tracks the App Layering process for each user and alerts administrators if any user is experiencing undue slowness during App Layering. This way, administrators are prompted to rapidly initiate remedial measures, so that the bottleneck can be resolved before it seriously impacts a user's productivity and experience. This test also monitors a log file i.e. *c:\ProgramData\Unidesk\Logs\ulayersvc.log* file created during App Layering process and reports the count of the warning messages and errors occurred. This way, administrators are alerted to the errors and warnings encountered during the App Layering process.

Note:

This test reports metrics only when the App Layering feature is enabled for the Citrix XenApp server.

Target of the test : Citrix XenApp Server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each user connected to the Citrix XenApp that is to be monitored.

This set also reports a set of measures for the **Summary** descriptor.

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** - Specify the port at which the specified host listens to. By default, this is *8080*.
4. **REPORT BY DOMAIN NAME** - By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who logged into the Citrix server. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.
5. **DD FOR INFORMATION** - By default, this flag is set to **No**, indicating that by default, the test does not generate detailed diagnostic measures for information events. eG Enterprise provides this option to restrict the amount of storage space on the eG database. However, if you want the test to generate and store detailed measures for information events, set the **DD FOR INFORMATION** flag to **Yes**.
6. **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**.

7. **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 enabled/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
App Layering duration	Indicates the amount of time taken for attaching the App Layered disks for this user.	Seconds	**This measure is not reported for the Summary descriptor.** Ideally, the value of this should be very low. An increase in the value of this measure may indicate the logon delay for the users.
App Layered disks attached to user session	Indicates the number of App Layered disks attached to this user.	Number	**This measure is not reported for the Summary descriptor.** Use the detailed diagnosis of this measure, to know the revision and name of the layered disks attached to the user sessions and the name of attached files.
Information messages	Indicates the number of information messages generated during attaching the App Layered disks for the users.	Number	A change in value of this measure may indicate infrequent but successful operations performed by one or more applications.

Measurement	Description	Measurement Unit	Interpretation
Warnings	Indicates the number of information messages generated during attaching the App Layered disks for the users.	Number	A high value of this measure indicates problems that may not have an immediate impact, but may cause future problems.
Errors	Indicates the number of errors encountered during attaching the App Layered disks for the users.	Number	A very low value (zero) is desired for this measure, as it indicates that App layering is being performed without any anomalies. An increasing trend or a high value indicates the occurrence of problems. If so, check the *ulayersvc.log* for more details.

5.5.2 Citrix Disconnects Test

A user session is terminated when a user logs off from the Citrix XenApp server or when the session is abruptly interrupted (e.g., due to server, network, or application errors). When a user logs off, all the applications started by the user are terminated. However, when a user disconnects, the applications started by the user will keep running on the server consuming resources. Hence, the number of disconnected sessions on a Citrix XenApp server should be kept to a minimum. Abrupt disconnects can significantly impact the end user experience, and hence, it is important to monitor the number of disconnected sessions at any point of time. This test measures the number of disconnected user sessions.

Target of the test : Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Citrix XenApp server that is to be 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 specified **HOST** listens to. By default, this is 1494.

4. **RECONNECT PERIOD** – This parameter is used by the test while computing the value for the **Quick reconnects** measure. This measure counts all the users who reconnected to the Citrix XenApp within the short period of time (in minutes) specified against **RECONNECT PERIOD**.
5. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who disconnected from the server recently. This way, administrators will be able to quickly determine which user belongs to which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.
6. **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.
7. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option.

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

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

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New disconnects:	Indicates the number of sessions that were disconnected during the last measurement period.	Number	The detailed diagnosis of this measure indicates the user, session ID, and client type for each newly disconnected session. This information can be used to track whether specific users are being disconnected often.

Measurement	Description	Measurement Unit	Interpretation
Quick reconnects:	Indicates the number of users who reconnected soon after a disconnect.	Number	The detailed diagnosis of this measure, if enabled lists the users who have reconnected quickly.
Total disconnects:	Indicates the total number of sessions that are in the disconnected state.	Number	

5.5.3 Citrix Logins Test

The Citrix Logins test monitors the new logins to the Citrix XenApp server.

Target of the test : Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Citrix XenApp that is to be 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 specified **HOST** listens to. By default, this is 1494.
4. **REPORT USING MANAGERTIME** – By default, this flag is set to **Yes**. This indicates that the user login time displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports will be based on the eG manager's time zone by default. Set this flag to **No** if you want the login times displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports to be based on the Citrix server's local time.
5. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user session that logged out. This default setting ensures that administrators are able to quickly determine the domains to which the users who logged out belonged. You can set this flag to **No** if you want detailed diagnosis to display only the username of the users who logged out.
6. **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**.

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

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

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

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New logins:	Indicates the number of new logins to this Citrix XenApp during the last measurement period.	Number	A consistent zero value could indicate a connection issue. Using the detailed diagnosis of the *New login*s measure, you can not only identify the users who logged in recently, but can also figure out when each user logged in and from which client machine.
Percent new logins:	Indicates the percentage of current sessions that logged in during the last measurement period.	Percent	
Sessions logging out:	Indicates the number of sessions that logged out.	Number	If all the current sessions suddenly log out, it indicates a problem condition that requires investigation.

Measurement	Description	Measurement Unit	Interpretation
			With the help of the detailed diagnosis of the *Sessions logging out* measure, you can identify the users who logged out, when every user logged in and from which client machine, and the duration of each user's session. In addition, you can also find out the time duration (in minutes) and the percentage of time for which the user was idle during each session. Abnormally long sessions on the server can thus be identified.

5.5.4 Citrix Sessions Test

This test reports performance statistics related to Citrix user sessions of the Citrix XenApp server.

Target of the test : Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Citrix XenApp that is to be 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 specified **HOST** listens to. By default, this is 1494.
4. **IGNORE DOWN SESSION IDS** - By default, this parameter is set to 65536,65537,65538– these are nothing but the default ports at which the listener component listens. If any of these ports go down, then by default, this test will not count any of the sessions that failed when attempting to connect to that port as a **Down session**. You can override this default setting by adding more ports or by removing one/more existing ports.
5. **REPORT USING MANAGERTIME** – By default, this flag is set to **Yes**. This indicates that the user login time displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports

will be based on the eG manager's time zone by default. Set this flag to **No** if you want the login times displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports to be based on the Citrix server's local time.

6. **REPORT BY DOMAIN NAME** - By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user session that logged out. This default setting ensures that administrators are able to quickly determine the domains to which the users who logged out belonged. You can set this flag to **No** if you want detailed diagnosis to display only the username of the users who logged out.
7. **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.
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.

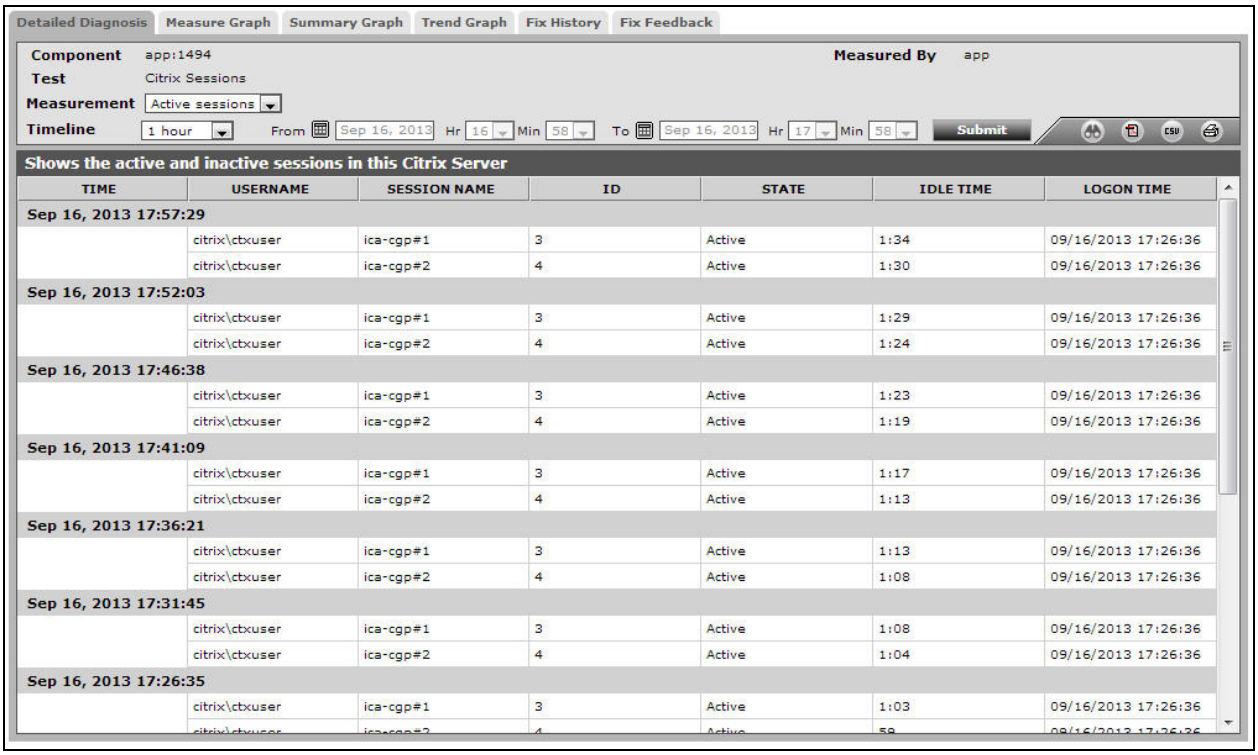
Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Established sessions:	Indicates the number of user sessions that are currently active on this server.	Number	This measure gives an idea of the server workload in terms of active sessions. Tracking the number of active sessions with time, a Citrix administrator can obtain information that can help him/her plan the capacity of their Citrix environment.

Measurement	Description	Measurement Unit	Interpretation
			The detailed diagnosis capability, if enabled, lists the active and inactive sessions on the Citrix XenApp server.
Idle sessions:	Indicates the number of sessions that are initialized and are currently ready to accept connections.	Number	To optimize the performance of a server, two default (idle) sessions are initialized before any client connections are made. For performance reasons, the number of idle sessions should be less than ten. Note that this test does not differentiate between RDP and ICA sessions.
Connected sessions:	Indicates the current number of sessions that are connected, but no user has logged on to the server.	Number	A consistent increase in the value of this measure could indicate that users are having trouble logging in. Further investigation may hence be required. Note that this test does not differentiate between RDP and ICA sessions.
Connecting sessions:	Indicates the number of sessions that are in the process of connecting.	Number	A very high value for this measure indicates a problem with the session or connection. Note that this test does not differentiate between RDP and ICA sessions.
Disconnected sessions:	Indicates the number of sessions from which users have disconnected, but which are still active and can be reconnected.	Number	Too many disconnected sessions running indefinitely on a Citrix XenApp server cause excessive consumption of the server resources. To avoid this, a session limit is typically configured for disconnected sessions on the Citrix XenApp server. When a session limit is reached for a disconnected

Measurement	Description	Measurement Unit	Interpretation
			session, the session ends, which permanently deletes it from the server. Note that this test does not differentiate between RDP and ICA sessions.
Listen sessions:	Indicates the current number of sessions that are ready to accept connections.	Number	Note that this test does not differentiate between RDP and ICA sessions.
Shadow sessions:	Indicates the current number of sessions that are remotely controlling other sessions.	Number	A non-zero value for this measure indicates the existence of shadow sessions that are allowed to view and control the user activity on another session. Such sessions help in troubleshooting/resolving problems with other sessions under their control.
Down sessions:	Indicates the current number of sessions that could not be initialized or terminated.	Number	Ideally, the value of this measure should be 0. By default, if sessions to any of these ports – *65536, 65537, 65538* – could not be initialized or terminated, they will not be counted as a 'down session'.
Init sessions:	Indicates the current number of sessions that are initializing.	Number	A high value for this measure could indicate that many sessions are currently experiencing initialization problems.

The detailed diagnosis capability of the *Established sessions* measure, if enabled, lists the active and inactive sessions on the Citrix XenApp server.



TIME	USERNAME	SESSION NAME	ID	STATE	IDLE TIME	LOGON TIME
Sep 16, 2013 17:57:29						
	citrix\ctxuser	ica-cgp#1	3	Active	1:34	09/16/2013 17:26:36
	citrix\ctxuser	ica-cgp#2	4	Active	1:30	09/16/2013 17:26:36
Sep 16, 2013 17:52:03						
	citrix\ctxuser	ica-cgp#1	3	Active	1:29	09/16/2013 17:26:36
	citrix\ctxuser	ica-cgp#2	4	Active	1:24	09/16/2013 17:26:36
Sep 16, 2013 17:46:38						
	citrix\ctxuser	ica-cgp#1	3	Active	1:23	09/16/2013 17:26:36
	citrix\ctxuser	ica-cgp#2	4	Active	1:19	09/16/2013 17:26:36
Sep 16, 2013 17:41:09						
	citrix\ctxuser	ica-cgp#1	3	Active	1:17	09/16/2013 17:26:36
	citrix\ctxuser	ica-cgp#2	4	Active	1:13	09/16/2013 17:26:36
Sep 16, 2013 17:36:21						
	citrix\ctxuser	ica-cgp#1	3	Active	1:13	09/16/2013 17:26:36
	citrix\ctxuser	ica-cgp#2	4	Active	1:08	09/16/2013 17:26:36
Sep 16, 2013 17:31:45						
	citrix\ctxuser	ica-cgp#1	3	Active	1:08	09/16/2013 17:26:36
	citrix\ctxuser	ica-cgp#2	4	Active	1:04	09/16/2013 17:26:36
Sep 16, 2013 17:26:35						
	citrix\ctxuser	ica-cgp#1	3	Active	1:03	09/16/2013 17:26:36

Figure 5.7: The detailed diagnosis of the Established sessions measure of the Citrix XenApp

5.5.5 Citrix Users in Sessions Test

The Citrix XenDesktop 7 environment is a shared environment in which multiple users may connect to a Citrix XenApp server/server farm and access a wide variety of applications. When server resources are shared, excessive resource utilization by a single user could impact the performance for other users. Therefore, continuous monitoring of the activities of each and every user on the server is critical. Towards this end, the **Citrix Users in Sessions** test assesses the traffic between the user terminal and the server, and also monitors the resources taken up by a user's session on the server. The results of this test can be used in troubleshooting and proactive monitoring. For example, when a user reports a performance problem, an administrator can quickly check the bandwidth usage of the user's session, the CPU/memory/disk usage of this user's session as well as the resource usage of other user sessions. The administrator also has access to details on what processes/applications the user is accessing and their individual resource usage. This information can be used to spot any offending processes/ applications.

Note:

This test will report metrics only if the XenApp server being monitored uses the .Net framework v3.0 (or above).

Target of the test : A Citrix XenApp server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each user connected to the Citrix XenApp that is to be 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 specified **HOST** listens to. By default, this is 1745.
4. **SHOW PUBLISHED APPS** – The detailed diagnosis of this test typically lists the applications accessed by a user during his/her session on the XenApp server, and the resource usage of each application. By default, when listing applications, the detailed diagnosis displays the underlying process name of each application (in the Process Name column), and not the actual display name using which the application is published on the XenApp server. This is because, the **SHOW PUBLISHED APPS** flag is set to **No** by default. If you set this flag to **Yes**, then the published application name will be displayed as the Process Name in the detailed diagnosis. **Prior to changing the flag status to 'Yes', you need to make sure that a 'Citrix XenDesktop Broker' component is also managed by the eG Enterprise system and is reporting metrics**.
5. **SHOW PUBLISHED DESKTOPS** – By default, this flag is set to **No**. If this flag is set to **Yes**, then the detailed diagnosis of this test will list the resource-intensive processes/applications accessed by a user along with the exact published desktop that has been used by the user to access the application. **Note that, in the detailed diagnosis, the 'host name' of the monitored server will be displayed as the 'published desktop name'.**
6. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the *domainname**username* of each user who accessed an application on the server. This way, administrators will be able to quickly determine which user logged into the server from which domain. If you want the detailed diagnosis to display only the *username* of these users, set this flag to **No**.
7. **COLLECT EXTENDED METRICS** – By default, this parameter is set to **No**, indicating that the test will report only a standard set of user experience metrics. To enable the test to collect additional metrics per user, set this flag to **Yes**.
8. **USE WMI** - By default, the eG agent uses WMI scripts to collect the user experience metrics from the XenApp server. This is why, the **USE WMI** flag is set to **Yes** by default. In some

environments, the WMI scripts used by the eG agent may not report valid values. Under such circumstances, it is best to switch off WMI script usage by setting the **USE WMI** flag to **No**. Once this is done, the eG agent will automatically use certain built-in executables to fetch the user experience metrics.

9. **SHOW ALL DESKTOP PROCESSES** - Using this flag, you can indicate whether the test should report top resource-intensive processes alone or all processes running in the background when the user accesses an application. By default, this flag is set to **No**, indicating that this test will report only top three resource-intensive processes e.g. CPU, Memory and IO Reads processes from the desktop OS processes. This helps the administrator optimize the database. To report all the processes, you need to set this flag to **Yes**.

10. **ENABLE BROWSER MONITORING** – By default, this flag is set to **No**, indicating that the eG agent does not monitor browser activity on the XenApp server. If this flag is set to **Yes,** then, whenever one/more IE (Internet Explorer) browser instances on the XenApp server are accessed, the detailed diagnosis of the *User sessions* measure will additionally reveal the URL being accessed via each IE instance and the resources consumed by every URL. Armed with this information, administrators can identify the web sites that are responsible for excessive resource usage by an IE instance.

11. **COMBINED IE REPORT** - **This flag is applicable only if the SHOW PUBLISHED APPS flag is set to Yes**.

    Typically, the detailed diagnosis of this test lists the applications that a user accessed on the XenApp server and the resource usage of each application. To fetch these details, the eG agent takes the published name (eg., Word) of each application that is open on the XenApp server, determines the underlying process and process arguments that drive the application (eg., winword.exe is the process that drives Word), and tries to find an exact match for this process name and arguments in the task manager. If an exact match is found, then, by default, the underlying process name of the application is displayed in the detailed diagnosis. The resource usage metrics that correspond to that process name in the task manager are also displayed as part of detailed diagnosis.

    For published applications that open in the Internet Explorer (IE) browser on XenApp, this process name matching algorithm may not work. This is because, if multiple applications on XenApp are opened using IE, each such application will open only in a different tab page of the IE browser. As a result, though the underlying process names will be different for each of these applications, in the task manager, the process names for all these applications will only be 'iexplore.exe'. Because of the name mismatch (between XenApp and the task manager), the eG agent will wrongly determine that no instance of an application is running, and will exclude that application name from the detailed diagnostics. To avoid this, with the **SHOW**

PUBLISHED APPS flag set to **Yes**, set this flag to **Yes**.

If this is done, then, the eG agent will be able to capture every application or application instance that is opened in a different tab page of an IE browser, despite the process name mismatch. Also, the eG agent will be able to collect detailed metrics of such applications and display them in the Detailed Diagnosis page against the process name 'Internet Explorer'.

12. **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.
13. **LOW BANDWIDTH IN MBPS** - Here, specify the bandwidth below which the test should consider the connection quality of the user on the target Citrix XenApp server as poor or weak. By default, this is set to 1 Mbps.
14. **HIGH BANDWIDTH IN MBPS** - Specify the bandwidth beyond which the test should consider the connection quality of the user on the target Citrix XenApp server as strong. By default, this is set to 8 Mbps.
15. **LOW LATENCY IN MS** - Indicate the latency below which the connection quality of the user on the target Citrix XenApp server is considered as *strong* by this test. By default, this is set to 120 milliseconds.
16. **HIGH LATENCY IN MS** - Indicate the latency beyond which the connection quality of the user on the target Citrix XenApp server is considered as *poor* or *weak* by this test. By default, this is set to 220 milliseconds.
17. **LOW ICA RTT IN MS** - Specify the ICA Round Trip Time (RTT) below which the connection quality of the user on the target Citrix XenApp server is considered as *strong* by this test. By default, this is set to 150 milliseconds.
18. **HIGH ICA RTT IN MS** - Indicate the ICA Round Trip Time (RTT) beyond which the connection quality of the user on the target Citrix XenApp server is considered as *poor* or *weak* by this test. By default, this is set to 260 milliseconds.
19. **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**.
20. **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
CPU usage for user's processes:	The CPU utilization for a session is the percentage of time that all of the threads/processes of a user session used the processor to execute instructions. If a user is connected via multiple sessions, the value reported is the sum of all cpu utilizations across all the sessions.	Percent	This value indicates the percentage of Cpu resources that are used by a specific user. Excessive CPU usage by a user can impact performance for other users. Check the detailed diagnosis to view the offending processes/applications.
Handles used by user's processes:	Indicates the total number of handles being currently held by all processes of a user.	Number	A consistent increase in the handle count over a period of time is indicative of malfunctioning of programs. Compare this value across users to see which user is using a lot of handles. Check detailed diagnosis for further information.
Audio bandwidth input:	Indicates the bandwidth used while transmitting sound/audio to this user.	Kbps	Comparing these values across users will reveal which user is sending/receiving bandwidth-intensive sound/audio files over the ICA channel.

Measurement	Description	Measurement Unit	Interpretation
Audio bandwidth output:	Indicates the bandwidth used while receiving sound/audio from this user.	Kbps	To minimize bandwidth consumption, you may want to consider disabling client audio mapping.
Input bandwidth:	Indicates the average bandwidth used for client to server communications for all the sessions of a user.	KB/Sec	
Output bandwidth:	Indicates the average bandwidth used for server to client communications for all the sessions of a user.	KB/Sec	
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. These measures will be reported only if the **COLLECT EXTENDED METRICS** flag is set to '**Yes**'.
COM bandwidth output:	Indicates the bandwidth used when receiving data from this user's COM port.		
Input compression:	Indicates the average compression ratio for client to server traffic for all the sessions of a user.	Number	
Output compression:	Indicates the average compression ratio for server to client traffic for all the sessions of a user.	Number	
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. If bandwidth consumption is too high, you may want to consider disabling client drive mapping on the client device. Client drive mapping allows

Measurement	Description	Measurement Unit	Interpretation
			users logged on to a virtual desktop from a client device to access their local drives transparently from the ICA session. Alternatively, you can conserve bandwidth by even refraining from accessing large files with client drive mapping over the ICA connection. These measures will be reported only if the **COLLECT EXTENDED METRICS** flag is set to '**Yes**'.
Drive bandwidth output:	Indicates the bandwidth used when the virtual desktop performs file operations on the client's drive.	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	
PN bandwidth input:	Indicates the bandwidth used from this user to virtual desktop by Program Neighborhood to obtain application set details.	Kbps	Comparing the values of these measures across users will reveal which user has been transmitting/receiving bandwidth-intensive PN traffic. These measures will be reported only if the **COLLECT EXTENDED METRICS** flag is set to '**Yes**'.
PN bandwidth output:	Indicates the bandwidth, used from the virtual desktop to this user by Program Neighborhood to obtain application set details.	Kbps	
I/O reads for user's processes:	Indicates the rate of I/O reads done by all processes being run by a user.	KBps	These metrics measure the collective I/O activity (which includes file, network and device I/O's) generated by all the processes being executed by a user. When viewed along with the system I/O metrics reported by the DiskActivityTest, these measures

Measurement	Description	Measurement Unit	Interpretation
			help you determine the network I/O. Comparison across users helps identify the user who is running the most I/O-intensive processes. Check the detailed diagnosis for the offending processes/applications.
I/O writes for user's processes:	Indicates the rate of I/O writes done by all processes being run by a user.	KBps	
Screen refresh latency - avg:	Indicates the average time interval measured at the client between the first step (user action) and the last step (graphical response displayed) of this user's interactions with the server. The value reported is the average of the latencies for all the current sessions of a user. **This measure maps to the 'ICA RTT' measure in Citrix Director.**	Secs	This is a measurement of the screen lag that a user experiences while interacting with the XenApp server. In other words, is the latency detected from when the user hits a key until the response is displayed. Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when interacting with the XenApp server. If both the Screen refresh latency and Client network latency measures report high values, it implies that network slowness is contributing to user-perceived Citrix slowness (i.e., the problem is not due to the Citrix servers, but probably due to the network connection that the user is connecting from - e.g., a wireless WAN). If Screen refresh latency is high and Client network latency is low, this implies that there is a bottleneck in the Citrix stack that is causing user experience to be poor (e.g., overloaded server or virtual platform, slowness in storage, etc.). Slowness can also occur because of client-side processing delays on the receiver end.

Measurement	Description	Measurement Unit	Interpretation
Screen refresh latency - deviation:	The latency deviation represents the difference between the minimum and maximum measured latency values for a session. The value reported is the average of the latency deviations for all the current sessions of a user.	Secs	Ideally, the deviation in latencies over a session should be minimum so as to provide a consistent experience for the user. This measure will be reported only if the **COLLECT EXTENDED METRICS** flag is set to '**Yes**'.
Screen refresh latency - last:	Represents the average client latency for the last request from a user. The latency is measured by the Citrix XenApp server based on packets sent to and from each client during a session - this includes network delay plus server side processing delays. The value reported is the average of the last latencies for all the current sessions of a user.	Secs	A consistently high latency may be indicative of performance degradations with the Citrix XenApp servers. Possible reasons for an increase in latency could be increased network delays, network congestion, server slow-down, too many simultaneous users on the server etc.
Memory usage for user's processes:	This value represents the ratio of the resident set size of the memory utilized by the user to the physical memory of the host system, expressed as a percentage. If a user is connected via multiple sessions, the value reported is the sum of all memory utilizations across all the sessions.	Percent	This value indicates the percentage of memory resources that are used up by a specific user. By comparing this value across users, an administrator can identify the most heavy users of the Citrix XenApp server. Check the detailed diagnosis to view the offending processes/applications.
User sessions:	Indicates the current number of sessions for a particular user.	Number	A value of 0 indicates that the user is not currently connected to the Citrix XenApp server. Use the detailed diagnosis of this

Measurement	Description	Measurement Unit	Interpretation
			measure to know the details of the sessions.
Input line speed:	Indicates the average line speed from the client to the server for all the sessions of a user.	Kbps	
Output line speed:	Indicates the average line speed from the server to the client for all the sessions of a user.	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 HDX channel. If bandwidth consumption is too high, you may want to consider disabling printing. Alternatively, you can avoid printing large documents over the HDX connection.
Printer bandwidth output:	Indicates the bandwidth used when the desktop responds to print jobs issued by this user.	Kbps	
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. These measures will be reported only if the **COLLECT EXTENDED METRICS** flag is set to '**Yes**'.
Speed screen data channel bandwidth output:	Indicates the bandwidth used from virtual desktop to this user for data channel traffic.	Kbps	
HDX media stream for flash v2 data bandwidth input:	Indicates the bandwidth used from this user to virtual desktop for flash v2 data traffic.	Kbps	Comparing the values of these measures across users will reveal which user has been transmitting/receiving bandwidth-intensive flash v2 data.
HDX media stream for flash v2 data bandwidth output:	Indicates the bandwidth used from the virtual desktop to this user for flash	Kbps	

Measurement	Description	Measurement Unit	Interpretation
	v2 data traffic.		
Page faults for user's processes:	Indicates the rate of page faults seen by all processes being run by a user.	Faults/Sec	*Page Faults* occur in the threads executing in a process. A page fault occurs when a thread refers to a virtual memory page that is not in its working set in main memory. If the page is on the standby list and hence already in main memory, or if the page is in use by another process with whom the page is shared, then the page fault will not cause the page to be fetched from disk. *Excessive page faults* could result in decreased performance. Compare values across users to figure out which user is causing most page faults.
CPU time used by user's sessions:	Indicates the percentage of time, across all processors, this user hogged the CPU.	Percent	The CPU usage for user's processes measure indicates the percentage of overall server CPU time that a user is using. For example, if a user is taking up one of the server's CPUs for 100% of the time and there are 8 CPUs on the server, CPU usage for user's processes will be 12.5% (100/800). While 12.5% may seem to be a low number, the fact that the user is taking up one of the CPUs of the server is significant. Hence, CPU time used by user's session measure is a better indicator of CPU usage by users. In the above example, since the user is consuming 100% of one processor, CPU time used by user's session will be 100%. A high value of this measure or a consistent increase in the value of this measure demands attention. Use the detailed diagnosis to know what CPU intensive activities are being performed by the user.

Measurement	Description	Measurement Unit	Interpretation
Input bandwidth usage:	Indicates the percentage HDX bandwidth consumed by client to server traffic of this user.	Percent	Compare the value of these measures across users to know which user is consuming the maximum HDX bandwidth.
Output bandwidth usage:	Indicates the percentage HDX bandwidth consumed by the server to client traffic of this user.	Percent	
Thinwire bandwidth input:	Indicates the bandwidth used from client to server for ThinWire traffic.	Kbps	Typically, ICA traffic is comprised of many small packets, as well as a some large packets. Large packets are commonly generated for initial session screen paints and printing jobs, whereas the ongoing user session is principally comprised of many small packets. For the most part, these small packets are the highest priority ICA data called Thinwire. Thinwire incorporates mouse movements and keystrokes. Compare the value of these measures across users to know which user's keystrokes and mouse movements are generating bandwidth-intensive traffic. **Note:** • This measure will be reported only if the collect extended metrics flag is set to Yes. • This measure will report the value 0, if Framehawk is enabled for a user.
Thinwire bandwith output:	Indicates the bandwidth used from server to client for ThinWire traffic.	Kbps	**Note:** • This measure will be reported only if the collect extended

Measurement	Description	Measurement Unit	Interpretation
			metrics flag is set to Yes. • This measure will report the value 0, if Framehawk is enabled for a user.
Seamless bandwidth input:	Indicates the bandwidth used from client to server for published applications that are not embedded in a session window.	Kbps	Compare the value of these measures across users to know which user is accessing bandwidth-intensive applications that are not in a session window. This measure will be reported only if the **COLLECT EXTENDED METRICS** flag is set to '**Yes**'.
Seamless bandwidth output:	Indicates the bandwidth used from server to client for published applications that are not embedded in a session window.	Kbps	This measure will be reported only if the **COLLECT EXTENDED METRICS** flag is set to '**Yes**'.
Resource shares:	Indicates the total number of resource shares used by this user.	Number	By comparing the value of this measure across users, you can identify the user who is hogging the resources.
Frame rate:	Indicates the rate at which frames are processed during this user session.	Frames/Sec	FPS is how fast your graphics card can output individual frames each second. It is the most time-tested and ideal measure of performance of a GPU. Higher the value of this measure, healthier is the GPU.
Framehawk frame rate:	Indicates the rate at which frames are processed by the Framehawk virtual channel, if it is enabled for this user session.	Frames/Sec	The Framehawk virtual channel optimizes the delivery of virtual desktops and applications to users on broadband wireless connections, when high packet loss or congestion occurs. A high value is desired for this measure, as it indicates faster delivery of applications to users, which in turn

Measurement	Description	Measurement Unit	Interpretation
			makes for a better user experience. You can compare the value of this measure with that of the Frame rate measure of a user to ascertain whether/not the Framehawk virtual channel has indeed enhanced that user's experience with applications deployed on XenApp. If this comparison reveals that the value of this measure is higher than that of the Frame rate measure, it is a clear indicator of the effectiveness of the Framehawk virtual channel. **Note:** This measure will report the value 0 if Framehawk is not enabled for a user or if the device from which the user is accessing the application does not support Framehawk.
Framehawk network bandwidth:	Indicates the bandwidth consumption of this user session when the Framehawk virtual delivery channel is used.	KB	This is a good measure of the effectiveness of Framehawk in optimizing the bandwidth usage over the virtual delivery channel. A low value is desired for this measure. **Note:** This measure will report the value 0 if Framehawk is not enabled for a user or if the device from which the user is accessing the application does not support Framehawk.
Framehawk latency:	Indicates the latency experienced by this user session when the Framehawk virtual delivery channel is used.	Secs	To judge the effectiveness of Framehawk, compare the value of this measure with that of the ICA network latency measure for a Framehawk-enabled user. If the comparison reveals a lower value for this measure, it implies that Framehawk has

Measurement	Description	Measurement Unit	Interpretation
			succeeded in minimizing the latencies over the delivery channel. **Note:** This measure will report the value 0 if Framehawk is not enabled for a user or if the device from which the user is accessing the application does not support Framehawk.
Framehawk network loss:	Indicates the percentage of packet loss experienced by this user session when the Framehawk virtual delivery channel is used.	Percent	If the value of this measure is very low, it indicates that Framehawk has been very effective in minimizing the loss ofpackets that typically occur when data is transmitted or received over a channel. **Note:** This measure will report the value 0 if Framehawk is not enabled for a user or if the device from which the user is accessing the application does not support Framehawk.
Client network latency:	Indicates the latency experienced by this user when transmitting/receiving data over the ICA channel. **This measure maps to the 'Network RTT' measure in Citrix Director.**	Secs	This measure represents the network latency detected between the ICA client and the Citrix XenApp server being monitored. If both the Screen refresh latency and Client network latency measures report high values, it implies that network slowness is contributing to user-perceived Citrix slowness (i.e., the problem is not due to the Citrix servers, but probably due to the network connection that the user is connecting from - e.g., a wireless WAN). If Screen refresh latency is high and Client network latency is low, this

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>implies that there is a bottleneck in the Citrix stack that is causing user experience to be poor (e.g., overloaded server or virtual platform, slowness in storage, etc.). Slowness can also occur because of client-side processing delays on the receiver end.</td></tr>
<tr><td>Total bandwidth</td><td>Indicates the total bandwidth usage of the sessions of this user.</td><td>Kbps</td><td>Compare the value of this measure across users to know which user is consuming the maximum bandwidth.</td></tr>
<tr><td>Total time in session:</td><td>Indicates the time that has elapsed since this user logged in.</td><td>Mins</td><td>Compare the value of this measure across users to know which user has been logged in for the longest time.</td></tr>
<tr><td>Active time in last measure period:</td><td>Indicates the percentage of time in the last measurement period during which this user actively used the server.</td><td>Percent</td><td>Ideally, the value of this measure should be 100%.
A low value for this measure denotes a high level of inactivity recently.</td></tr>
<tr><td>Time since last activity:</td><td>Indicates the time that has elapsed since this user performed an action on the server.</td><td>Minutes</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 IDLE TIME 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>Note:
By default, this measure reports the Measure Values listed in the table above. In the graph of this measure</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			however, the value of this measure is represented using their numeric equivalents only.
Total idle time in session:	Indicates the total time for which this user was idle during the session.	Minutes	If the value of this measure is the same as the value of the *Total time in session* measure for a user, it means that the user has been idle throughout the session. If the value of this measure is close to the value of the *Total time in session* measure for a user, it implies that the user has been idle for a long time. If the value of this measure is much lesser than the value of the *Total time in session* measure for a user, it means that the user has been active for most part of the session.
Working set memory for user's processes:	Indicates the current size of the working set of this user's processes	MB	The Working Set is the set of memory pages touched recently by the threads in a process. If free memory in the server is above a threshold, pages are left in the Working Set of a process even if they are not in use. When free memory falls below a threshold, pages are trimmed from Working Sets. If they are needed they will then be soft-faulted back into the Working Set before leaving main memory. If multiple processes are running in the user's session, the memory working set reported is the sum of the working sets for all the user's processes. Comparing the working set across users indicates which user(s) are taking up excessive memory. Check the detailed diagnosis to view the offending processes/applications.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Processes running in the user's session:</td><td>Indicates the count of processes running in this user's session.</td><td>Number</td><td></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>Note:
By default, this measure reports the Measure Values listed in the table above. In the graph of this measure however, the value of this measure is represented using their numeric equivalents only.
A major alert will be generated when this test reports the "Poor connection" value for this measure. Likewise, a minor alert will be generated when the value of this measure is "Weak connection".</td></tr>
<tr><td>Input delay for user's sessions - max</td><td>Indicates the maximum amount of time lag detected between this user's input through any input device (e.g., mouse, keyboard) and the time at which the</td><td>Seconds</td><td>Poor application performance is one of the most difficult problems to diagnose by the administrators. Traditionally, diagnosis was done</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
	application picked up the input.		by collecting CPU, memory, disk I/O and a few other metrics. The data collected from traditional metrics were not sufficient to figure out the root cause of poor performance of the applications since the variations measured by the metrics were large. In virtual environments where multiple users accessed an application from remote at the same time, users faced difficulties in accessing the application whenever there was an increase in the count of users. The more the users are accessing the application, the higher was the CPU usage of the systems in the environment and the higher was the user input delays i.e., the users were forced to wait for a longer duration to interact with the application. The user input delay is measured by how long any user input (such as mouse or keyboard usage) stays in the queue before it is picked up by a process. These two measures capture such user input delays at the user session level. These insights enable administrators to accurately identify which user's Citrix experience is being scarred by user input delays. These measures will be reported only on Windows 2019 (and above).

Measurement	Description	Measurement Unit	Interpretation
Input delay for user's sessions - avg	Indicates the average amount of time lag detected between this user's input through any input device (e.g., mouse, keyboard) and the time at which the application picked up the input.	Seconds	Ideally, the values of these measures should be 0 or very low.

5.5.6 Citrix Users By Browsers Test

In recent times, browsers have become one of the common ways to access many applications in enterprises. The same browser may be used for accessing multiple applications. Further, users may even use browsers to access non-corporate web sites from their environment. In modern architectures, a lot of the processing is done by the scripts executed on browsers. This further adds processing tasks to the browsers. Citrix administrators need to know when exactly specific browser instances started taking up excessive resources (CPU, memory, disk) and most importantly, what URLs were accessed when a browser started taking up resources. The **Citrix Users By Browsers** test helps administrators in this regard!

This test auto-discovers the browsers accessed by each user connected to a Citrix XenApp server/server farm, and for each browser, reports the number of sessions initiated and the number of processes running. This test also reports the URLs that were accessed when a browser started taking up resources. Using this test, administrators can figure out the URL and browser that is responsible for excessive resource utilization by a user.

Target of the test : Any Citrix server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the Citrix XenApp server monitored

Configurable parameters for the test

Parameters	Description
Test Period	How often should the test be executed. By default, this is 15 minutes.
Host	The host for which the test is to be configured.

Parameters	Description
Port	Refers to the port used by the Citrix server .
Report by Domain Name	By default, the flag is set to **Yes**. This implies that by default, this test will report metrics for every domainname\username configured for this test. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the test to report metrics for the username alone, then set this flag to **No**.
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.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
User sessions	Indicates the total number of user sessions initiated through this browser.	Number	
Processes running in user's sessions	Indicates the number of processes running for this browser.	Number	The detailed diagnosis of this measure lists the Process ID, Website Title, Website URL and the Website domain.
CPU usage for	Indicates the percentage	Percent	A high value for this measure is a

Measurement	Description	Measurement Unit	Interpretation
user's processes	of CPU utilized by the processes running for this browser.		cause of concern. Comparing the value of this measure across browsers helps administrators in identifying the browser that is utilizing too much of CPU resources. The detailed diagnosis of this measure lists the session ID, Process ID, the CPU utilized by the process and the memory utilized by the process.
Memory usage for user's processes	Indicates the percentage of memory utilized by the processes running for this browser.	Percent	A high value for this measure is a cause of concern. Comparing the value of this measure across browsers helps administrators in identifying the browser that is utilizing too much of memory resources. The detailed diagnosis of this measure lists the session ID, Process ID, the CPU utilized by the process and the memory utilized by the process.

5.5.7 Citrix Users in Last Minute Test

The Citrix XenDesktop 7 environment is a shared environment in which multiple users may connect to a Citrix XenApp server/server farm and access a wide variety of applications. When server resources are shared among multiple users, sudden surges in resource utilization of a single user even for a minute could impact the performance of other users. Therefore, continuous monitoring of the activities of each and every user on the server is critical. To achieve this, administrators can use the **Citrix Users in Last Minute** test! This test monitors the resources utilization at every minute during each user's session.

Using this test, administrators can quickly check the CPU/memory/disk usage of each user's session as well as the latency experienced by each user. In the process, this test also sheds light on the rate at which the data was read/written during running different processes in each user's session. The detailed diagnosis provided by the test helps administrator to view details on what processes/applications the user is accessing and their individual resource usage. This information can be used to spot any offending processes/ applications.

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 *Citrix XenApp server* as the **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.

Note:

This test will report metrics only if the XenApp server being monitored uses the .Net framework v3.0 (or above).

Target of the test : A Citrix XenApp server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each user connected to the Citrix XenApp that is to be monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed. By default, this is set to *60 seconds*.
2. **HOST -** The host for which the test is to be configured.
3. **PORT** – The port number at which the specified **HOST** listens to. By default, this is 1494.
4. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the *domainname\username* of each user who accessed an application on the server. This way, administrators will be able to quickly determine which user logged into the server from which domain. If you want the detailed diagnosis to display only the *username* of these users, set this flag to **No**.
5. **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**.

6. **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
CPU usage for user's processes	The CPU utilization for a session is the percentage of time that all of the threads/processes of a user session used the processor to execute instructions. If a user is connected via multiple sessions, the value reported is the sum of all cpu utilizations across all the sessions.	Percent	This value indicates the percentage of Cpu resources that are used by a specific user. Excessive CPU usage by a user can impact performance for other users. Check the detailed diagnosis to view the offending processes/applications.
Memory usage for user's processes	This value represents the ratio of the resident set size of the memory utilized by the user to the physical memory of the host system, expressed as a percentage. If a user is connected via multiple sessions, the value reported is the sum of all memory utilizations across all the sessions.	Percent	This value indicates the percentage of memory resources that are used up by a specific user. By comparing this value across users, an administrator can identify the most heavy users of the Citrix XenApp server. Check the detailed diagnosis to view the offending processes/applications.
Client network	Indicates the latency	Secs	This measure represents the network

Measurement	Description	Measurement Unit	Interpretation
latency	experienced by this user when transmitting/receiving data over the ICA channel.		latency detected between the ICA client and the Citrix XenApp server being monitored. If both the Screen refresh latency and Client network latency measures report high values, it implies that network slowness is contributing to user-perceived Citrix slowness (i.e., the problem is not due to the Citrix servers, but probably due to the network connection that the user is connecting from - e.g., a wireless WAN). If Screen refresh latency is high and Client network latency is low, this implies that there is a bottleneck in the Citrix stack that is causing user experience to be poor (e.g., overloaded server or virtual platform, slowness in storage, etc.). Slowness can also occur because of client-side processing delays on the receiver end.
Screen refresh latency - avg	Indicates the average time interval measured at the client between the first step (user action) and the last step (graphical response displayed) of this user's interactions with the server. The value reported is the average of the latencies for all the current sessions of a user.	Secs	This is a measurement of the screen lag that a user experiences while interacting with the XenApp server. In other words, is the latency detected from when the user hits a key until the response is displayed. Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when interacting with the XenApp server. If both the Screen refresh latency and Client network latency measures report high values, it implies that

Measurement	Description	Measurement Unit	Interpretation
			network slowness is contributing to user-perceived Citrix slowness (i.e., the problem is not due to the Citrix servers, but probably due to the network connection that the user is connecting from - e.g., a wireless WAN). If Screen refresh latency is high and Client network latency is low, this implies that there is a bottleneck in the Citrix stack that is causing user experience to be poor (e.g., overloaded server or virtual platform, slowness in storage, etc.). Slowness can also occur because of client-side processing delays on the receiver end.
I/O reads for user's processes	Indicates the rate of I/O reads done by all processes being run by this user.	KBps	These metrics measure the collective I/O activity (which includes file, network and device I/O's) generated by all the processes being executed by a user. When viewed along with the system I/O metrics reported by the DiskActivityTest, these measures help you determine the network I/O. Comparison across users helps identify the user who is running the most I/O-intensive processes. Check the detailed diagnosis for the offending processes/applications.
I/O writes for user's processes	Indicates the rate of I/O writes done by all processes being run by this user.	KBps	

5.5.8 Citrix Multimedia Audio Logs Test

To troubleshoot issues with the audio experience on Citrix XenApp, you can use the the **Citrix Multimedia Audio Logs** test. This test periodically searches the Citrix-Multimedia-AudioSVC/Admin logs for specific patterns of event IDs/event sources/event descriptions and alerts administrators if messages matching the configured patterns are found.

Target of the test : A Citrix XenApp server

Agent deploying the test : An internal agent

Outputs 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 Citrix-Multimedia-AudioSVC/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*.
   - Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use *all* to indicate that all the event sources

have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded.

- 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 made 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 the audio is in a healthy state. An increasing trend or high value indicates the existence of problems.

Measurement	Description	Measurement Unit	Interpretation
			Use the detailed diagnosis of this measure 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 audio operations. Use the detailed diagnosis of this measure 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 audio problems that may not have an immediate impact, but may cause future problems. Use the detailed diagnosis of this measure 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 an audio component cannot automatically recover from. This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems. An increasing trend or high value indicates the existence of fatal/irrepairable problems in the audio. The detailed diagnosis of this measure describes all the critical audio events that were generated during the last measurement period.
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.

5.5.9 Citrix Multimedia Rave Log Test

RAVE (Remote Audio and Video Extensions) is the technology behind SpeedScreen Multimedia Acceleration. RAVE supports high quality playback of media streams that can be decoded by a media player that uses DirectShow or DirectX Media Objects (DMO). To determine whether SpeedScreen Multimedia Acceleration is functioning or not and to investigate issues in the same, administrators can use the Citrix-Multimedia-Rave/Admin logs that Windows provides. This test provides administrators with insights into these logs. It scans the Citrix-Multimedia-Rave/Admin logs for specific patterns of event IDs/event sources/event descriptions. If entries matching these patterns are found in the logs captured recently, this test reports the number and nature of such messages.

Target of the test : A Citrix XenApp server

Agent deploying the test : An internal agent

Outputs 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 Citrix-Multimedia-Rave/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*.
   - Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use *all* to indicate that all the event sources have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded.
   - 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 selectivelable 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
Error messages:	This refers to the number of error events that were generated.	Number	A very low value (zero) indicates that SpeedScreen Multimedia Acceleration is functioning properly. An increasing trend or high value indicates the existence of problems. Use the detailed diagnosis of this measure 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 SpeedScreen Multimedia Acceleration operations. Use the detailed diagnosis of this measure for more details.
Warnings:	This refers to the	Number	A high value of this measure

Measurement	Description	Measurement Unit	Interpretation
	number of warnings that were generated when the test was last executed.		indicates Speed Screen Multimedia Acceleration problems that may not have an immediate impact, but may cause future problems. Use the detailed diagnosis of this measure 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 an SpeedScreen Multimedia Acceleration cannot automatically recover from. This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems. An increasing trend or high value indicates the existence of fatal/irrepairable problems in the RAVE techology. The detailed diagnosis of this measure describes all the critical events related to RAVE that were generated during the last measurement period.
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

Measurement	Description	Measurement Unit	Interpretation
			events that were generated during the last measurement period.

5.5.10 Citrix Multimedia Flash Log Test

If Flash redirection does not work for clients connecting to the XenDesktop server 7.0 (or above), administrators can use the Citrix-Multimedia-Flash/Admin logs to investigate the reasons for the same. The **Citrix Multimedia Flash Log** test scans the Citrix-Multimedia-Flash/Admin logs for specific patterns of event IDs/event sources/event descriptions. If entries matching these patterns are found in the logs captured recently, this test reports the number and nature of such messages.

Target of the test : A Citrix XenApp server

Agent deploying the test : An internal agent

Outputs 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 Citrix-Multimedia-Flash/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*.
   - Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use *all* to indicate that all the event sources have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded.
   - 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 enabled/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
Error messages:	This refers to the number of error events that were generated.	Number	A very low value (zero) indicates that Flash technology is functioning properly. An increasing trend or high value indicates the existence of problems. Use the detailed diagnosis of this measure 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 Flash operations. Use the detailed diagnosis of this measure 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 Flash problems that may not have an immediate impact, but may cause future problems. Use the detailed diagnosis of this measure 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 Flash cannot automatically recover from. This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems.

Measurement	Description	Measurement Unit	Interpretation
			An increasing trend or high value indicates the existence of fatal/irrepairable problems in the Flash techology. The detailed diagnosis of this measure describes all the critical events related to Flash that were generated during the last measurement period.
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.

5.5.11 Citrix Broker Agent Test

A broker agent lies at the heart of any VDI deployment, and is the key component for assigning resources to end users. The Citrix broker is what the client talks to in order to know what VM it is allowed to access. It is the middle component between desktops in the data center and the client and its waits for connections. When someone logs in, the Citrix broker is the one that checks with Active Directory to make sure the user is authorized. Then it checks its own DB to figure out what desktop this user has access to and finally allows the user access to the list of desktops and eventually hands that off. It also allows you to manage the Desktop sessions and Application sessions etc.

By keeping an eye on the Citrix Broker Agent, you can understand the current session load on the broker, the clients contributing to the load, and the nature of the sessions. This is exactly what the **Citrix Broker Agent** Test does. This test monitors the Citrix broker agent and reports the count of

clients registered with the Citrix broker, the session load imposed by these clients on the Citrix server, and the nature of this load - – i.e., are they application sessions? or are they desktop sessions?

Target of the test : Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Citrix XenApp server that is to be 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 specified **HOST** listens to. By default, this is 1494.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Registrations:	Indicates the number of times this XenApp server registered with the broker agent during the last measurement period.	Number	
Deregistrations:	Indicates the number of times this XenApp server deregistered from the broker agent during the last measurement period.	Number	The value 1 for this measure indicates that the XenApp server is disconnected from the broker. This can hurt user access to desktops/applications on the XenApp server. You may want to investigate and ascertain the reason for the disconnect/deregistration and clear the bottleneck quickly, so that user-server communication is not disrupted.
Total application	Indicates the number of	Number	If the value of the Total sessions

Measurement	Description	Measurement Unit	Interpretation
sessions:	application sessions running on the Citrix server during the last measurement period.		measure increases continuously, it may indicate a probable overload on the XenApp server. In such a situation, you can compare the value of this measure with that of the Total desktop sessions measure to know what type of sessions are contributing the most to the overload.
Total desktop sessions:	Indicates the number of desktop sessions running on the Citrix server during the last measurement period.	Number	If the value of the Total sessions measure increases continuously, it may indicate a probable overload on the XenApp server. In such a situation, you can compare the value of this measure with that of the Total application sessions measure to know what type of sessions are contributing the most to the overload.
Total sessions:	Indicates the total number of sessions on the Citrix server during the last measurement period.	Number	This is a good indicator of the load on the XenAp server.

Chapter 5: Citrix Session Start-up Details Test

Figure 5.8 depicts a typical user logon process to Citrix XenApp.

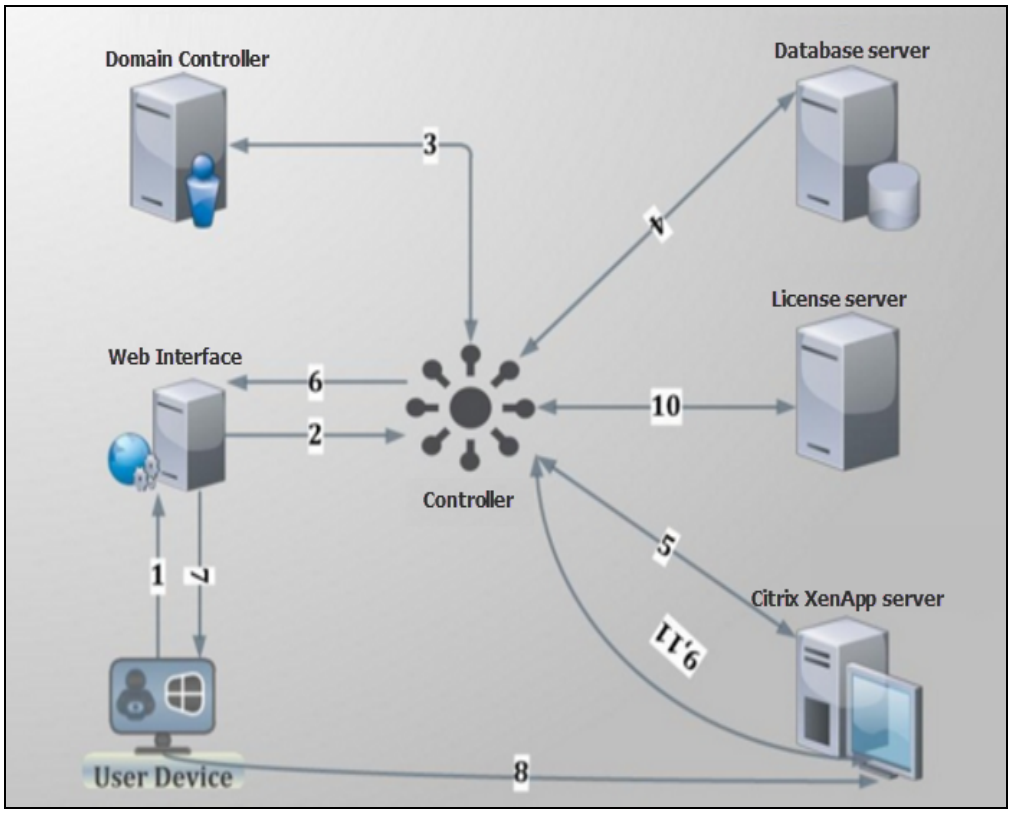


Figure 5.8: Citrix user logon process

The process depicted by Figure 5.8 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 application is available for current user.
5. Controller then interacts with Citrix XenApp server to gather information about the availability of application.
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 application.
8. Once the application is assigned, the user establishes connection with that application.
9. The Citrix XenApp server 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 Citrix XenApp server.

From the discussion above, it can be inferred that login processing happens at two different places - at the delivery controller, and inside the Citrix XenApp server. While login, authentication, and application brokering happen on the delivery controller, session creation and setup happens inside the XenApp server. 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 XenApp server - 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 XenApp server, then use the Citrix Session Start-up Details test.

With the Citrix Session Start-up Details test, administrators can receive deep visibility into the XenApp end of the Citrix logon process. This test takes an administrator into the XenApp server, reveals the users who are currently logged on to the server, and accurately reports the average time it took for the sessions of each user to start inside the server. 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 an application, e.g., by clicking an icon, to the point at which an instance of the ICA client has finished opening a connection to Presentation Server. While connection-brokering mechanisms, such as Web Interface for Citrix XenApp server or Program Neighborhood Agent,

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 XenApp server. This includes user authentication, client device mapping, profile loading, login scripts execution, and finally, starting the user's application (in the case of a desktop this will be explorer.exe). If a session already exists and a new application is being started through session sharing, only the application start stage will be considered as part of server start-up.

Target of the test : A Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each user who is currently logged on to the Citrix XenApp server that is 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 number at which the specified **HOST** listens to. By default, this is 1494.
4. **REPORT USING MANAGERTIME** – By default, this flag is set to **Yes**. This indicates that the user login time displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports will be based on the eG manager's time zone by default. Set this flag to **No** if you want the login times displayed in the **DETAILED DIAGNOSIS** page for this test and in the Thin Client reports to be based on the Citrix server's local time.
5. **REPORT BY DOMAIN NAME** - By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the *domainname\username* of each user session that logged out. This default setting ensures that administrators are able to quickly determine the domains to which the users who logged out belonged. You can set this flag to **No** if you want detailed diagnosis to display only the username of the users who logged out.
6. **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.

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

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

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

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
New sessions:	Indicates the number of sessions currently open for this user on the XenApp server.	Number	Compare the value of this measure across users to know which user has the maximum number of open sessions. In the event of an overload, this will point you to that user who is contributing the most to the workload of the XenApp server. Use the detailed diagnosis of this measure to view the complete details of each user session. Such details includes the name and IP address of the client from which every session was launched, when session creation started, and when it ended. With the help of this information, administrators can quickly pinpoint those sessions that may have taken too long to be created.
Session start- up duration:	Indicates the average time taken by the sessions of	Seconds	Compare the value of this measure across users to know which user's sessions took the longest to start-up. To

Measurement	Description	Measurement Unit	Interpretation
	this user to complete start- up activities.		know what is causing this slowness, compare the values reported by all the other 'duration' measures of this test for that user. This will quickly lead you to where that user's session start- up process is spending the maximum time.
Group policy processing duration:	Indicates the time taken by this user's session to process group policies.	Seconds	Compare the value of this measure across users to know which user's sessions took the longest time to process group policies. If that 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. . 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. Logon performance improves when fewer Group Policies are applied. Merge GPOs when possible instead of having multiple GPOs.
Logon script execution duration:	Indicates the time taken for the login	Seconds	Compare the value of this measure across users to know for which user the

Measurement	Description	Measurement Unit	Interpretation
	script to execute for this user.		login script took the longest time to execute. If this user complains of slowness, then, you can compare the value of this measure with that of the other 'duration' measures of that user to figure out what could have really caused the slowness.
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 ICA connection between this user's client device and XenApp server has been established.	Seconds	In the case of a shared session, this duration will normally be much shorter, as many of the set-up costs associated with the creation of a new connection to the server are not incurred. Backup URL count When any user complains of slowness, you may want to compare the value of this measure with that of the Server side session start-up processing duration measure of that user to know whether a client-side issue or a server-side issue is responsible for the slowness. 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*

Measurement	Description	Measurement Unit	Interpretation
			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 XenApp plugin is the application launch mechanism. It records the number of back-up URL retries before a successful launch. Note that this is the only start-up metric that is a measure of attempts, rather than time duration.	Number	If this metric has a value higher than 1, it indicates that the Web Interface server is unavailable and the XenApp Plugin (formerly known as Program Neighborhood Agent) is attempting to connect to back-up Web Interface servers to launch the application. **A value of 2** means that the main Web Interface server was unavailable, but the XenApp Plugin managed to the launch the application 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.

Measurement	Description	Measurement Unit	Interpretation
			• 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 XenApp plugin is the application launch mechanism** . It measures the time needed by this user's sessions to retrieve the list of applications from the Web Interface service.	Seconds	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 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 an ICA session with the XenApp server.
Configuration file download duration:	**This measure is relevant when the XenApp plugin is the application launch**	Seconds	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

Measurement	Description	Measurement Unit	Interpretation
	mechanism . It measures the time this user's sessions took to retrieve the configuration file from the XML server.		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 XenApp plugin is the application launch mechanism** . It measures the time required by this user's sessions to obtain the user credentials.	Seconds	Note that this is only measured when the credentials are entered manually by the user. Because this metric may be artificially inflated if a user fails to provide credentials in a timely manner, it is subtracted from the *Start-up client duration*. However, in the event that the user manually inputs the credentials, and the value of this measure is higher than that of all the other *client start-up* metrics that this test reports, it is a clear indicator that any connection delay that the user may have experienced is owing to slowness in obtaining user credentials.
ICA file download	This measure is	Seconds	The overall process here is:

Measurement	Description	Measurement Unit	Interpretation
duration:	relevant when the XenApp plugin or Web Interface is the application launch mechanism. This is the time it takes for this user's client to download the ICA file from the web server.		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 XenApp server 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 plugin (client). This measure represents the time it takes for the complete process (step 1 to 6). The measure stops counting time when the client receives the ICA file data. The *Launch page web server duration* measure on the other hand, covers the Web server portion of the process (that is, steps 3 and 4). If the *ICA file download duration* is high, but the *Launch page web server duration* is normal, it implies that the server-side processing of the launch was successful, but there were

Measurement	Description	Measurement Unit	Interpretation
			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 application launch mechanism** . It measures the time needed by this user's sessions to process the launch page (launch.aspx) on the Web Interface server.	Seconds	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, such as the XenApp server. Check to see if the network connection between Web Interface and XenApp is slow or some XenApp servers are down or overloaded. If the Web server seems okay, consider reviewing the XenApp farm 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	Seconds	This metric is collected when a client device directly queries the XML Broker to retrieve published application information stored in IMA (for example, when using the XenApp Plugin or a Custom ICA Connection). This

Measurement	Description	Measurement Unit	Interpretation
	address.		measure is only gathered for new sessions since session sharing occurs during startup if a session already exists. When this metric is high, it indicates the XML Broker is taking a lot of time to resolve the name of a published application to an IP address. Possible causes include a problem on the client, issues with the XML Broker, such as the XML Broker being overloaded, a problem with the network link between the two, or a problem in IMA. Begin by evaluating traffic on the network and the XML Broker.
Name resolution web server duration:	**This measure is relevant when the XenApp plugin or Web Interface is the application launch mechanism** . It is the time it takes the XML service to resolve the name of a published application to a XenApp Server address.	Seconds	When this metric is high, there could be an issue with the Web Interface server or the XenApp plugin site (formerly known as the Neighborhood Agent site), 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 published application to a XenApp 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 XenApp plugin (previously known as Program Neighborhood Agent) or from a user clicking a Web Interface page icon. This metric applies to all sessions launched

Measurement	Description	Measurement Unit	Interpretation
			through the Web Interface or the XenApp plugin (formerly, the Program Neighborhood Agent).
Session lookup duration:	Indicates the time this user's sessions take to query every ICA session to host the requested published application.	Seconds	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, from the moment wfica32.exe is launched to the establishment of the connection.	Seconds	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 *Server side session start-up processing duration*, you may want to compare the value of this measure with all other client start- up measures to determine whether/not session creation is the process that is slowing down the application launch.
Ticket response web server duration:	**This measure is relevant when the XenApp plugin or Web Interface is the application launch mechanism**. This is the time this user's sessions take to get a ticket (if required) from the STA server or XML service.	Seconds	When this metric is high, it can indicate that the Secure Ticket Authority (STA) server or the XML Broker are overloaded.
Reconnect enumeration	**This measure is relevant when the**	Seconds	Compare the value of this measure with that of other *client start-up* metrics for a

Measurement	Description	Measurement Unit	Interpretation
duration:	**XenApp plugin is the application launch mechanism**. This is the time it takes this user's client to get a list of reconnections.		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 XenApp plugin or Web Interface is the application launch mechanism**. This is the time it takes the Web Interface to get the list of reconnections for this user from the XML service.	Seconds	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.
Server side session start-up processing duration:	This is the high-level server- side connection start-up metric. It includes the time spent on the XenApp server to perform the entire start-up operation.	Seconds	In the event of an application starting in a shared session, this metric is normally much smaller than when starting a completely new session, which involves potentially high- cost tasks such as profile loading and login script execution. 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,*

Measurement	Description	Measurement Unit	Interpretation
			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 the server in creating the session for this user.	Seconds	This duration starts when the ICA client connection has been opened and ends when authentication begins. **This should not be confused with 'Server side session start- up processing duration'**. Note: When monitoring Citrix XenApp servers below v7.6, this measure may sometimes report abnormally high values. If you want to disregard such values, then do the following: • Edit the eg_tests.ini file (in the <EG_ INSTALL_ DIR>\manager\config directory). • Look for the **CitrixEuemMaxSCSD** parameter in the file. • By default, this parameter is set to *600* (seconds). This means that, if the *Session creation server duration* measure reports a value that is higher than 600 seconds (by default), then

Measurement	Description	Measurement Unit	Interpretation
			eG Enterprise will hide this measure from the UI. You can change the value of the **CitrixEuemMaxSCSD** parameter to suit your needs. • Finally, save the eg_tests.ini file. When monitoring Citrix XenApp servers above v7.6 however, this test will **not** report such abnormal values for the *Session creation server duration* measure. So, the **CitrixEuemMaxSCSD** parameter is not applicable in this case.
User credential obtention by server duration:	Indicates the time taken by the server to obtain the credentials of this user.	Seconds	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 *Server side session start-up processing 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.

Measurement	Description	Measurement Unit	Interpretation
Pass- through credentials duration:	Indicates the time spent by the server performing network operations to obtain credentials for this user.	Seconds	This only applies to a Security Support Provider Interface login (a form of pass-through authentication where the client device is a member of the same domain as the server and Kerberos tickets are passed in place of manually entered credentials).
Program neighbourhood credentials obtention server duration:	Indicates the time needed for the server to cause the Program Neighborhood instance running on the client ("Program Neighborhood Classic") to obtain this user's credentials.	Seconds	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*.
Credential authentication duration:	Indicates the time spent by the server when authenticating the user's credentials against the authentication provider, which may be Kerberos, Active Directory or a Security Support Provider Interface (SSPI).	Seconds	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,* and *Endpoint printers mapping duration* – to know what is the root-cause of delays in server start-up.

Measurement	Description	Measurement Unit	Interpretation
Profile load server duration:	Indicates the time required by the server to load this user's profile.	Seconds	If this metric is high, consider your Terminal Services profile configuration. Citrix Consulting has found that when customers have logon times greater than 20 seconds, in most cases, this can be attributed to poor profile and policy design. Roaming profile size and location contribute to slow session starts. When a user logs onto a session where Terminal Services roaming profiles and home folders are enabled, the roaming profile contents and access to that folder are mapped during logon, which takes additional resources. In some cases, this can consume significant amounts of the CPU usage. Consider using the Terminal Services home folders with redirected personal folders to mitigate this problem. In general, consider using **Citrix Profile management** to manage user profiles in Citrix environments. This tool also provides logging capabilities to help 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 the server to run this user's login script (s).	Seconds	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

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>scripts or use environment variables instead.</td></tr>
<tr><td>Endpoint resources mapping duration:</td><td>Indicates the time needed for the server to map this user’s client drives, devices and ports.</td><td>Seconds</td><td>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.</td></tr>
<tr><td>Endpoint printers mapping duration:</td><td>Indicates the time required for the server to synchronously map this user’s client printers.</td><td>Seconds</td><td>If the configuration is set such that printer creation is performed asynchronously, no value is recorded for this measure as it is does not impact completion of the session start-up.

On the other hand, if excessive time is spent mapping printers, it is often the result of the printer autocreation policy settings. The number of printers added locally on the users' client devices and your printing configuration can directly affect your session start times. When a session starts, XenApp 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>Has user's session been reconnected?</td><td>Indicates whether/not this user session reconnected.</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>Yes</td><td>1</td></tr><tr><td>No</td><td>0</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 Measure Values 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>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>
Note:

By default, this measure reports the above- mentioned Measure Values 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>
Note:

By default, this measure reports the above- mentioned Measure Value 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>
Note:

By default, this measure reports the above- mentioned Measure Value 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>
</table>

Measurement	Description	Measurement Unit	Interpretation
User account discovery	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 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.
Domain Controller 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.

Measurement	Description	Measurement Unit	Interpretation
			With the help of the detailed diagnostics, you can identify the Group Policy that is associated with the user, 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	
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	Indicates the	Number	Ideally, the value of this measure should

Measurement	Description	Measurement Unit	Interpretation
extensions with error state	number of errors registered when client side extensions were used for processing group policies for this user during the last measurement period.		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 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

Measurement	Description	Measurement Unit	Interpretation
			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 controller is slow. If the network connection between the VM and domain controller is slow for a

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>user, then this measure will report the value Yes. If it is fast, then this measure will report the value No (connection is fast).

The numeric values that correspond to the above-mentioned measure values are as follows:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Yes</td><td>1</td></tr>
<tr><td>No (connection is fast)</td><td>2</td></tr>
</table>

Note:

By default, this test reports the Measure Values 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.
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 Group Policy has configured for this link, use

</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			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

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>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.

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>
Note:

By default, this test reports the Measure Values 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</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			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 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.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
A 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 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>
Note:
By default, this test reports the Measure Values 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>

5.5.12 Citrix Receivers Test

If a user complains of slowness when accessing applications/dekstops launched on a Citrix server, administrators may instantly want to know which type/version of Receiver that user is connecting from . This knowledge will ease the troubleshooting pains of administrators as it will clearly indicate if the slowdown occurred owing to the usage of an unsupported or an outdated Receiver. To obtain this knowledge, administrators can use the **Citrix Receivers** test. With the help of this test, administrators can identify the Citrix Receivers that are in use, determine which user is logging into the Citrix environment using which Receiver, and in the process, figure out if any Receiver-related issues are contributing to a user's unsatisfactory experience with Citrix.

Target of the test : Any Citrix server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every Reveiver type/version auto-discovered

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 Citrix server.
4. **REPORT BY DOMAIN NAME** – By default, this flag is set to **Yes**. This implies that by default, the detailed diagnosis of this test will display the domainname\username of each user who logged into the Citrix server. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the detailed diagnosis to display the username alone, then set this flag to **No**.
5. **REPORT BY RECEIVER TYPE -** By default, this flag is set to **No**. This implies that by default, this test will report one set of metrics for every Receiver version. To make sure that the test reports metrics for each Receiver type instead, set this flag to **Yes.**
6. **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
Users connected from this type:	Indicates the number of users who are currently connected to Citrix via Receivers of this type/version.	Number	Use the detailed diagnosis of this measure to know which user connected via a Receivers of a particular type/version.

5.5.13 Citrix Users EDT Performance Test

Adaptive Transport – a new transport mechanism for virtual servers is faster, more scalable, improves application interactivity, and more interactive on long-haul WAN and internet connections. When Adaptive Transport is used, ICA virtual channels intelligently switch the underlying protocol for user sessions between Enlightened Data Transport protocol and TCP to deliver faster, scalable and reliable performance.

When users have connected to the virtual desktops via EDT protocol, administrators may often want to know how the experience of each user is. This is why, the **Citrix Users EDT Performance** test auto-discovers the users who are logged into the virtual desktops (via the EDT protocol) provisioned using the Citrix XenApp servers, and measures the bandwidth usage, packets transmission and reception, and latency of each user with the server. In the process, bandwidth-hungry, latent user sessions can be accurately isolated.

Target of the test : Citrix XenApp

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every user who is currently logged into the virtual desktops (via the EDT protocol) provisioned using the Citrix XenApp servers.

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 specified **HOST** listens to. By default, this is 1494.
4. **CITRIX HOME** - By default, the **CITRIX HOME** parameter is set to *none* indicating that the eG agent would automatically discover the location at which the Virtual Delivery Agent (VDA) is installed for collecting the metrics of this test. If the Virtual Delviery Agent is installed in a different location in your Citrix environment, then indicate that location in the **CITRIX HOME** text box.
5. **REPORT BY DOMAIN NAME** - By default, the flag is set to **Yes**. This implies that by default, this test will report metrics for every domainname\username configured for this test. This way, administrators will be able to quickly determine which user logged in from which domain. If you want the test to report metrics for the username alone, then set this flag to **No**.
6. **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.
7. **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
Number of sessions	Indicates the number of sessions intiated by this user.	Number	The value 0 indicates that the user is not currently connected to the Citrix server.

Measurement	Description	Measurement Unit	Interpretation
Bandwidth	Indicates the bandwidth usage of all sessions of this user.	Kbps	Compare the value of this measure across users to know which user is consuming the maximum bandwidth.
Round trip time	Indicates the round trip latency between the virtual machine and this user.	Seconds	Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing higher latency when connecting to a virtual machine via EDT protocol.
Flow window	Indicates the size of the flow window.	KB	The flow window and congestion window are used to control the congestion in the network. The smaller the value of both the windows, the data will be sent without any delay. The larger the value, the data will be added up in the sent queue and it will be sent with delay.
Congestion window	Indicates the size of the congestion window.	KB	
Sent packets	Indicates the number of EDT packets sent by this user.	Packets	
Received packets	Indicates the number of EDT packets received by this user.	Packets	
Retransmitted packets	Indicates the number of EDT packets that were retransmitted by the user.	Packets	
Lost sent packets	Indicates the number of packets lost by this user during transmission.	Packets	Ideally, the value of this measure should be zero. Comparing the value of these measures across users will

Measurement	Description	Measurement Unit	Interpretation
			enable administrators to quickly and accurately identify users who have extensively lost packets during transmission and reception of packets via EDT protocol.
Lost received packets	Indicates the number of packets lost by this user during reception.	Packets	
Sent acknowledgements	Indicates the number of acknowledgements that were received by this user for sending the EDT packets.	Number	
Sent negative acknowledgements	Indicates the number of negative acknowledgements that were received by this user for sending the EDT packets.	Number	
Received acknowledgements	Indicates the number of acknowledgements that were received by this user for reception of EDT packets.	Number	
Received negative acknowledgements	Indicates the number of negative acknowledgements that were received by this user for reception of EDT packets.	Number	

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.