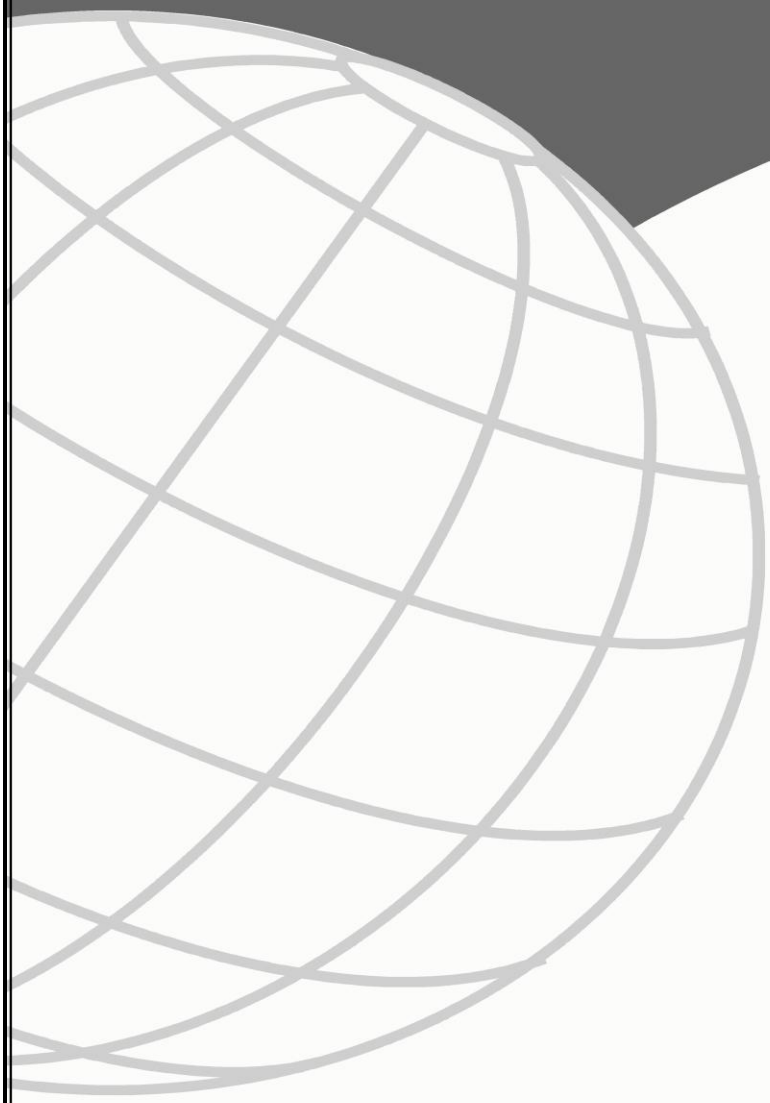


Enabling Service Excellence

Monitoring KVM servers

eG Enterprise v6

Restricted Rights Legend

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

Trademarks

Microsoft Windows, Windows NT, Windows 2000, Windows 2003 and Windows 2008 are either registered trademarks or trademarks of Microsoft Corporation in United States and/or other countries.

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

Copyright

©2014 eG Innovations Inc. All rights reserved.

Table of Contents

INTRODUCTION **1**

HOW DOES EG ENTERPRISE MONITOR KVM ENVIRONMENTS? **3**

2.1 Pre-requisites for Monitoring KVM Infrastructures 3

2.1.1 General Pre-requisites 3

2.1.2 Pre-requisites for auto-discovering the VMs on a KVM Server 4

2.2 Pre-requisites for Obtaining the "Inside View" of Windows VMs, using the eG VM Agent 4

2.3 Pre-requisites for Obtaining the "Inside View" of VMs, without using the eG VM Agent 4

2.4 Configuring the Remote Agent to Obtain the Inside View of Windows VMs, using the eG VM Agent 5

2.5 Communication between the eG Agent and the eG VM Agent 8

2.5.1 Licensing of the eG VM Agent 9

2.5.2 Benefits of the eG VM Agent 9

2.6 Configuring Windows Virtual Machines to Support the eG Agent's Inside View without the eG VM Agent 9

2.6.1 Enabling ADMIN$ Share Access on Windows Virtual Guests 10

2.6.2 Configuring Windows Firewalls to Allow File and Print Sharing 19

2.7 Monitoring Models offered by eG Enterprise for KVM Infrastructures 22

MONITORING KVM SERVERS **23**

3.1 The Operating System Layer 23

3.1.1 Host Details - KVM Test 24

3.1.2 Memory - KVM Test 25

3.1.3 Storage Pools Test 28

3.1.4 Storage Volumes Test 29

3.2 The Network Layer 31

3.2.1 Virtual Networks Test 31

3.3 The Outside View of VMs Layer 32

3.3.1 KVM Virtual Machines Test 33

3.3.2 KVM VM Details Test 36

3.3.3 VM Connectivity Test 46

3.3.4 VM Jobs Test 48

3.4 The Inside View of VMs Layer 50

3.4.1 Disk Activity - VM Test 51

3.4.2 Disk Space - VM Test 61

3.4.3 System Details - VM Test 65

3.4.4 Uptime - VM Test 71
3.4.5 Windows Memory - VM Test 75
3.4.6 Windows Network Traffic - VM Test 81
3.4.7 Network Traffic - VM Test 85
3.4.8 TCP - VM Test 88
3.4.9 TCP Traffic - VM Test 92
3.4.10 Handles Usage - VM Test 96
3.4.11 Windows Services - VM Test 101
3.4.12 Memory Usage - VM Test 105
3.4.13 Page File - VM Test 112
3.4.14 Domain Time Sync – VM Test 116
3.4.15 Disk Alignment – VM Test 120
MONITORING KVM SERVERS WITH VMS HOSTING DESKTOP APPLICATIONS 125
4.1 The Outside View of VMs Layer 126
4.1.1 KVM VDI Logins Test 127
4.1.2 KVM Virtual Machines Test 131
4.1.3 KVM VM Details Test 133
4.1.4 VDI Applications Test 138
4.2 The Inside View of Desktops layer 143
4.2.1 Terminal to Desktop Connection Test 144
4.2.2 Domain Time Sync – VM Test 148
4.2.3 Browser Activity – VM Test 152
4.3 Troubleshooting 158
4.3.1 Troubleshooting the Failure of the eG Remote Agent to Obtain the 'Inside View' of a Windows VM 158
4.3.2 Troubleshooting the Failure of the eG Remote Agent to Connect to or Report Measures for Linux Guests 160
4.3.3 Troubleshooting the Failure of the eG Remote Agent to Connect to or Report Measures for Windows 2008 or Windows Vista VMs 163
CONCLUSION 164

Table of Figures

Figure 1.1: The architecture of a KVM Server 1
Figure 2.1: Welcome screen of the eG VM Agent installation wizard 6
Figure 2.2: Accepting the license agreement 6
Figure 2.3: Specifying the install directory of the eG VM Agent 7
Figure 2.4: Specifying the VM agent port 7
Figure 2.5: A summary of your specifications 8
Figure 2.6: Finishing the installation 8
Figure 2.7: The ADMIN$ share does not exist 10
Figure 2.8: Admin$ share pre-exists 11
Figure 2.9: Creating the ADMIN$ share 11
Figure 2.10: Clicking the Add button 12
Figure 2.11: Selecting the administrative user to whom access rights are to be granted 12
Figure 2.12: The administrator account granted access permissions 13
Figure 2.13: Defining the Security settings for the ADMIN$ share 13
Figure 2.14: Adding the administrator account 14
Figure 2.15: The Administrator account in the Security list 14
Figure 2.16: Selecting the Share option from the shortcut menu 15
Figure 2.17: Clicking on Advanced Sharing 16
Figure 2.18: Enabling the ADMIN$ share 16
Figure 2.19: Clicking on the Add button 17
Figure 2.20: Allowing a domain administrator to access the folder 17
Figure 2.21: Allowing full access to the local/domain administrator 18
Figure 2.22: Applying the changes 18
Figure 2.23: Selecting the guest OS 19
Figure 2.24: Opening the Windows Firewall 20
Figure 2.25: The General tab of the Windows Firewall dialog box 20
Figure 2.26: Deselecting the 'Don't allow exceptions' check box 21
Figure 2.27: Enabling 'File and Printer Sharing' 21
Figure 2.28: Opening ports 22
Figure 3.1: The layer model of the KVM Server 23
Figure 3.2: The tests mapped to the Operating System layer 24
Figure 3.3: The detailed diagnosis of the Used physical memory measure 27
Figure 3.4: The detailed diagnosis of the Percentage of physical memory used measure 28
Figure 3.5: The tests mapped to the Network layer 31
Figure 3.6: The tests mapped to the Outside View of VMs layer 32
Figure 3.7: The detailed diagnosis of the Registered VMs measure 35
Figure 3.8: The detailed diagnosis of the Running VMs measure 36
Figure 3.9: The detailed diagnosis of the Data reads measure 43
Figure 3.10: the detailed diagnosis of the Read requests measure 43
Figure 3.11: The detailed diagnosis of the Data writes measure 43
Figure 3.12: The detailed diagnosis of the Write requests measure 44
Figure 3.13: The detailed diagnosis of the Data transmitted measure 44
Figure 3.14: The detailed diagnosis of the Data received measure 45
Figure 3.15: The detailed diagnosis of the Packets transmitted measure 45
Figure 3.16: The detailed diagnosis of the Packets received measaure 45
Figure 3.17: A list of guest operating systems on a KVM server host and their current state 51
Figure 3.18: The tests mapped to the Inside View of VMs layer 51
Figure 3.1: Configuring a VM test 58
Figure 3.2: The VM user configuration page 58
Figure 3.3: Adding another user 59
Figure 3.4: Associating a single domain with different admin users 60
Figure 3.5: The test configuration page displaying multiple domain names, user names, and passwords 61
Figure 3.6: The top 10 CPU consuming processes 70
Figure 3.19: The detailed diagnosis of the Handles used by processes measure 101
Figure 3.20: The detailed diagnosis of the Processes using handles above limit in VM measure 101
Figure 4.1: The layer model of the KVM VDI server 126
Figure 4.2: The tests mapped to the Outside View of VMs layer 126
Figure 4.1: The current state of the desktops configured on the KVM VDI server host that is monitored 143
Figure 4.2: The tests associated with the Inside View of Desktops layer of a KVM VDI server 143
Figure 4.3: The detailed diagnosis of the Running browser instances measure 157

1

Introduction

KVM is a full virtualization solution for x86 processors supporting hardware virtualization (Intel VT or AMD-V). It consists of two main components: A set of Kernel modules (kvm.ko, kvm-intel.ko, and kvm-amd.ko) providing the core virtualization infrastructure and processor specific drivers and a userspace program (qemu-kvm) that provides emulation for virtual devices and control mechanisms to manage VM Guests (virtual machines). The term KVM more properly refers to the Kernel level virtualization functionality, but is in practice more commonly used to reference the userspace component.

KVM is an open source software using which you can run multiple virtual machines that are running unmodified Linux or Windows images. Each virtual machine has a private virtualized hardware i.e., a network card, disk, graphics adapter, etc.

VM Guests (virtual machines), virtual storage and networks can be managed with libvirt-based and QEMU tools. libvirt is a library that provides an API to manage VM Guests based on different virtualization solutions, among them KVM and Xen. It offers a graphical user interface as well as a command line program. The QEMU tools are KVM/QEMU specific and are only available for the command line.

When you install the KVM module, it creates a bare metal hypervisor on the Linux kernel. You can then load virtual machine images onto the hypervisor, running separate operating systems. The KVM architecture hosts the virtual machine images as regular Linux processes, so that each virtual machine image can use all of the features of the Linux kernel, including hardware, security, storage, and applications.

The following illustration shows the main components of a KVM Server host.

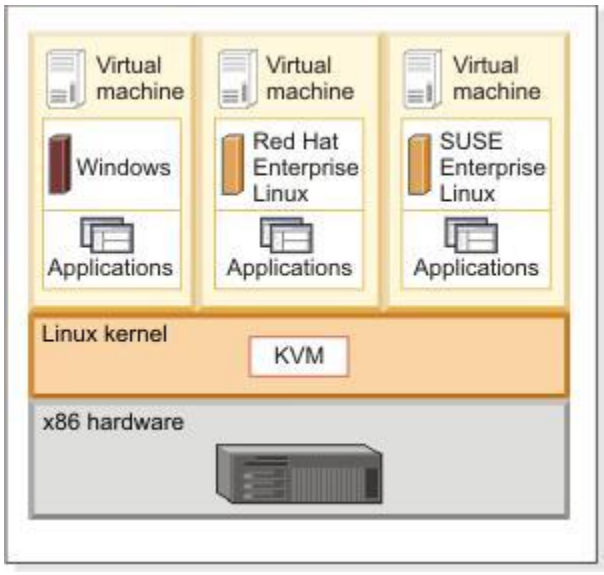


Figure 1.1: The architecture of a KVM Server

You can use any type of storage that is supported by Linux to store virtual machine images, including local disks, SCSI, or network-attached storage such as NFS and SAN. The KVM hypervisor also supports virtual machine images on shared file systems such as the Global File System (GFS2) allowing the images to be shared by multiple hosts.

With the KVM hypervisor, you can perform live migrations and move a running virtual machine between physical

hosts with no interruption to service. You can save the current state of a virtual machine to disk so that you can restart running the virtual machine from its previous state at a later time.

Because the KVM architecture hosts the virtual machine images as regular Linux processes, you can use the standard Linux security measures to isolate the images and provide resource controls. The Linux kernel includes SELinux along with sVIRT to isolate virtual images. In addition, you can use control groups (cgroups) to further restrict a set of tasks to a set of resources and monitor resource use. For more information about securing your KVM environment, see KVM security.

The KVM architecture supports the memory management features of Linux. In addition, with Kernel Same-page Merging (KSM) virtual images can share memory pages. If the virtual images have identical memory pages, those pages are merged into a single page that is then shared by the virtual images.

The KVM hypervisor supports a variety of guest operating systems, including Linux distributions, Microsoft Windows, and other platforms including OpenBSD, FreeBSD, OpenSolaris, Solaris x86, and MS DOS.

2 How does eG Enterprise Monitor KVM Environments?

To monitor virtual infrastructures, eG Enterprise uses a patented In-N-Out monitoring approach that is designed to address the requirements outlined in Chapter 1. The eG Single Agent not only monitors the KVM kernel, but also uses the **libvirt** APIs and commands to report real-time resource usage of each of the guest VMs. The metrics collected report on the percentage of the KVM server's resources that each the VMs on the server are using - i.e., the relative loading of the guest VMs. This represents the view of how a guest VM and its applications are doing - from the "outside" - i.e., from outside the guest VM.

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

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

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

2.1 Pre-requisites for Monitoring KVM Infrastructures

2.1.1 General Pre-requisites

- Enable the remote agent to communicate with the eG manager port (default: 7077).
- If VMs running on multi-byte operating systems are to be monitored (e.g., *Windows Japanese*), then the remote agent monitoring such VMs should also run on a multi-byte operating system.
- 32-bit VMs that are to be monitored in an agentless manner should be configured with at least 2 GB RAM,

and 64-bit VMs require at least 4 GB RAM. If more than four KVM servers are being monitored in an agentless manner, then the RAM capacity of the VMs should be increased proportionately.

2.1.2 Pre-requisites for auto-discovering the VMs on a KVM Server

By default, the remote agent monitoring the KVM server cannot determine the IP address of the guest VMs hosted on the server. Instead, the MAC address of the guest VMs are determined by the remote agent. To obtain the IP address of the guest VMs from the MAC address, administrators are required to execute the following command from the command prompt of the KVM server host:

arp –an

Once this command is executed, the IP address of the VMs will be retrieved from the network neighbor cache of the KVM server and displayed. If the IP address is not available in the cache, then administrators have to manually ping the IP address of the VMs once from the KVM server host. The VMs will now be auto discovered after a wait period of 5 to 10 minutes. To auto discover the VMs immediately, you may need to restart the eG remote agent.

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

- Install the eG VM Agent on each Windows VM. For details on how to install the eG VM Agent, refer to Section 2.4 of this document.
- Enable the remote agent to communicate with the port at which the eG VM Agent listens (default port: 60001).
- Set the inside view using flag for all the "inside view" tests to **eG VM Agent (Windows)**.

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

- Ensure that the remote agent has IP connectivity to at least one of the network interfaces of the VMs.
- The **ADMIN$** share should be enabled for all Windows-based virtual guests being monitored and the administrative account must have permissions to this share drive. Refer to Section 2.6.1 of this document for a step-by-step procedure to achieve this.
- To enable the remote agent to communicate with the Windows VMs, an administrative account login and password (either a local account or a domain account) must be provided when configuring the eG monitoring capabilities.
- In case of VMs with the Windows XP/Windows 2003/Windows 2008/Windows Vista/Windows 7 operating systems, the firewall on the guest should be explicitly configured to allow Windows File and Print Sharing services which are required for the remote agent on the vSphere/ESX host to communicate with the guest operating system. Refer to Section 2.6.2 of this document for a detailed procedure.
- For monitoring a Windows VM, TCP port 139 must be accessible from the remote agent to the VM.
- For monitoring a Linux VM, the SSH port (TCP port 22) must be enabled for communication between the remote agent and the VM being monitored.

Note

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

- Login to the eG manager.
- Edit the **eg_tests.ini** file (in the **<EG_INSTALL_DIR>\manager\config** directory) on the eG manager host.
- In the **[AGENT_SETTINGS]** section of the file, set the **JavaSshPortForVm** parameter to an SSH port of your choice. By default, this parameter is set to *22*.
- If your environment consists of multiple Linux VMs, each listening on a different SSH port, then, you can specify a comma-separated list of SSH ports against the **JavaSshPortForVm** parameter. For example: *7711,7271,8102*
- Finally, save the file.

- For obtaining the "inside view" of VMs running Windows Vista/Windows 7/Windows 2008 operating systems, the **eGurkhaAgent** service of the eG remote agent should be configured to run using *domain administrator* privileges. Refer to the *eG User Manual* for the procedure. For obtaining the "inside view" of other Windows VMs however, the remote agent service requires no such privileges.
- Set the inside view using flag for all the "inside view" tests to **Remote connection to VM (Windows)**.

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

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

In many production environments, strict security restrictions are enforced, and it may not be possible to configure a monitoring solution with domain administration privileges for each of the VMs. To handle such environments, the eG VM monitor uses a lightweight monitoring component called the **eG VM Agent**, which is installed inside each of the VMs to obtain metrics regarding the health of the VMs. The **eG VM Agent** can be best described as a software that can be installed on the Windows virtual machines of a virtual infrastructure to allow a single eG agent to obtain an inside view of these VMs, without **domain administrator privileges**.

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

- Manually install the **eG VM Agent** on every Windows VM using the executable that eG Enterprise includes;
- Bundle the **eG VM Agent** as part of a template VM, and use this template to create multiple VMs; this way, the eG VM Agent is automatically available in all the VMs that are created using the template;
- Use a software distribution solution such as Microsoft System Center to distribute the **eG VM Agent** software to existing VMs from a central location;

Use the install procedure that is ideal for your environment, and quickly get the **eG VM Agent** up and running. The

detailed manual installation procedure has been discussed hereunder:

1. To install the eG VM Agent on a 32-bit VM, double-click on the **eGVMAgent.exe**, and to install the same on a 64-bit VM, double-click the **eGVMAgent_64.exe**.
2. Figure 2.1 then appears. Click on the **Next** button in Figure 2.1 to continue.

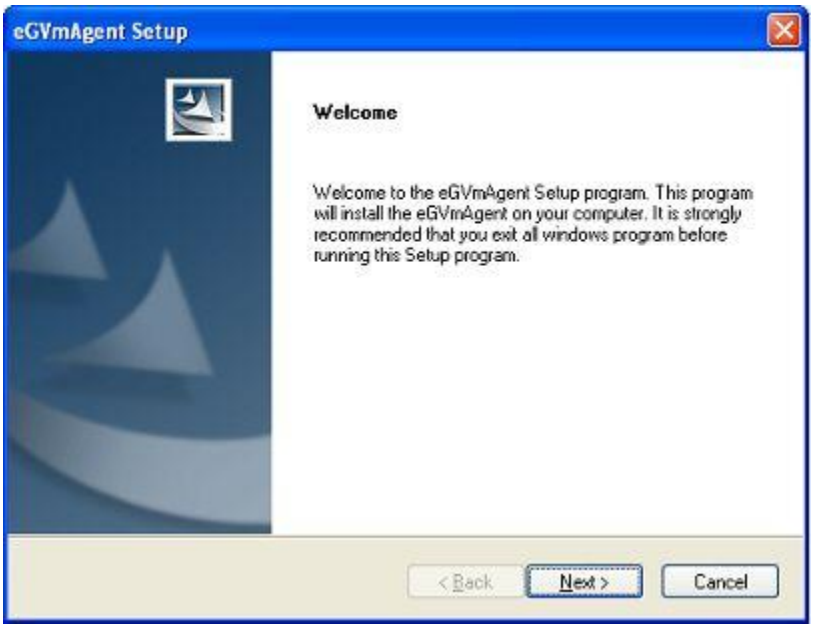


Figure 2.1: Welcome screen of the eG VM Agent installation wizard

3. When Figure 2.2 appears, click on **Yes** to accept the displayed license agreement.

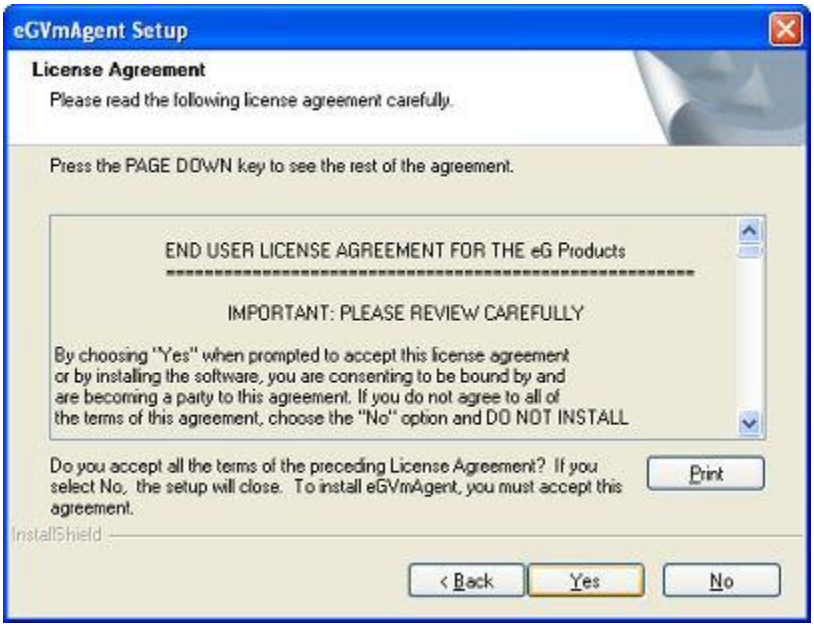


Figure 2.2: Accepting the license agreement

4. Use the **Browse** button in Figure 2.3 to indicate the location in which the agent should be installed, and click the **Next** button to proceed.

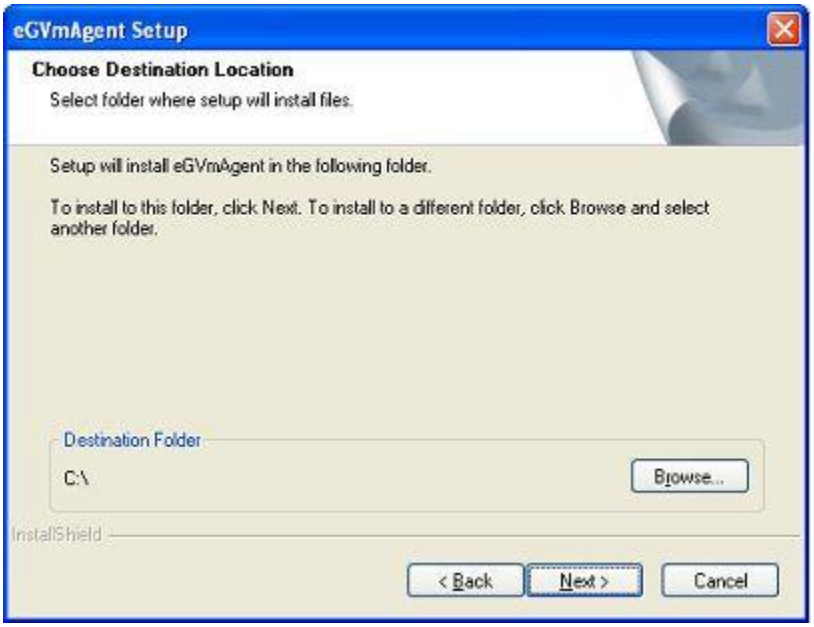


Figure 2.3: Specifying the install directory of the eG VM Agent

5. Next, specify the port at which the VM agent listens for requests from the eG agent. The default port is 60001. After port specification, click on the **Next** button in Figure 2.4 to proceed.

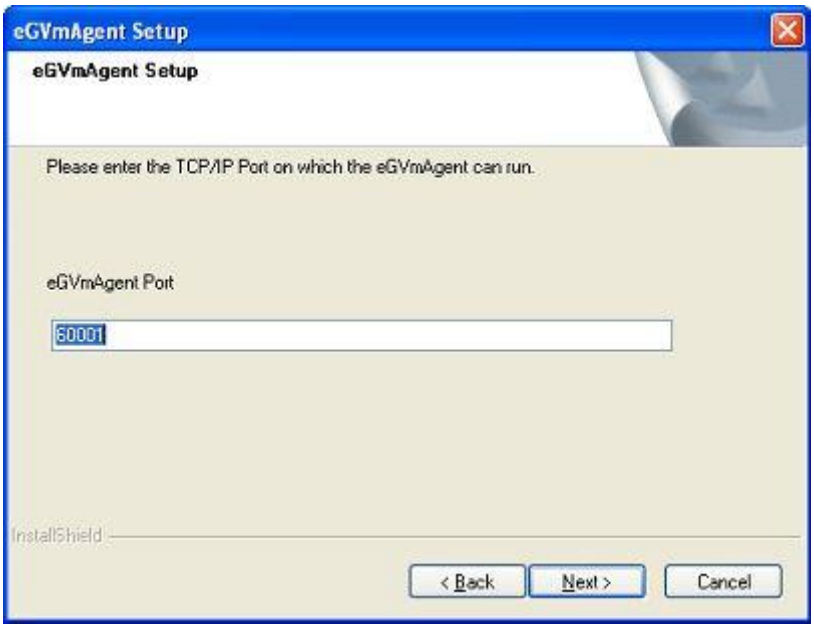


Figure 2.4: Specifying the VM agent port

6. A summary of your specifications then follows (see Figure 2.5). Click **Next** to proceed.

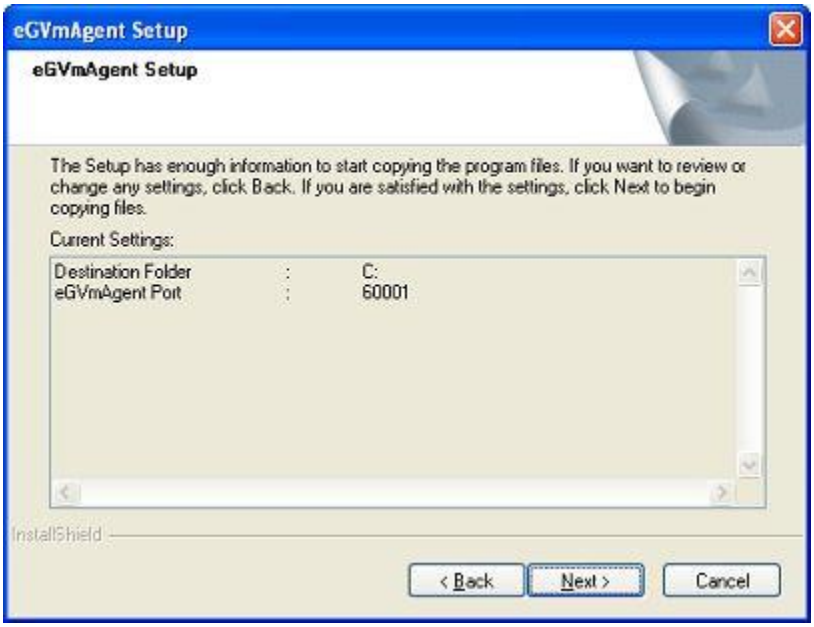


Figure 2.5: A summary of your specifications

7. Finally, click the **Finish** button in Figure 2.6 to complete the installation.

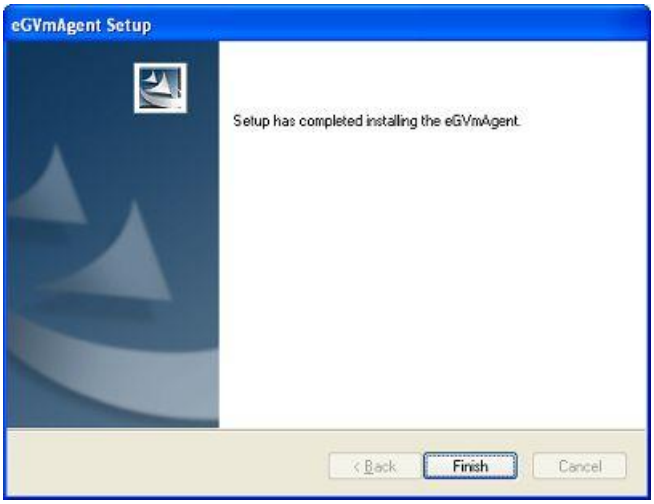


Figure 2.6: Finishing the installation

2.5 Communication between the eG Agent and the eG VM Agent

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

Upon successful installation, the eG VM agent starts automatically and begins listening for requests at default TCP port 60001. However, if, during the installation process, you have configured a different port for the eG VM agent,

then, after completing the installation, follow the steps below to make sure that the eG agent communicates with the eG VM agent via the port that you have configured:

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

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

2.5.1 Licensing of the eG VM Agent

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

2.5.2 Benefits of the eG VM Agent

The eG VM Agent offers several key benefits:

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

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

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

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

- The **ADMIN$** share will have to be available on the Windows guests
- The Windows Firewall should be configured to allow Windows File and Print Sharing

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

2.6.1 Enabling ADMIN$ Share Access on Windows Virtual Guests

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

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

8. Open the Windows Explorer on the virtual machine, browse for the corresponding **Windows** directory in the C drive, right-click on it, and select the **Sharing** option from the shortcut menu.
9. If the **ADMIN$** share does not pre-exist on the Windows guest, then Figure 2.7 appears indicating the same.

Figure 2.7: The ADMIN$ share does not exist

On the other hand, if the **ADMIN$** share pre-exists, Figure 2.8 appears. In such a case, first, remove the **ADMIN$** share by selecting the **Do not share this folder** option from Figure 2.8 and clicking the **Apply** and **OK** buttons. After this, you will have to repeat step 1 of this procedure to open Figure 2.7. Then, proceed as indicated by step 3 onwards.

Figure 2.8: Admin$ share pre-exists

10. To create (or re-create) the **ADMIN$** share, select the **Share this folder** option from Figure 2.9, and provide **ADMIN$** share against the **Share name** text box (see Figure 2.9).

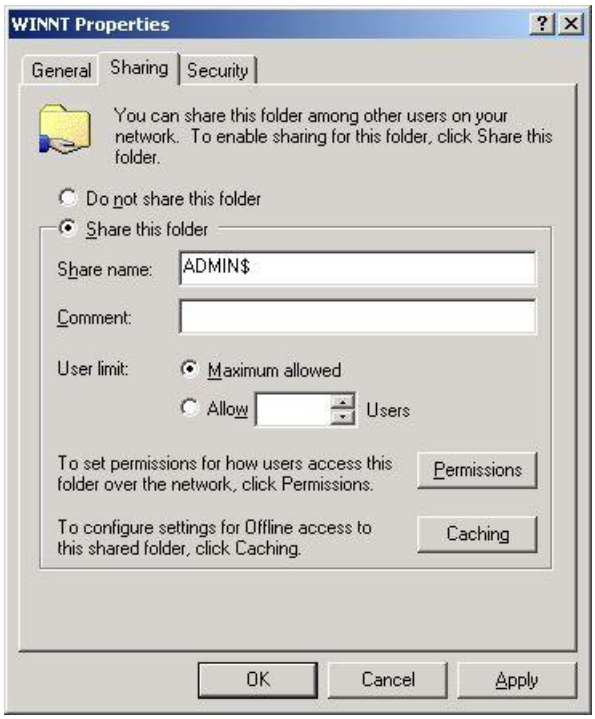


Figure 2.9: Creating the ADMIN$ share

11. Next, to enable the eG agent to communicate effectively with the Windows guest, you need to ensure that the permission to access the **ADMIN$** share is granted to an administrative user (local/domain); also, the **credentials of this user should be passed while configuring the eG monitoring capabilities** - i.e., while configuring the VMware tests. To grant the access permissions, click on the **Permissions** button in Figure 2.9.

12. By default, the **ADMIN$** share can be accessed by **Everyone** (see Figure 2.10). To grant access rights to a specific administrative (local/domain) user, select the **Add** button in Figure 2.10. When Figure 2.11 appears, select the domain to search from the **Look in** list. The valid user accounts configured on the chosen domain then appear in the box below. From this box, choose the administrator's account and click on the **Add** button to add the chosen user account to the box below the **Add** button.

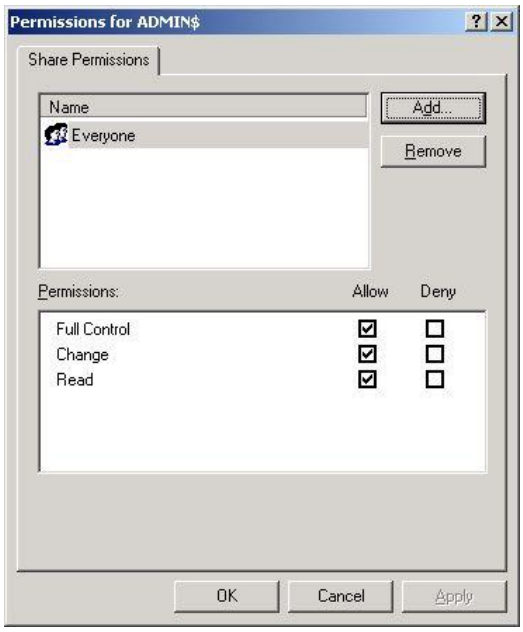


Figure 2.10: Clicking the Add button

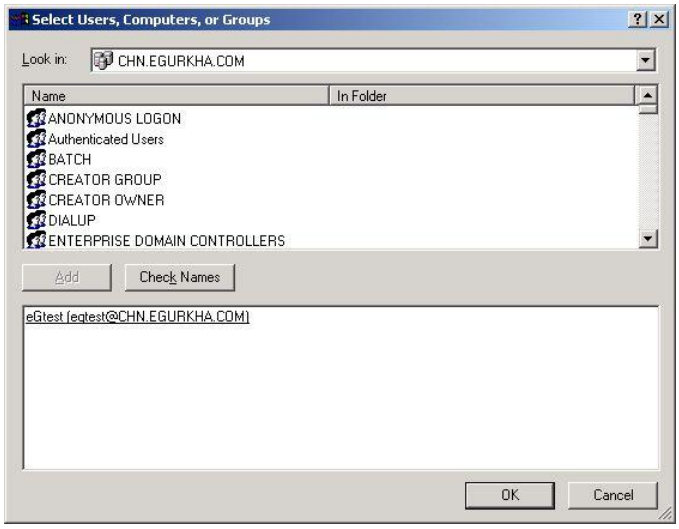


Figure 2.11: Selecting the administrative user to whom access rights are to be granted

13. Finally, click the **OK** button. You will then switch to Figure 2.12, where the newly added administrator account will appear.

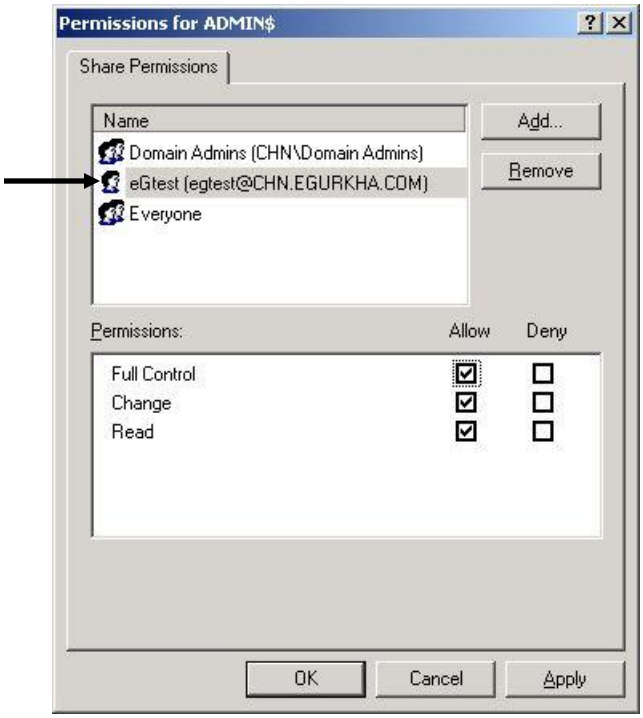


Figure 2.12: The administrator account granted access permissions

14. Select the newly added administrator account from Figure 2.12, and then, using the **Permissions** section, grant the administrator **Full Control**, **Change**, and **Read** permissions.
15. Finally, click the **Apply** and **OK** buttons in Figure 2.12 to register the changes.
16. Once you return to Figure 2.9, click on the **Security** tab to define the security settings for the **ADMIN$** share (see Figure 2.13).

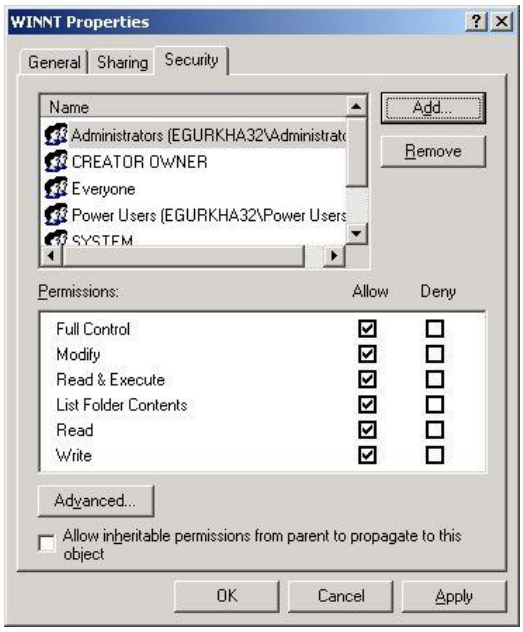


Figure 2.13: Defining the Security settings for the ADMIN$ share

17. Here again, you need to add the same administrator account, which was granted access permissions earlier. To do so, click the **Add** button in Figure 2.13, pick a domain from the **Look in** list of Figure 2.14, select the said administrator account from the domain users list below, and click the **Add** button (in Figure 2.14) to add the chosen account. Then, click the **OK** button in Figure 2.14.

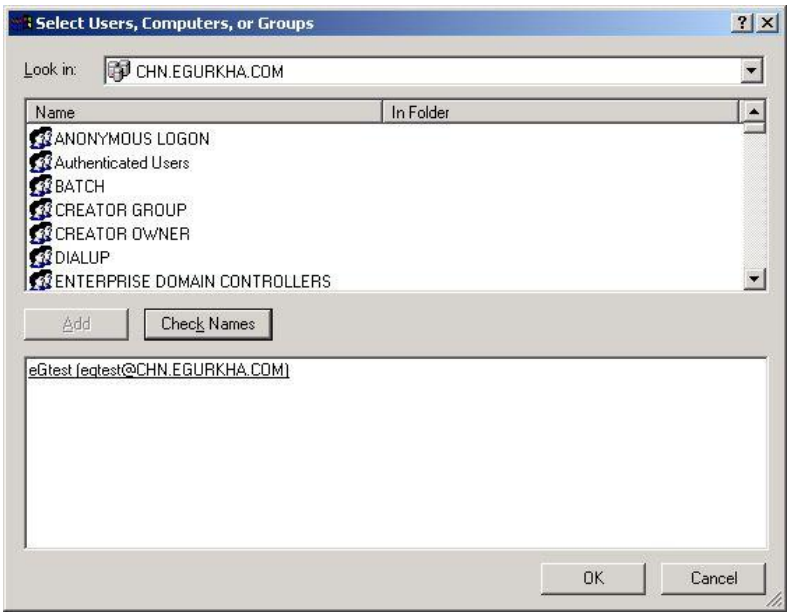


Figure 2.14: Adding the administrator account

18. This will bring you back to Figure 2.13, but this time, the newly added domain administrator account will be listed therein as indicated by Figure 2.15.

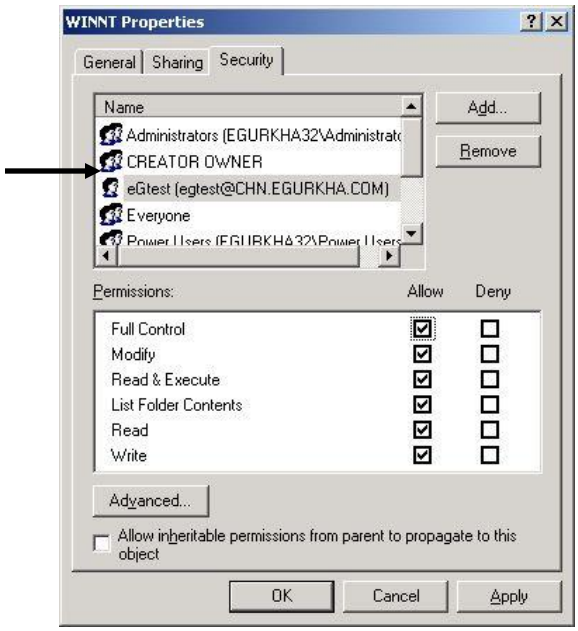


Figure 2.15: The Administrator account in the Security list

19. Finally, click the **Apply** and **OK** buttons in Figure 2.15.

2.6.1.2 Enabling ADMIN$ Share Access on Windows 2008 VMs

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

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

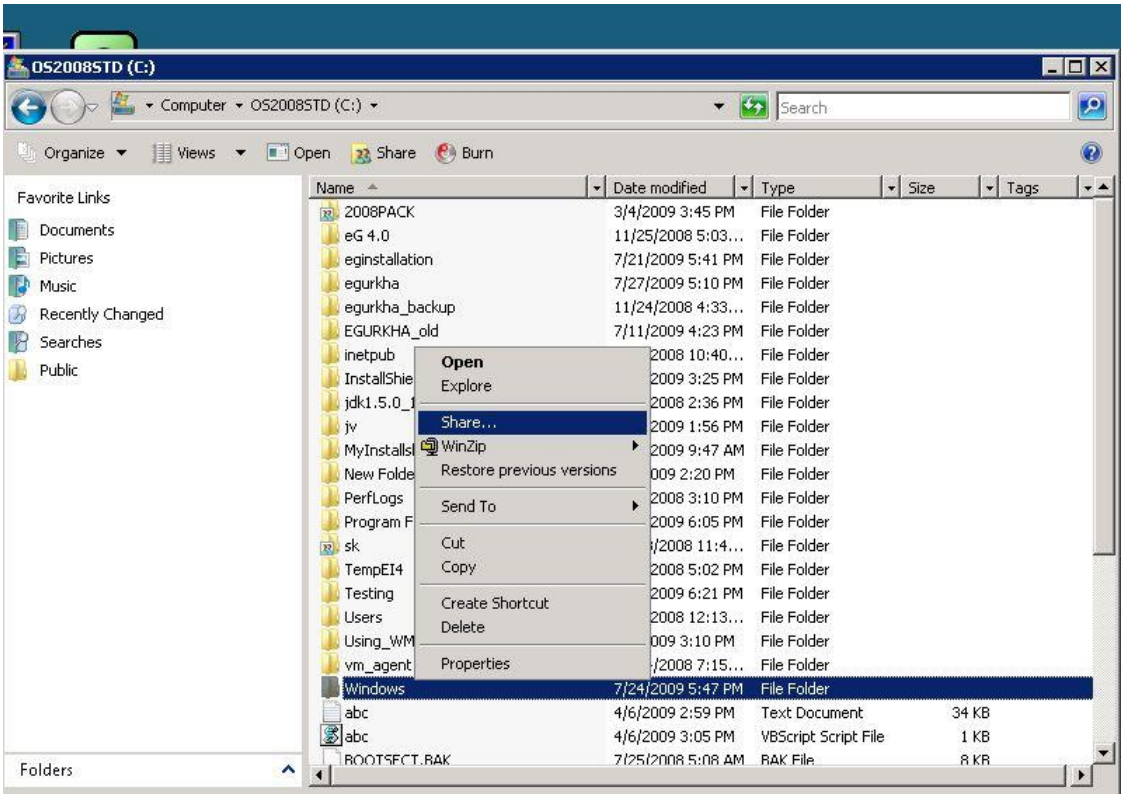


Figure 2.16: Selecting the Share option from the shortcut menu

21. Figure 2.17 will then appear. Click on **Advanced Sharing** in Figure 2.17.

Figure 2.17: Clicking on Advanced Sharing

22. Select the **Share this folder** check box in Figure 2.18 that appears, enter **ADMIN$** against **Share name**, and click on the **Permissions** button in Figure 2.18, to allow only a local/domain administrator to access the folder.

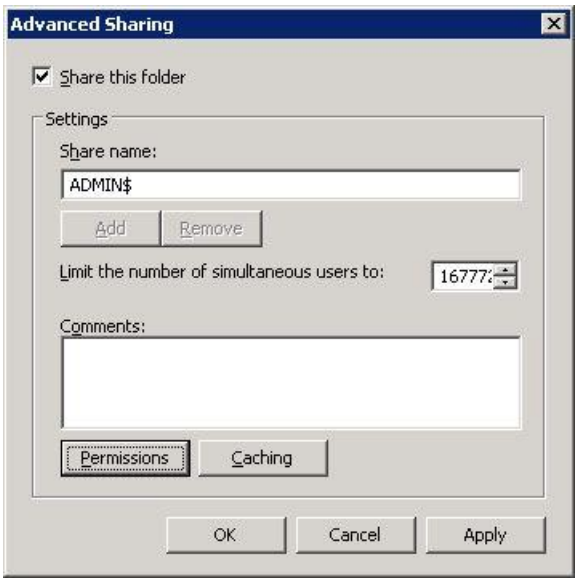


Figure 2.18: Enabling the ADMIN$ share

23. When Figure 2.19 appears, click on the **Add** button therein.

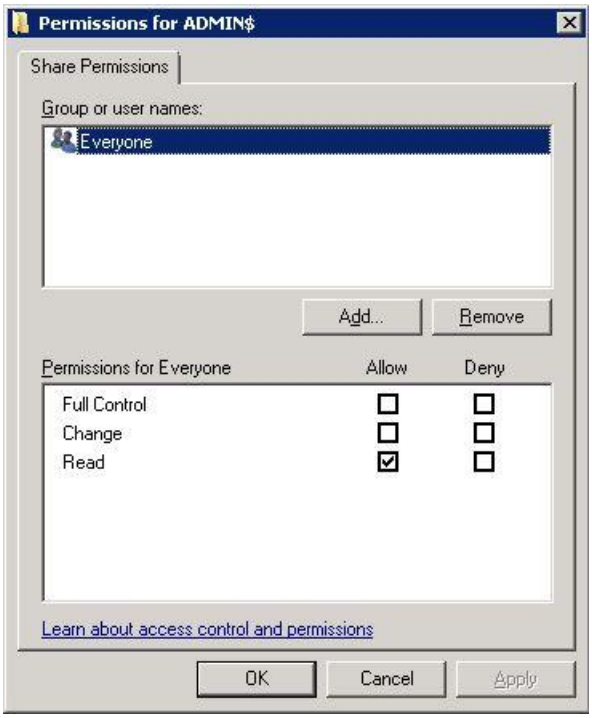


Figure 2.19: Clicking on the Add button

24. To allow a domain administrator to access the folder, first, ensure that a valid domain is specified in the **From this location** box of Figure 2.20. If you want to grant access to a local administrator instead, ensure that the name of the local host is displayed in the **From this location** box. To change this specification, use the **Locations** button in Figure 2.20. Then, enter the name of the local/domain administrator in the **Enter the object names to select** text area, and click the **OK** button.

Figure 2.20: Allowing a domain administrator to access the folder

25. The newly added user will be listed in the **Group or user names** section, as depicted by Figure 2.21. Select this user, and then, check all the three check boxes under **Allow** in the **Permissions for <user>** section in Figure 2.21. Then, click the **Apply** and **OK** buttons therein.

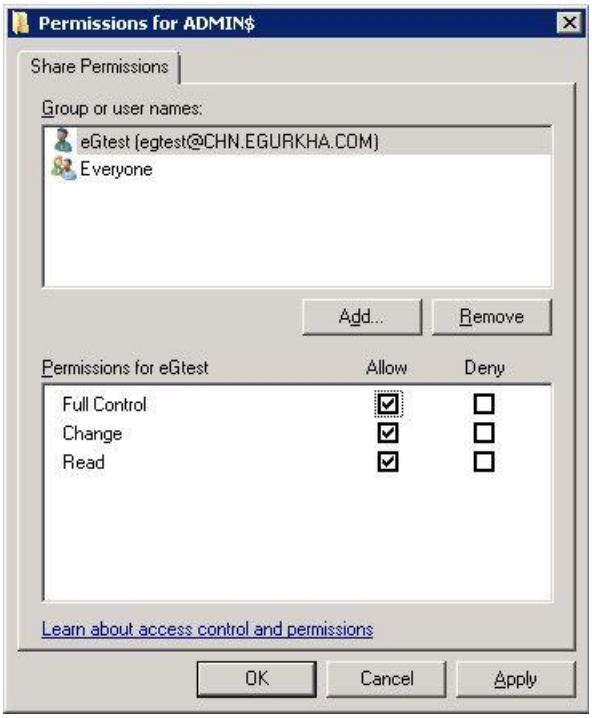


Figure 2.21: Allowing full access to the local/domain administrator

26. When Figure 2.22 appears, click on the **Apply** and **OK** buttons therein to register the changes.

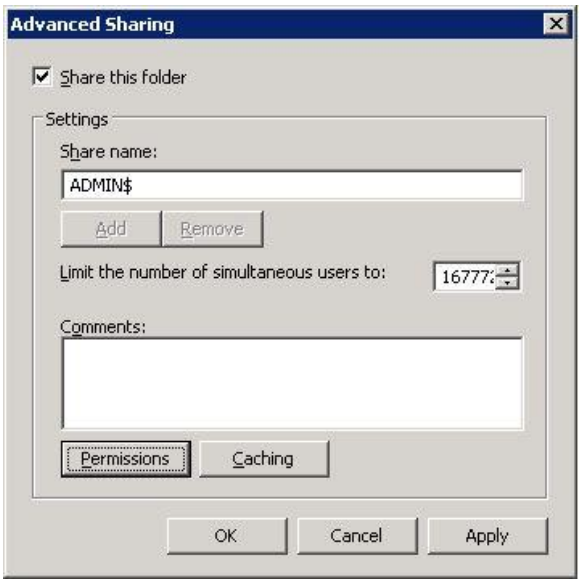


Figure 2.22: Applying the changes

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

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

      HKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\WINDOWS\CURRENTVERSION\POLICIES\SYSTEM

- Create a new entry with the below information

  Key Name : LocalAccountTokenFilterPolicy

  Key Type : DWORD (32-bit)

  Key Value : 1

- Exit registry editor.

Note

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

2.6.2 Configuring Windows Firewalls to Allow File and Print Sharing

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

To achieve this, do the following:

1. Open the Virtual Infrastructure Client console, and from the tree-structure in its left pane, select the guest OS (Windows XP/Windows 2003/Windows Vista/Windows 2008/Windows 7) on which the firewall should be configured (see Figure 2.23).

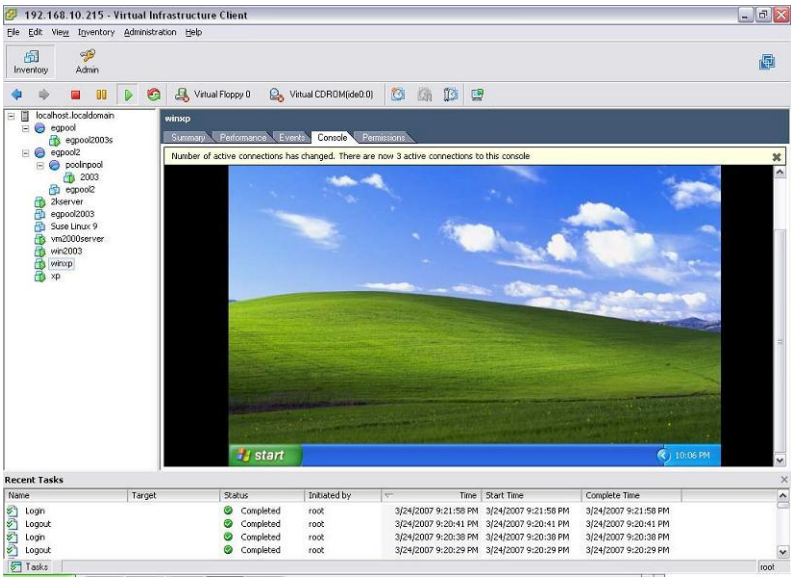


Figure 2.23: Selecting the guest OS

2. Follow the menu sequence: Start -> All Programs -> Control Panel (see Figure 2.23), and then double-click on the **Windows Firewall** option within.

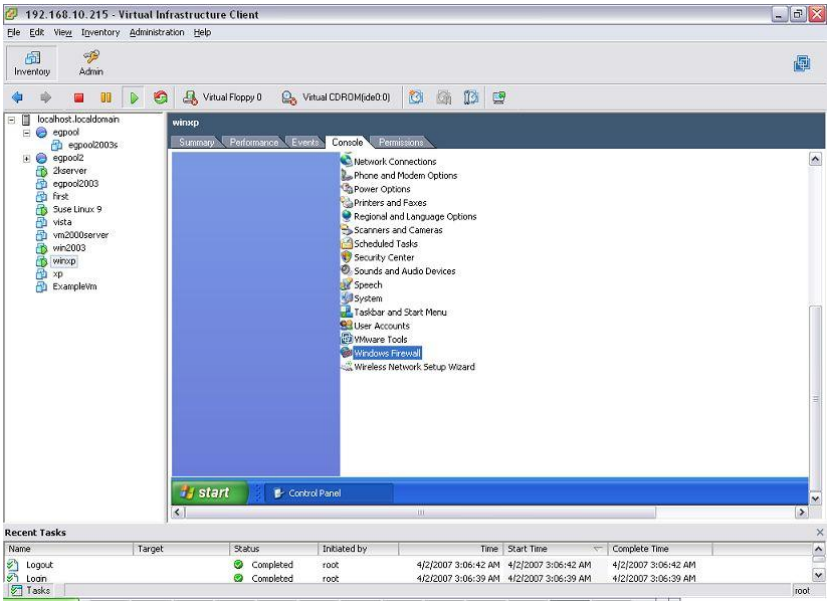


Figure 2.24: Opening the Windows Firewall

3. Figure 2.24 then appears, with the **General** tab selected by default.

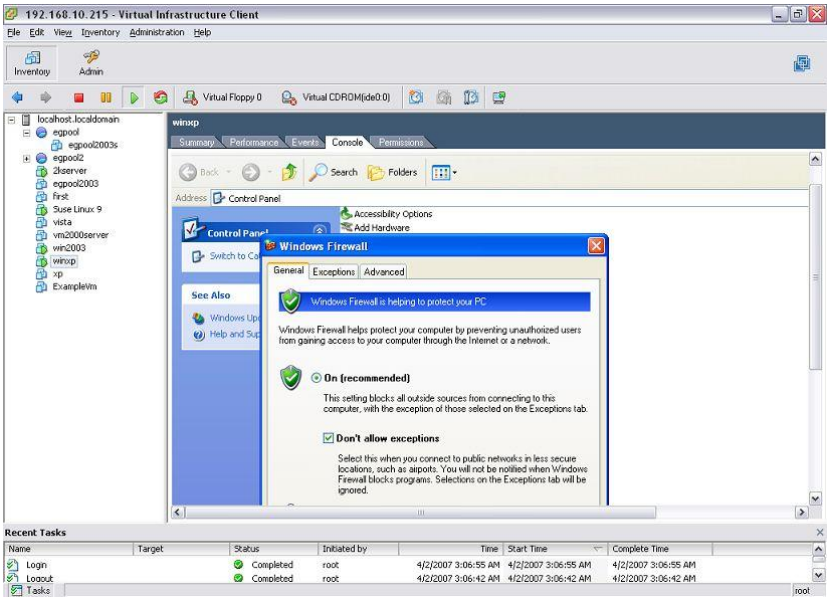


Figure 2.25: The General tab of the Windows Firewall dialog box

4. Deselect the **Don't allow exceptions** check box as indicated by Figure 2.25.

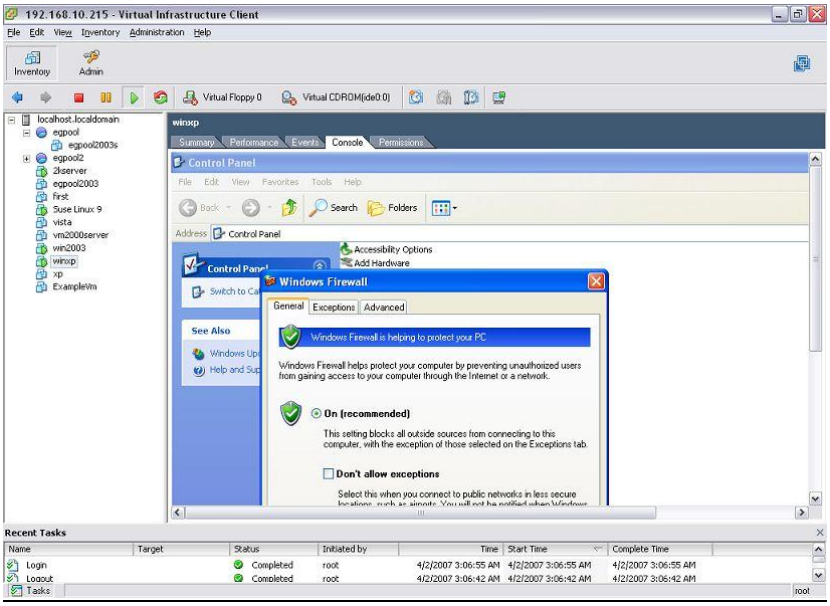


Figure 2.26: Deselecting the 'Don't allow exceptions' check box

5. Next, click on the **Exceptions** tab, and ensure that the **File and Printer Sharing** option is enabled (see Figure 2.26).

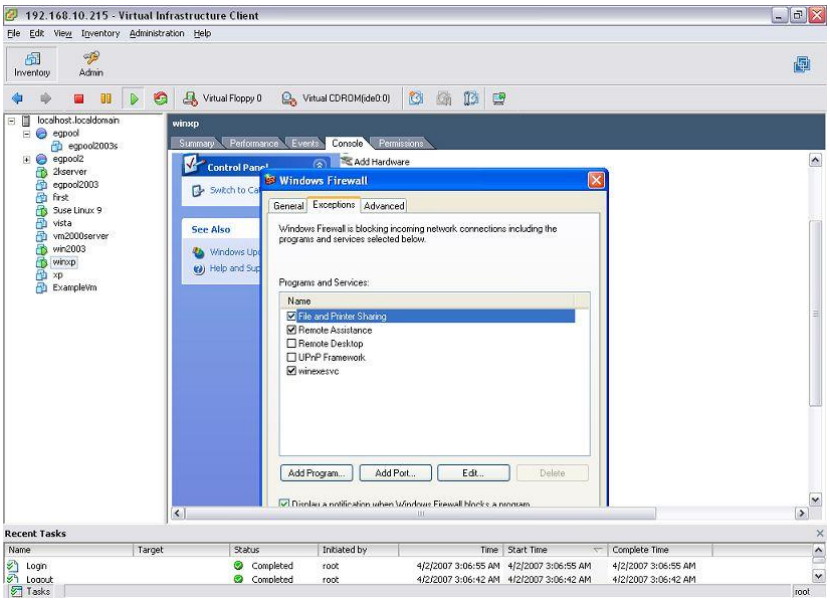


Figure 2.27: Enabling 'File and Printer Sharing'

6. Then, click the **Edit** button in Figure 2.27 to open the ports required for the agent-guest communication. Ensure that at least one of the listed TCP ports are enabled.

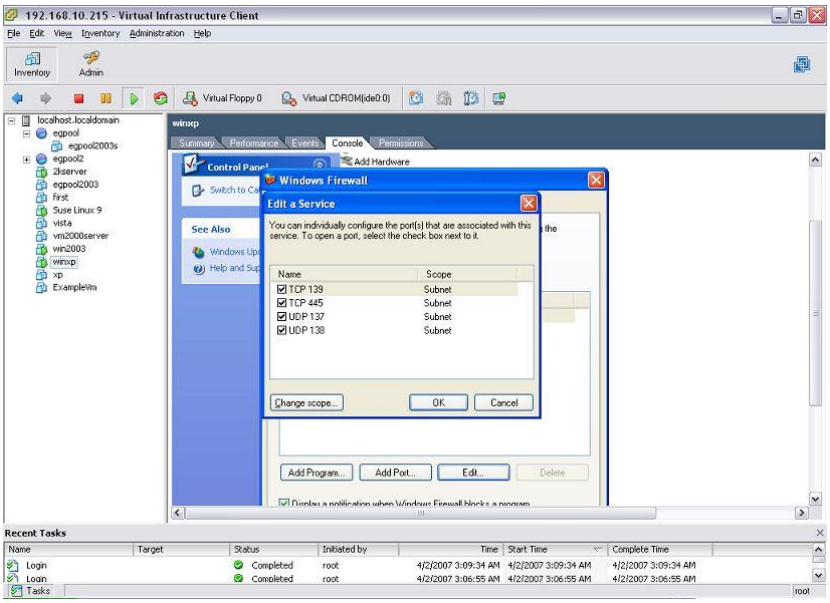


Figure 2.28: Opening ports

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

2.7 Monitoring Models offered by eG Enterprise for KVM Infrastructures

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

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

3 Monitoring KVM servers

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

Figure 3.1: The layer model of the KVM Server

The **TCP** and **Application Layers** have already been discussed in detail in the *Monitoring Unix and Windows servers* document. Therefore, let us now discuss the other layers of Figure 3.1 in detail.

3.1 The Operating System Layer

Using the tests mapped to this layer, administrators can determine the following:

- The number of virtual components such as sockets, cores, threads etc
- The number of virtual CPUs
- The percentage of physical memory used by the KVM host at its base
- The current status of each storage pool and the percentage of space utilized by each storage pool
- The current status of each storage volume and the percentage of space utilized by each storage volume

Since most of the tests of the **Operating System** layer have been already discussed in the *Monitoring Unix and Windows servers* document, let us now discuss the tests that pertain to the KVM server alone.

Figure 3.2: The tests mapped to the Operating System layer

3.1.1 Host Details - KVM Test

This test provides the administrators with a series of configuration measures such as the number of sockets, cores, nodes and threads. Additionally, this test focuses on the number of virtual CPUs and the CPUs that are active on the KVM server host. This way, this test enables the administrators to judge their KVM server infrastructure and load the virtual machines on the server accordingly!

Purpose	Reports a series of configuration measures such as the number of sockets, cores, nodes and threads. Additionally, this test focuses on the number of virtual CPUs and the CPUs that are active on the KVM server host. This way, this test enables the administrators to judge their KVM server infrastructure and load the virtual machines on the server accordingly!		
Target of the test	A KVM server		
Agent deploying the test	An internal agent		
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		
ll Outputs of the test	One set of results for the KVM server that is to be monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	CPU sockets: Indicates the total number of CPU sockets that are available on this server.	Number	A high value for this measure indicates that the administrators may still load additional virtual machines on the KVM server.

	Cores per socket: Indicates the total number of cores available in each socket of this server.	Number	
	Active CPUs: Indicates the number of CPUs that are currently active on this server.	Number	A high value is desired for this measure.
	Virtual CPUs: Indicates the number of virtual CPUs that are available on this server.	Number	A high value for this measure indicates that the virtual machines are adequately allocated with memory resources.
	CPU frequency: Indicates the average frequency at which the CPUs are operating on this server.	Mhz	
	Numa cells: Indicates the number of nodes i.e., NUMA cells available on this server.	Number	Non-uniform memory access (NUMA) is a memory architecture implemented in multi-processor systems where the memory is spread among processors instead of being central like in standard Symmetric Processing Servers (SMP). This design reduces the risk of bottleneck on the links between CPUs and memory. A guest virtual machine on a NUMA system can be pinned to a processing core so that its memory allocations are always local to the node it is running on. This avoids cross-node memory transports which have less bandwidth and can significantly degrade performance. A high value is desired for this measure.
	Threads per core: Indicates the average number of threads that are available per core on this server.	Number	

3.1.2 Memory - KVM Test

This test reports the memory usage of the KVM server and enables administrators to identify whether/not adequate memory resources are available on the KVM server host for use by the virtual machines.

Purpose	Reports the memory usage of the KVM server and enables administrators to identify whether/not adequate memory resources are available on the KVM server host for use by the virtual machines.
Target of the test	A KVM server
Agent deploying the	An internal agent

test			
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG manager license should allow the detailed diagnosis capability; ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the KVM server that is to be monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Total physical memory: Indicates the total amount of physical memory that is allocated for this KVM server.	MB	
	Total configured memory: Indicates the total amount of memory that is configured for this KVM server.	MB	
	Used physical memory: Indicates the amount of physical memory that is currently used by the KVM server.	MB	Ideally, the value of this measure should be low. The detailed diagnosis of this measure if enabled, lists the VMs and their current memory usage.
	Free physical memory: Indicates the amount of physical memory that is currently available for use in this KVM server.	MB	A high value is desired for this measure.

	Percentage of physical memory used: Indicates the percentage of physical memory that is utilized by this KVM server.	Percent	A very high value for this measure indicates a shortage of memory resources. If more memory is not made available soon, then this could significantly degrade the performance of the virtual machines hosted on this server. The detailed diagnosis of this measure if enabled, lists the top 10 memory consuming processes, the PIDs of the processes and the memory utilized by each process.

The detailed diagnosis of the *Used physical memory* measure if enabled, lists the VMs and their current memory usage. This way administrators can identify the VM that is utilizing the maximum memory resources.

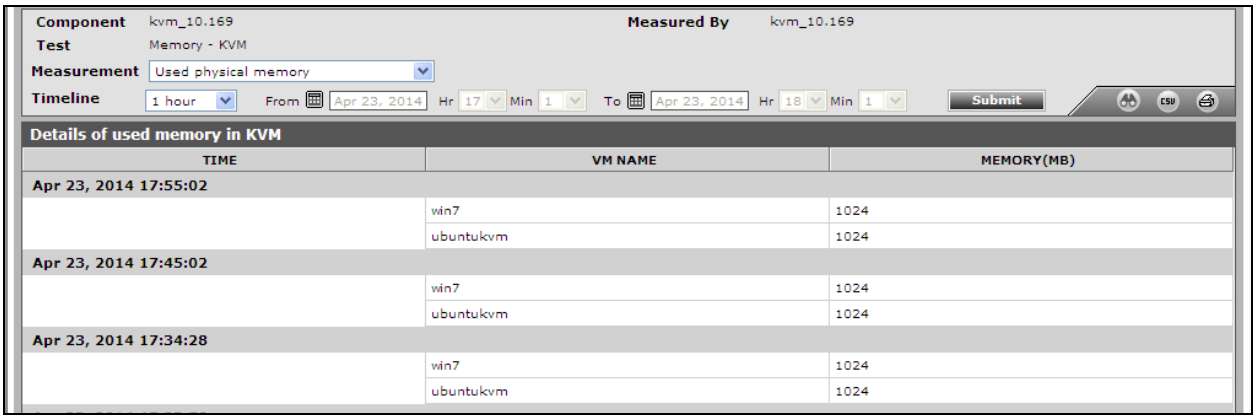


Figure 3.3: The detailed diagnosis of the Used physical memory measure

The detailed diagnosis of this measure if enabled, lists the top 10 memory consuming processes, the PIDs of the processes and the memory utilized by each process. By looking at the detailed diagnosis, administrators can instantly identify the process that is consuming too much memory resources in the server and take remedial actions immediately.

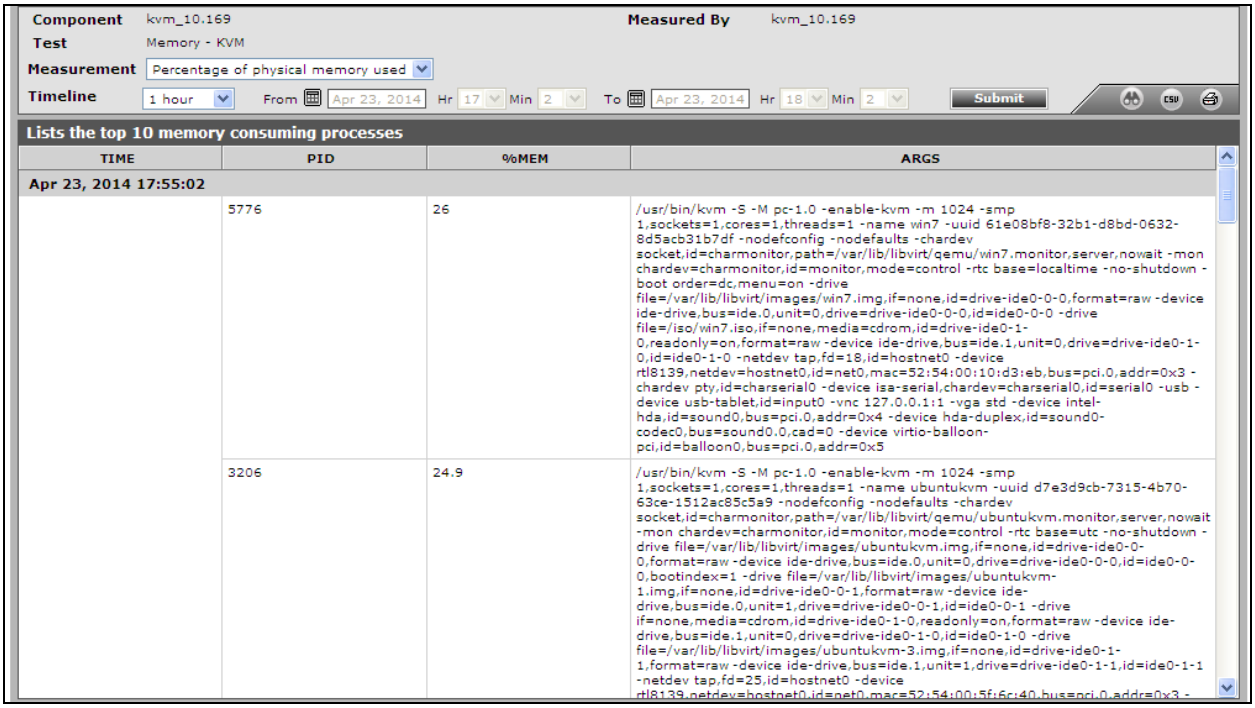


Figure 3.4: The detailed diagnosis of the Percentage of physical memory used measure

3.1.3 Storage Pools Test

A Storage pool is a logical storage group containing one or more volumes, which are virtual disks in all kinds of formats.

This test auto discovers the storage pools of the KVM server and reports the current status of each storage pool. In addition, this test provides you in detail the space utilization of each storage pool. This way, administrators are proactively alerted to potential space crunches in the storage pools of the KVM server.

Purpose	Auto discovers the storage pools of the KVM server and reports the current status of each storage pool. In addition, this test provides you in detail the space utilization of each storage pool. This way, administrators are proactively alerted to potential space crunches in the storage pools of the KVM server.		
Target of the test	A KVM server		
Agent deploying the test	An internal agent		
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		
Outputs of the test	One set of results for the KVM server that is to be monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr><td rowspan="2">test</td><td>Storage pool status:
Indicates the current status of this storage pool.</td><td></td><td>The numeric values that correspond to each of the Measure Values that this test can take are listed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Unknown</td><td>0</td></tr><tr><td>Normal</td><td>1</td></tr><tr><td>Warning</td><td>2</td></tr><tr><td>Error</td><td>3</td></tr></table>Note:
By default, this measure reports one of the Measure Values listed in the table above. The graph of this measure however, represents the status of this storage pool using the numeric equivalents - ‘0’ to ‘3’.</td></tr>
<tr><td>Storage pool capacity:
Indicates the total capacity of this storage pool.</td><td>MB</td><td></td></tr>
<tr><td></td><td>Space used in pool:
Indicates the amount of space that is already utilized in this storage pool.</td><td>MB</td><td>Ideally, the value of this measure should be low.</td></tr>
<tr><td></td><td>Free space in pool:
Indicates the amount of space that is currently available for use in this storage pool.</td><td>MB</td><td>A high value is desired for this measure.</td></tr>
<tr><td></td><td>Percent usage of space in pool:
Indicates the percentage of space that is utilized by this storage pool.</td><td>Percent</td><td>A very high value for this measure indicates a space crunch in the storage pool. If adequate resources are not made available soon, then this could significantly degrade the performance of the virtual machines that share the resources from the storage pools that are in short of adequate resources.</td></tr>
</table>

3.1.4 Storage Volumes Test

This test auto discovers each storage volume of the storage pools available in the KVM server and reports the type of each storage volume and how well each volume utilizes the space allocated to it. This way, administrators are proactively alerted to potential space crunches in the storage volumes.

Purpose	Auto discovers each storage volume of the storage pools available in the KVM server and reports the type of each storage volume and how well each volume utilizes the space allocated to it. This way, administrators are proactively alerted to potential space crunches in the storage

<table>
<tr><td></td><td colspan="3">volumes.</td></tr>
<tr><td>Target of the test</td><td colspan="3">A KVM server</td></tr>
<tr><td>Agent deploying the test</td><td colspan="3">An internal agent</td></tr>
<tr><td>Configurable parameters for the test</td><td colspan="3">1. TEST PERIOD - How often should the test be executed
2. HOST – The host for which the test is to be configured</td></tr>
<tr><td>Outputs of the test</td><td colspan="3">One set of results for the KVM server that is to be monitored</td></tr>
<tr><td rowspan="4">Measurements made by the test</td><td>Measurement</td><td>Measurement Unit</td><td>Interpretation</td></tr>
<tr><td>Volume type:
Indicates the type of this storage volume.</td><td></td><td>The storage volume can either be Block based or File based.
The numeric values that correspond to each of the volume types discussed above are listed in the table below:<table><tr><th>Type</th><th>Numeric Value</th></tr><tr><td>Block based</td><td>0</td></tr><tr><td>File based</td><td>1</td></tr></table>Note:
By default, this measure reports one of the Types listed in the table above. The graph of this measure however, represents the type of this storage volume using the numeric equivalents - ‘0’ to ‘1’.</td></tr>
<tr><td>Volume capacity:
Indicates the total capacity of this storage volume.</td><td>MB</td><td></td></tr>
<tr><td></td><td>Space used in volume:
Indicates the amount of space that is already utilized in this storage volume.</td><td>MB</td><td>Ideally, the value of this measure should be low.</td></tr>
<tr><td></td><td>Free space in volume:
Indicates the amount of space that is currently available for use in this storage volume.</td><td>MB</td><td>A high value is desired for this measure.</td></tr>
</table>

	Percent usage of space in volume: Indicates the percentage of space that is utilized by this storage volume.	Percent	A very high value for this measure indicates a space crunch in the storage volume. If adequate resources are not made available soon, then this could significantly degrade the performance of the virtual machines that share the resources from the storage volumes that are in short of adequate resources.

3.2 The Network Layer

This layer tracks the network connectivity and traffic to and fro from each network interfaces of the KVM server host. This layer also tracks the virtual networks of the KVM server and reports if each virtual network is active or not.

The **Network** test and **Network Traffic** test depicted by Figure 3.5 has already been discussed in the *Monitoring Unix and Windows Servers* document. The following section will hence discuss the **Virtual Networks** test only.

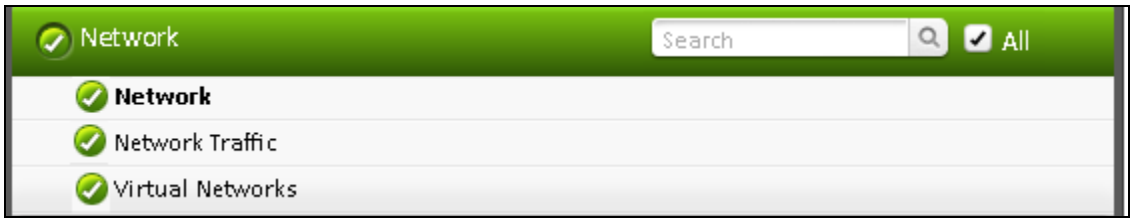


Figure 3.5: The tests mapped to the Network layer

3.2.1 Virtual Networks Test

This test auto discovers the virtual networks available in the KVM server and reports whether/not each virtual network is active over the network.

Purpose	Auto discovers the virtual networks available in the KVM server and reports whether/not each virtual network is active over the network.		
Target of the test	A KVM server		
Agent deploying the test	An internal agent.		
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		
Outputs of the test	One set of results for each virtual network that is to be monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Is active?:
Indicates the current status of this virtual network i.e., indicates whether this virtual network is active over the network or not.</td>
<td>Percent</td>
<td>This test reports a value Yes if the virtual network is active and No if otherwise. This test also reports Error if the virtual networks encountered errros.

The numeric values that correspond to each of the Measure Values reported by this test are listed in the table 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>2</td></tr>
<tr><td>Error</td><td>3</td></tr>
</table>
Note:

By default, this measure reports one of the Measure values listed in the table above. The graph of this measure however, represents the current status of this virtual network using the numeric equivalents - ‘1’ to ‘3’.</td>
</tr>
</table>

3.3 The Outside View of VMs Layer

This layer provides the host operating system’s view of the resource usage levels of each of the virtual guests hosted on it. Using the information reported by this test, administrators can:

- Determine which of the guests is taking up more resources (CPU, memory, network, or disk) than the others. This information can help with load balancing or capacity planning. For example, if one of the guests is receiving a very high rate of requests compared to the others, this guest may be a candidate for migration to another KVM server, so as to minimize the impact it has on the other guests on the current ESX server.
- Determine times when sudden or steady spikes in the physical resource utilization are caused by the guest machines
- Track the overall status of the virtual machines - how many are registered, which ones are powered on, and at what times, etc.
- Understand how resources are shared amongst all available resource pools, and identify resource pools that have been over-utilized.

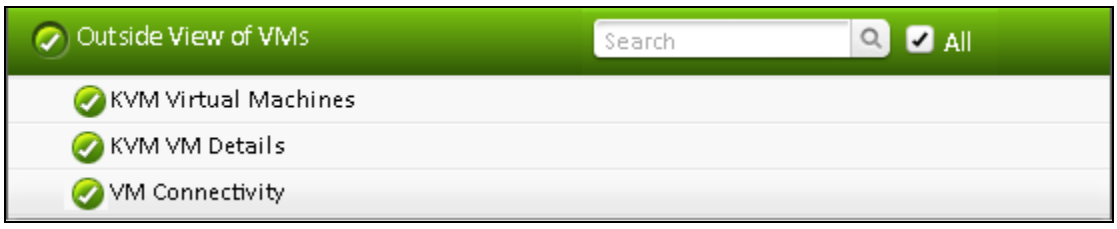


Figure 3.6: The tests mapped to the Outside View of VMs layer

3.3.1 KVM Virtual Machines Test

Whenever users complain of inaccessibility of their virtual machines, administrators need to promptly determine the reason for the same - is it because the VMs are not running currently? is it because they are not even registered? or is it because they have been moved to another server? The **KVM Virtual Machines** test provides administrators with this information. This test tracks the status and movement of each virtual machine on the target KVM server, reports the number and names of virtual machines in various states, and also captures the migration of virtual machines to other servers.

Purpose	Tracks the status and movement of each virtual machine on the target KVM server, reports the number and names of virtual machines in various states, and also captures the migration of virtual machines to other servers.		
Target of the test	A KVM server		
Agent deploying the test	An internal agent.		
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 4. **DETAILED DIAGNOSIS** – To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG manager license should allow the detailed diagnosis capability; ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for the KVM server that is to be monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Registered VMs: Indicates the total number of virtual machines that have been registered with this KVM server.	Number	The detailed diagnosis of this measure if enabled, lists each registered virtual machine, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.

	Running VMs: Indicates the total number of virtual machines that are currently running.	Number	A high value is desired for this measure. The detailed diagnosis of this measure if enabled, lists each virtual machine that is currently running, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	VMs not running: Indicates the number of virtual machines that are currently not running.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that is currently not running, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	Resource blocked VMs: Indicates the number of virtual machines that are currently blocked.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that is blocked, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	Crashed VMs: Indicates the number of virtual machines that currently crashed.	Number	Ideally, the value of this measure should be 0. The detailed diagnosis of this measure if enabled, lists each virtual machine that crashed, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	No state VMs: Indicates the number of virtual machines that are currently holding the status 'No state'.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that with the 'No state' status, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	Paused VMs: Indicates the number of virtual machines that are currently paused.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that is currently paused, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	Added VMs: Indicates the number of virtual machines that were newly added to the KVM server.	Number	This measure is a good indicator of the load on the KVM Server. The detailed diagnosis of this measure if enabled, lists each virtual machine that is added, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.

	Removed VMs: Indicates the number of virtual machines that were removed from the KVM server.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that is removed, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	VMs with users: Indicates the number of virtual machines on which users are currently logged in.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine, the IP address of the virtual machine, the Operating system that is loaded on the virtual machine and the user who is logged into the virtual machine. **Note that this measure will be available only for the KVM VDI server.**
	VMs without users: Indicates the number of virtual machines without any users logged in.	Number	**Note that this measure will be available only for the KVM VDI server.**

The detailed diagnosis of the *Registered VMs* measure if enabled, lists each registered virtual machine, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine. This way, administrators can quickly identify the Vms that are currently registered.

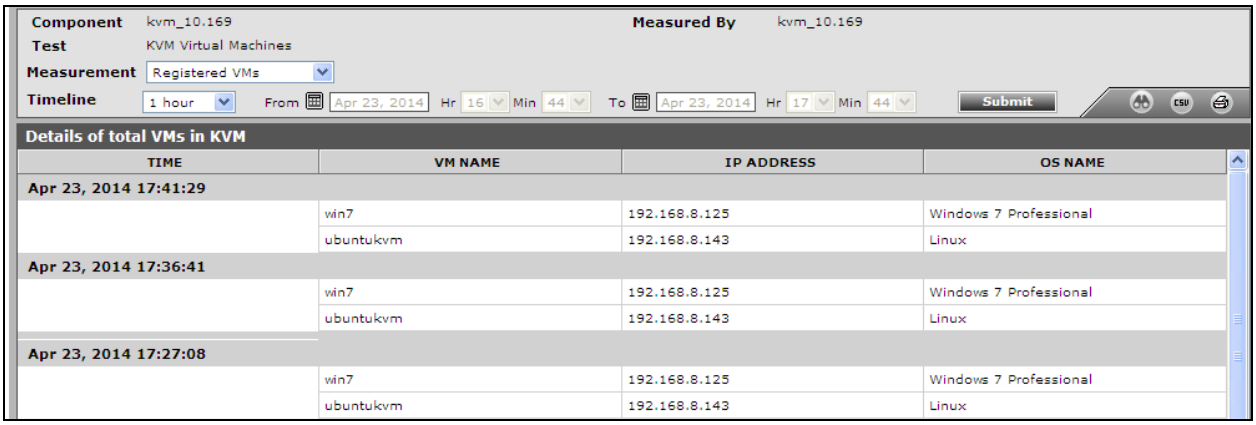


Figure 3.7: The detailed diagnosis of the Registered VMs measure

The detailed diagnosis of the *Running VMs* measure if enabled, lists each virtual machine that is currently running, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.

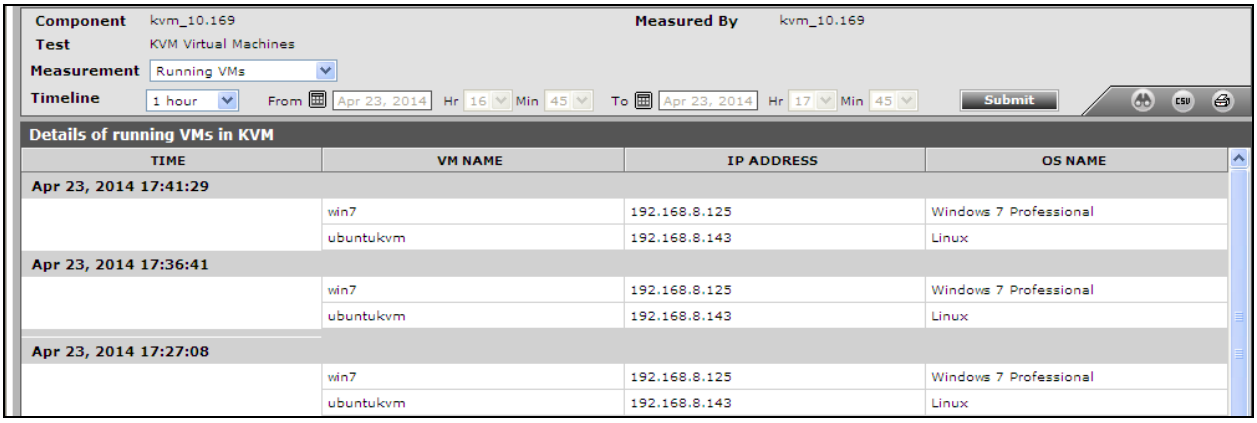


Figure 3.8: The detailed diagnosis of the Running VMs measure

3.3.2 KVM VM Details Test

This test monitors the amount of the physical server's resources that each virtual machine on a KVM server is taking up. Using the metrics reported by this test, administrators can determine which virtual machine is taking up most CPU, which virtual machine is generating the most network traffic, which virtual machine is over-utilizing memory, which virtual machine has the maximum disk activity, etc.

Purpose	Monitors the amount of the physical server's resources that each virtual machine on a KVM server is taking up. Using the metrics reported by this test, administrators can determine which virtual machine is taking up most CPU, which virtual machine is generating the most network traffic, which virtual machine is over-utilizing memory, which virtual machine has the maximum disk activity, etc.
Target of the test	A KVM server
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** – The IP address of the Sonic firewall 3. **PORT** – The port at which the specified **HOST** listens

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

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

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

Outputs of the test: One set of results for each virtual machine of the KVM server that is to be monitored

Measurements made by the test

Measurement	Measurement Unit	Interpretation
VM state: Indicates the current status of this VM.		The numeric values that correspond to each of the **Measure Value**s that this test can take are listed in the table below: **Measure Value** / **Numeric Value** Blocked: 0 Running: 1 Crashed: 2 Nostate: 3 Paused: 4 Shutdown: 5 Shutoff: 6 **Note:** By default, this measure reports one of the **Measure Value**s listed in the table above. The graph of this measure however, represents the status of each VM using the numeric equivalents - '0' to '6'.

	Is VM persistent?: Indicates whether/not the configuration of this VM is persistent.		The numeric values that correspond to each of the **Measure Value**s that this test can take are listed in the table below: **Measure Value** — **Numeric Value** Transient — 0 Yes — 1 Error — 2 **Note:** By default, this measure reports one of the **Measure Value**s listed in the table above. The graph of this measure however, represents whether the configuration of this VM is persistent or not using the numeric equivalents - '0' to '2'.
	Physical CPU utilization: Indicates the percentage of CPU utilized by this VM.	Percent	A very high value of this measure indicates that the VM is currently utilizing high memory resources.
	Virtual CPUs: Indicates the number of virtual CPUs that are allocated to this VM.	Number	
	Allocated memory: Indicates the amount of memory that is currently allocated to this VM.	MB	
	Used memory: Indicates the amount of memory that is used by this VM.	MB	A low value is desired for this measure.
	Free memory: Indicates the amount of memory that is available for use by this VM.	MB	A high value is desired for this measure. The memory that is used for reclaimable cache is not considered as free memory.
	Memory utilization: Indicates the percentage of memory that is currently utilized by this VM.	Percent	A high value for this measure indicates that the VM is currently running short of memory resources. Comparing the value of this measure across the VMs will help you identify the VM that is using the maximum memory resources.

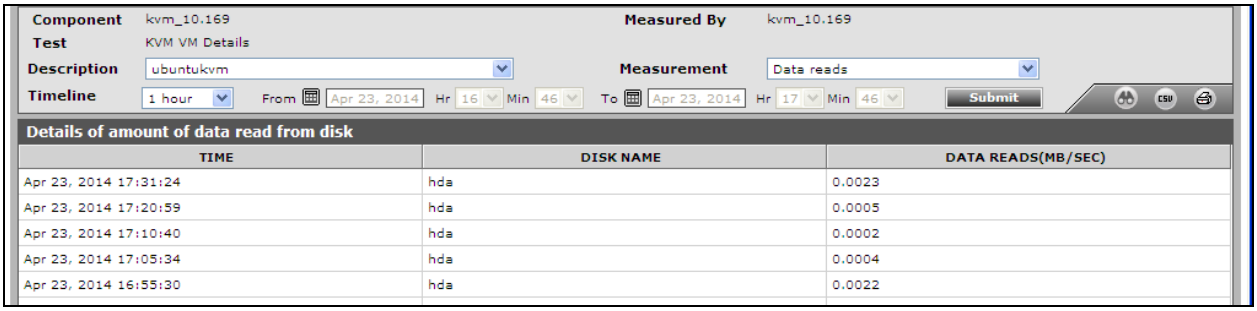
	Memory swap-in: Indicates the amount of memory that is being swapped in by the server from the disk for this VM.	MB	
	Memory swap-out: Indicates the amount of memory that is being swapped to the disk by the server for this VM.	MB	
	Page faults: Indicates the number of page faults that occurred for the threads matching all processes.	Number	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.
	Unused memory: Indicates the amount of memory that is completely left unused in this VM.	MB	The value of this measure is the sum total of the *Free memory* and the memory that is used for reclaimable caches.
	Available memory: Indicates the amount of memory that is currently available in this VM.	MB	

	Balloon memory: Indicates the amount of balloon memory that is currently available for use in this VM.	MB	Memory ballooning is a virtual memory management technique used to free unused memory. Having multiple virtual machines (VMs) on a single physical server requires virtual memory management techniques to control resource sharing and to prevent shortages. Some processor chipsets use hardware to offload a portion of the virtual memory management work by creating two layers of page tables, the data structure that provides the mapping between virtual addresses and physical addresses. The layers, however, make it difficult for the hypervisor to see a VM's memory contents, how much memory that VM requires or whether the VM is consuming too much memory. Balloon drivers, which are installed in each VM, transfer the memory shortage from the host (where the shortage exists) to the VM. The hypervisor alerts the balloon driver of low memory instances and instructs it to inflate, which locks a set of unused memory in the VM. The hypervisor can then reassign the physical memory to another VM. This swap activity can potentially impact performance depending upon the amount of memory to recoup and/or the quality of the storage IOPS delivered to the VM. In a VMware environment, the balloon driver only activates when memory becomes scarce, so it's best to have no ballooning activity at all. In a Windows Server environment, the balloon driver allocates RAM to the VM on-demand.
	RSS memory: Indicates the amount of resident memory that is allocated to the process of this VM.	MB	The resident set size is the portion of a process's memory that is held in RAM. The rest of the memory exists in swap or the filesystem (never loaded or previously unloaded parts of the executable).
	Disk errors: Indicates the number of errors that occurred during the disk reads/disk writes of this VM.	Number	Ideally, the value of this measure should be zero. Use the detailed diagnosis of this measure to figure out the nature of the errors and the disk on which the errors had occurred.

	Data reads: Indicates the rate at which data is read from the disk of this VM.	MB/sec	A high value of this measure indicates that the disk is experiencing high I/O activity. The detailed diagnosis of this measure if enabled, lists the name of the disk and the rate at which data is read from this disk.
	Read requests: Indicates the number of read requests handled by the disk of this VM.	MB/sec	The detailed diagnosis of this measure if enabled, lists the name of the disk and the number of requests handled.
	Data writes: Indicates the rate at which data is written to the disk of this VM.	MB/sec	The detailed diagnosis of this measure if enabled, lists the name of the disk and the rate at which data is written to the disk.
	Write requests: Indicates the number of write requests handled by the disk of this VM.	MB/sec	The detailed diagnosis of this measure if enabled, lists the name of the disk and the number of write requests handled by the disk.
	Data transmitted: Indicates the rate at which data is transmitted from this VM.	Mbps	A high value for this measure indicates that the data transmission is high for this VM. The detailed diagnosis of this measure if enabled, lists the name of the network interface through which data is transmitted and the rate at which data is transmitted.
	Packets transmitted: Indicates the rate at which packets are transmitted from this VM.	Packets/sec	A high value for this measure indicates that the data transmission is high for this VM. The detailed diagnosis of this measure if enabled, lists the name of the network interface through which the packets are transmitted and the rate at which the packets are transmitted.
	Data dropped during transmission: Indicates the number of data packets that were dropped during transmission.	Number	The detailed diagnosis of this measure if enabled, lists the name of the network interface that dropped the data and the number of data packets dropped.
	Errors during transmission: Indicates the number of errors encountered by this VM during transmission.	Number	Ideally, the value of this measure should be zero. The detailed diagnosis of this measure if enabled, lists the name of the network interface and the number of errors that were enocuntered.
	Data received: Indicates the rate at which data is received on this VM.	Mbps	The detailed diagnosis of this measure if enabled, lists the name of the network interface and the rate at which data was received.

	Packets received: Indicates the rate at which data packets were received by this VM.	Packets/sec	The detailed diagnosis of this measure if enabled, lists the name of the network interface and the rate at which the data packets were received.
	Data dropped during reception: Indicates the number of data packets that were dropped during reception by this VM.	Packets/sec	Ideally, the value of this measure should be zero. The detailed diagnosis of this measure if enabled, lists the name of the network interface and the number of data packets that were dropped during reception.
	Errors during reception: Indicates the number of errors encountered during data reception by this VM.	Number	Ideally, the value of this measure should be zero. The detailed diagnosis of this measure if enabled, lists the name of the network interface and the number of errors encountered during data reception.
	Allocated size: Indicates the cumulative allocated size of the disks of this VM.	MB	The detailed diagnosis of this measure if enabled, lists the name of each disk and the size allocated to each disk.
	Physical size: Indicates the physical size of this VM.	MB	The detailed diagnosis of this measure if enabled, lists the name of each disk and the physical size that is available in each disk.
	Logical size: Indicates the current logical size of this VM.	MB	The detailed diagnosis of this measure if enabled, lists the name of each disk and the logical size of each disk.
	Free physical size: Indicates the physical size of this VM that is currently free.	MB	A high value is desired for this measure. The detailed diagnosis of this measure if enabled, lists the name of each disk and the physical size that is currently free.
	Percentage of physical size utilized: Indicates the percentage of space that is already utilized by this VM.	Percent	A value close to 100% indicates that the VM is currently running out of physical space. The detailed diagnosis of this measure if enabled, lists the name of each disk and the percentage of space utilized by each disk of the VM.

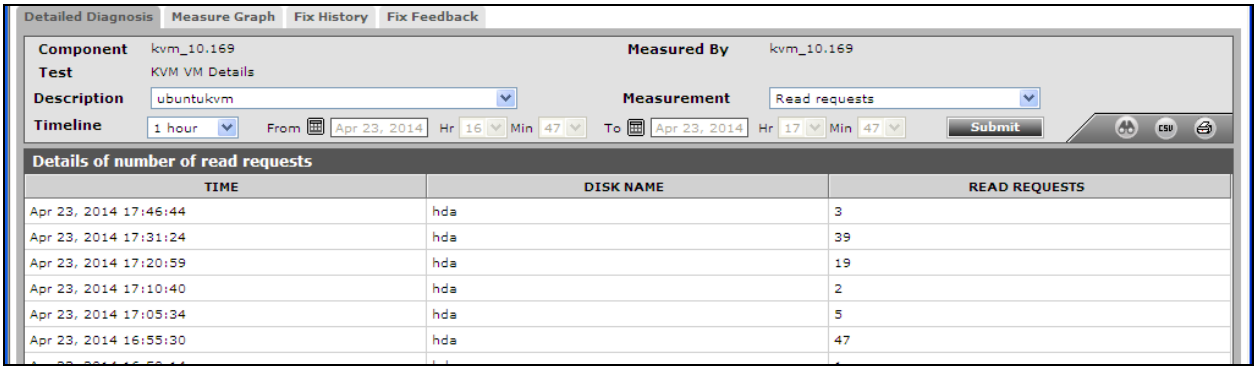
The detailed diagnosis of the Data reads measure lists the name of the disk and the rate at which data is read from this disk. Administrators can instantly identify the disk that is experiencing high I/O activity using the detailed diagnosis.



TIME	DISK NAME	DATA READS(MB/SEC)
Apr 23, 2014 17:31:24	hda	0.0023
Apr 23, 2014 17:20:59	hda	0.0005
Apr 23, 2014 17:10:40	hda	0.0002
Apr 23, 2014 17:05:34	hda	0.0004
Apr 23, 2014 16:55:30	hda	0.0022

Figure 3.9: The detailed diagnosis of the Data reads measure

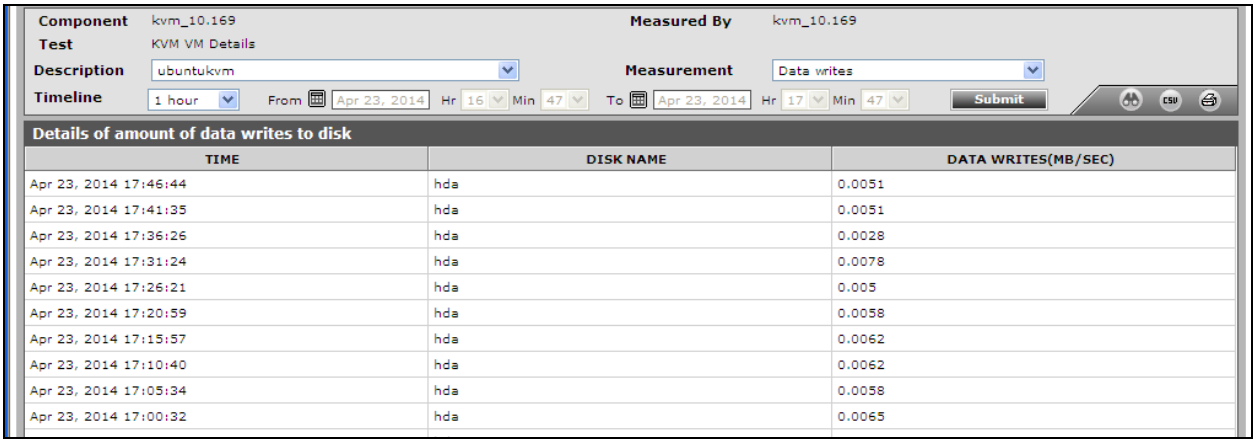
The detailed diagnosis of the Read requests measure if enabled, lists the name of the disk and the number of requests handled. This way, administrators can identify the disk that is handling the maximum number of requests.



TIME	DISK NAME	READ REQUESTS
Apr 23, 2014 17:46:44	hda	3
Apr 23, 2014 17:31:24	hda	39
Apr 23, 2014 17:20:59	hda	19
Apr 23, 2014 17:10:40	hda	2
Apr 23, 2014 17:05:34	hda	5
Apr 23, 2014 16:55:30	hda	47

Figure 3.10: the detailed diagnosis of the Read requests measure

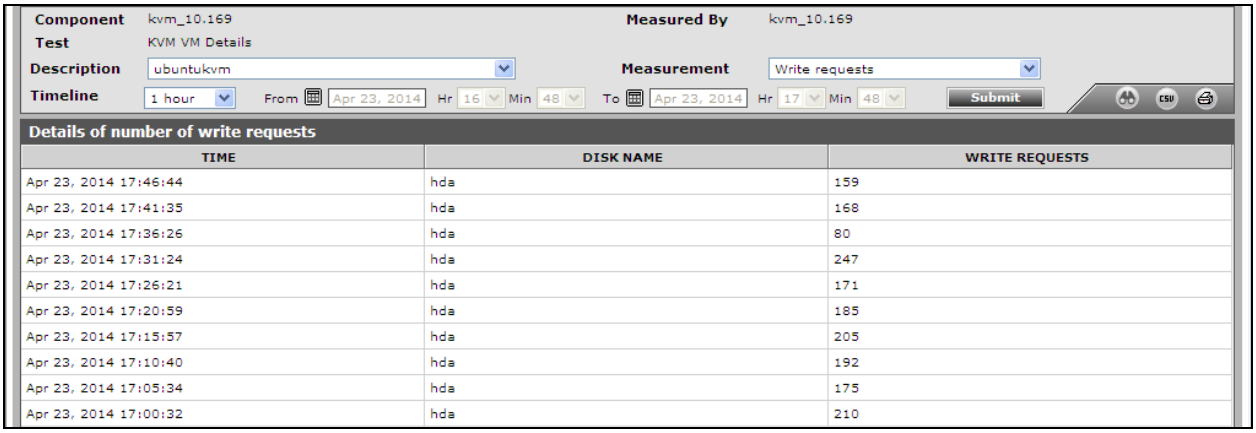
The detailed diagnosis of the Data writes measure lists the name of the disk and the rate at which data is written to the disk.



TIME	DISK NAME	DATA WRITES(MB/SEC)
Apr 23, 2014 17:46:44	hda	0.0051
Apr 23, 2014 17:41:35	hda	0.0051
Apr 23, 2014 17:36:26	hda	0.0028
Apr 23, 2014 17:31:24	hda	0.0078
Apr 23, 2014 17:26:21	hda	0.005
Apr 23, 2014 17:20:59	hda	0.0058
Apr 23, 2014 17:15:57	hda	0.0062
Apr 23, 2014 17:10:40	hda	0.0062
Apr 23, 2014 17:05:34	hda	0.0058
Apr 23, 2014 17:00:32	hda	0.0065

Figure 3.11: The detailed diagnosis of the Data writes measure

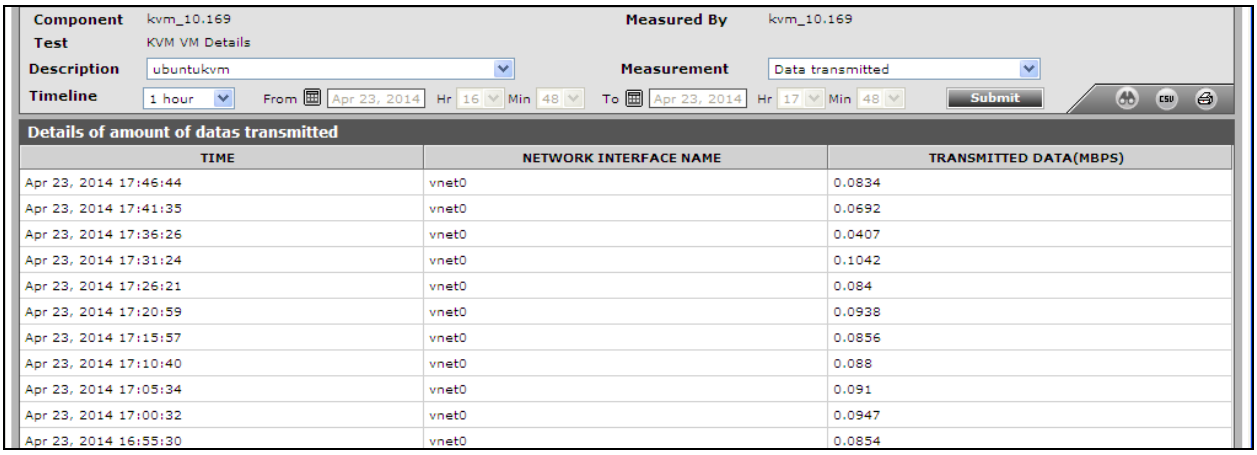
The detailed diagnosis of the Write requests measure if enabled, lists the name of the disk and the number of write requests handled by the disk.



TIME	DISK NAME	WRITE REQUESTS
Apr 23, 2014 17:46:44	hda	159
Apr 23, 2014 17:41:35	hda	168
Apr 23, 2014 17:36:26	hda	80
Apr 23, 2014 17:31:24	hda	247
Apr 23, 2014 17:26:21	hda	171
Apr 23, 2014 17:20:59	hda	185
Apr 23, 2014 17:15:57	hda	205
Apr 23, 2014 17:10:40	hda	192
Apr 23, 2014 17:05:34	hda	175
Apr 23, 2014 17:00:32	hda	210

Figure 3.12: The detailed diagnosis of the Write requests measure

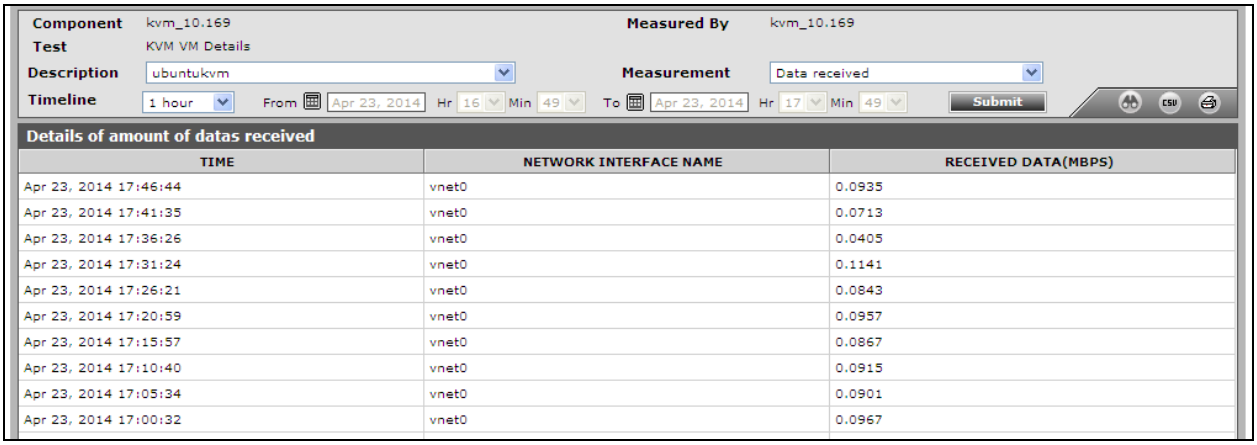
The detailed diagnosis of the *Data transmitted* measure if enabled, lists the name of the network interface through which data is transmitted and the rate at which data is transmitted.



TIME	NETWORK INTERFACE NAME	TRANSMITTED DATA(MBPS)
Apr 23, 2014 17:46:44	vnet0	0.0834
Apr 23, 2014 17:41:35	vnet0	0.0692
Apr 23, 2014 17:36:26	vnet0	0.0407
Apr 23, 2014 17:31:24	vnet0	0.1042
Apr 23, 2014 17:26:21	vnet0	0.084
Apr 23, 2014 17:20:59	vnet0	0.0938
Apr 23, 2014 17:15:57	vnet0	0.0856
Apr 23, 2014 17:10:40	vnet0	0.088
Apr 23, 2014 17:05:34	vnet0	0.091
Apr 23, 2014 17:00:32	vnet0	0.0947
Apr 23, 2014 16:55:30	vnet0	0.0854

Figure 3.13: The detailed diagnosis of the Data transmitted measure

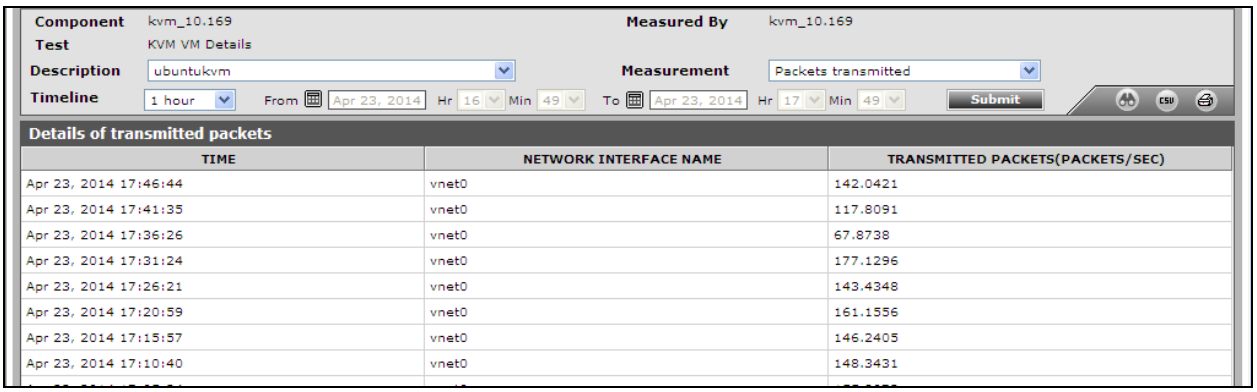
The detailed diagnosis of the Data received measure if enabled, lists the name of the network interface and the rate at which data was received.



TIME	NETWORK INTERFACE NAME	RECEIVED DATA(MBPS)
Apr 23, 2014 17:46:44	vnet0	0.0935
Apr 23, 2014 17:41:35	vnet0	0.0713
Apr 23, 2014 17:36:26	vnet0	0.0405
Apr 23, 2014 17:31:24	vnet0	0.1141
Apr 23, 2014 17:26:21	vnet0	0.0843
Apr 23, 2014 17:20:59	vnet0	0.0957
Apr 23, 2014 17:15:57	vnet0	0.0867
Apr 23, 2014 17:10:40	vnet0	0.0915
Apr 23, 2014 17:05:34	vnet0	0.0901
Apr 23, 2014 17:00:32	vnet0	0.0967

Figure 3.14: The detailed diagnosis of the Data received measure

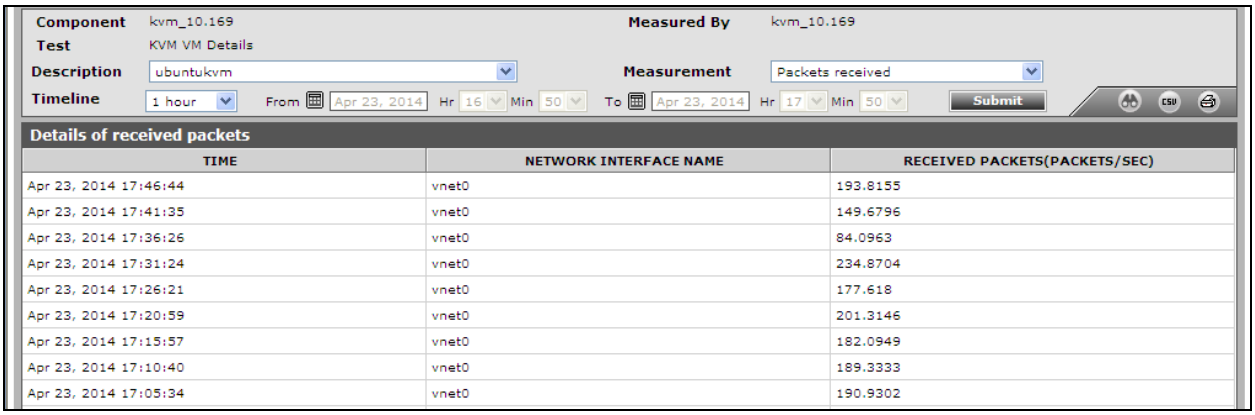
The detailed diagnosis of the *Packets transmitted* measure, lists the name of the network interface through which the packets are transmitted and the rate at which the packets are transmitted.



TIME	NETWORK INTERFACE NAME	TRANSMITTED PACKETS(PACKETS/SEC)
Apr 23, 2014 17:46:44	vnet0	142.0421
Apr 23, 2014 17:41:35	vnet0	117.8091
Apr 23, 2014 17:36:26	vnet0	67.8738
Apr 23, 2014 17:31:24	vnet0	177.1296
Apr 23, 2014 17:26:21	vnet0	143.4348
Apr 23, 2014 17:20:59	vnet0	161.1556
Apr 23, 2014 17:15:57	vnet0	146.2405
Apr 23, 2014 17:10:40	vnet0	148.3431

Figure 3.15: The detailed diagnosis of the Packets transmitted measure

The detailed diagnosis of the Packets received measure if enabled, lists the name of the network interface and the rate at which the data packets were received.



TIME	NETWORK INTERFACE NAME	RECEIVED PACKETS(PACKETS/SEC)
Apr 23, 2014 17:46:44	vnet0	193.8155
Apr 23, 2014 17:41:35	vnet0	149.6796
Apr 23, 2014 17:36:26	vnet0	84.0963
Apr 23, 2014 17:31:24	vnet0	234.8704
Apr 23, 2014 17:26:21	vnet0	177.618
Apr 23, 2014 17:20:59	vnet0	201.3146
Apr 23, 2014 17:15:57	vnet0	182.0949
Apr 23, 2014 17:10:40	vnet0	189.3333
Apr 23, 2014 17:05:34	vnet0	190.9302

Figure 3.16: The detailed diagnosis of the Packets received measaure

3.3.3 VM Connectivity Test

Sometimes, a VM could be in a powered-on state, but the failure of the VM operating system or any fatal error in VM operations could have rendered the VM inaccessible to users. In order to enable administrators to promptly detect such 'hidden' anomalies, the eG agent periodically runs a connectivity check on each VM using this test, and reports whether the VM is accessible over the network or not.

Purpose	Runs a connectivity check on each VM and reports whether the VM is accessible over the network or not
Target of the test	A KVM server host
Agent deploying the test	An internal/remote agent
Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST -** The host for which the test is to be configured. 3. **PORT** - The port at which the specified host listens. By default, this is NULL. 4. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows)**. Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. 5. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your exclude vms specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 6. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments

might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the ignore vms inside view parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your ignore vms inside view specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM server host by default.

Note

While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

7. **PACKETSIZE** - The size of packets used for the test (in bytes)
8. **PACKETCOUNT** - The number of packets to be transmitted during the test
9. **TIMEOUT** - How long after transmission should a packet be deemed lost (in seconds)
10. **PACKETINTERVAL** - Represents the interval (in milliseconds) between successive packet transmissions during the execution of the network test for a specific target.
11. **REPORTUNAVAILABILITY** – By default, this flag is set to **No**. This implies that, by default, the test will not report the unavailability of network connection to any VM. In other words, if the *Network availability* measure of this test registers the value *0* for any VM, then, by default, this test will not report any measure for that VM; under such circumstances, the corresponding VM name will not appear as a descriptor of this test. You can set this flag to **Yes**, if you want the test to report and alert you to the unavailability of the network connection to a VM.

Outputs of the test	One set of results for each VM configured on the KVM server host being monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Avg network delay: Indicates the average delay between transmission of packet to a VM and receipt of the response to the packet at the source.	Secs	An increase in network latency could result from misconfiguration of the router(s) along the path, network congestion, retransmissions at the network, etc.
	Min network delay: The minimum time between transmission of a packet and receipt of the response back.	Secs	A significant increase in the minimum round-trip time is often a sure sign of network congestion.
	Packet loss: Indicates the percentage of packets lost during transmission from source to target and back.	Percent	Packet loss is often caused by network buffer overflows at a network router or by packet corruptions over the network. The detailed diagnosis for this measure provides a listing of routers that are on the path from the external agent to target server, and the delays on each hop. This information can be used to diagnose the hop(s) that could be causing excessive packet loss/delays.
	Network availability of VM: Indicates whether the network connection is available or not.	Percent	A value of 100 indicates that the VM is connected. The value 0 indicates that the VM is not connected. Typically, the value 100 corresponds to a *Packet loss* of 0.

3.3.4 VM Jobs Test

Live migration refers to the process of moving a running virtual machine or application between different physical machines without disconnecting the client or application. Memory, storage, and network connectivity of the virtual machine are transferred from the original host machine to the destination. For a VM migration to be smooth and hassle-free, it is essential to monitor the memory, storage and time that is required for migration. The **VM Jobs** test exactly does the same!

This test captures the total amount of memory, storage and files of the VM that are to be transferred during migration and provides exact pointers to how much of these have already been transferred and how much of these are currently pending for a migration to be complete. Additionally, you could figure out the time since the migration process in the VM started and the estimated time of completion.

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 *KVM server* as the **Component type**, set *Performance* as the **Test type**, choose this test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list.

Purpose	Captures the total amount of memory, storage and files of the VM that are to be transferred during migration and provides exact pointers to how much of these have already been transferred and how much of these are currently pending for a migration to be complete. Additionally, you could figure out the time since the migration process in the VM started and the estimated time of completion.		
Target of the test	A KVM server		
Agent deploying the test	An internal agent.		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** – The IP address of the Sonic firewall 3. **PORT** – The port at which the specified host listens		
Outputs of the test	One set of results for each virtual machine hosted on the KVM server that is being monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Total data: Indicates the total amount of data that is to be transferred during live migration of this VM.	MB	
	Data processed by job: Indicates the total amount of data that is processed since the beginning of this VM migration.	MB	A high value is desired for this measure.
	Data yet to be processed by job: Indicates the amount of data that is yet to be transferred for this VM to be completely migrated.	MB	A high value for this measure indicates that VM migration may take too long to complete.
	Total files: Indicates the total number of files that are to be transferred during live migration of this VM.	Number	

	Files processed by job: Indicates the total number of files that are processed since the beginning of this VM migration.	Number	
	Files yet to be processed by job: Indicates the total number of files that are yet to be transferred for this VM to be completely migrated.	Number	An abnormally high value for this measure is a cause of concern.
	Total memory: Indicates the total amount of memory that is currently required for migrating this VM.	MB	
	Memory processed by job: Indicates the amount of memory that is already consumed for migrating this VM.	MB	
	Memory yet to be processed by job: Indicates the amount of memory that is required to migrate this VM completely.	MB	
	Time elapsed since the start of job: Indicates the time elapsed since the beginning of this VM migration.	Secs	An abnormally high value for this measure indicates trouble during migration of the VMs. This could be due to problems such as processing bottleneck, inadequate memory resources allocated for migration, network congestion etc.
	Time needed for job to finish: Indicates the time that is still required for completing this VM migration.	Secs	An abnormally high value for this measure indicates trouble during migration of the VMs. This could be due to problems such as processing bottleneck, inadequate memory resources allocated for migration, network congestion etc.

3.4 The Inside View of VMs Layer

The **Outside View of VMs** layer provides an "outside" view of the different VM guests - the metrics reported at this layer are based on what the KVM server is seeing about the performance of the individual guest VMs. However, an outside view of the VM guest operating system and its applications may not be sufficient. For instance, suppose one of the disk partitions of the guest operating system has reached capacity. This information cannot be gleaned from host operating system. Likewise, bottlenecks such as a longer process run queue or a higher disk queue length are

more visible using an internal monitor. Internal monitoring (from within the guest operating system) also provides details about the resource utilization of different application(s) or processes.

The tests mapped to the **Inside View of VMs** layer provide an "inside" view of the workings of each of the guests - these tests execute on a KVM server host, but send probes into each of the guest operating systems to analyze how well each guest utilizes the resources that are allocated to it, and how well it handles user sessions, TCP traffic, and network loading.

By default however, clicking on the **Inside View of VMs** layer, does not display the list of tests associated with that layer. Instead, appears. This figure provides you with a list of all VM guests and their respective state (see Figure 3.17).

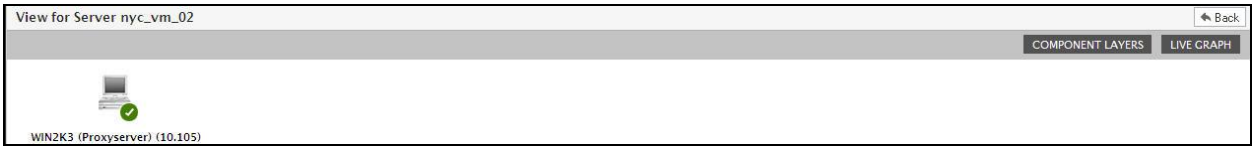


Figure 3.17: A list of guest operating systems on a KVM server host and their current state

To return to the layer model of the *KVM* server and view the tests associated with the **Inside View of VMs** layer, click on the **COMPONENT LAYERS** link in Figure 3.17. You can now view the list of tests mapped to the **Inside View of VMs** layer, as depicted by Figure 3.18 below.

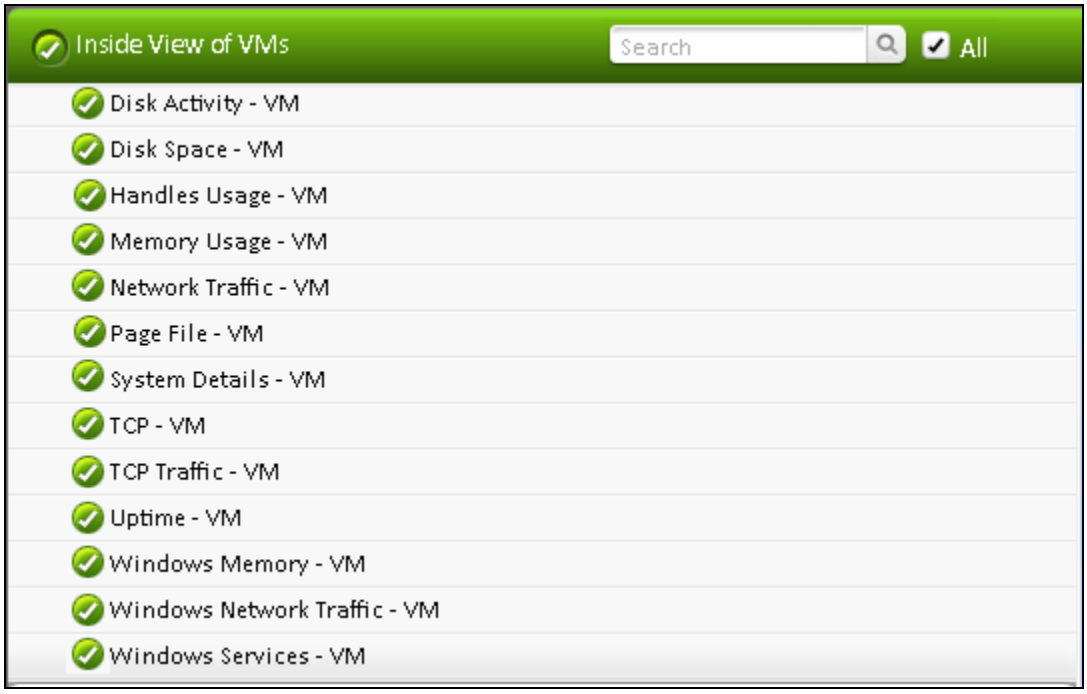


Figure 3.18: The tests mapped to the Inside View of VMs layer

3.4.1 Disk Activity - VM Test

This test reports statistics pertaining to the input/output utilization of each physical disk on a guest.

Purpose	To measure the input/output utilization of each physical disk on each guest of a KVM server
Target of the	A KVM server host

test	
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the domain, admin user, and admin password parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The **ADMIN USER** and **ADMIN PASSWORD** will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the **DOMAIN** parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new

public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
11. **USE SUDO** – This parameter is of significance to **Linux** and **Solaris platforms** only. By default, the **USE SUDO** parameter is set to **false**. This indicates that, by default, Disk Activty -VM test will report the detailed diagnosis for the *Percent virtual disk busy* measure of each disk partition being monitored by executing the `/usr/bin/iotop` command or `/usr/sbin/iotop` command. However, in some highly secure environments, this command cannot be executed directly. In such cases, set this parameter to **true**. This will enable the eG agent to execute the `sudo/usr/bin/iotop` command or `sudo/usr/sbin/iotop` and retrieve the detailed diagnosis of the *Percent virtual disk busy* measure.
12. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *2:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

	13. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG manager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for every combination of *virtual_guest:disk_partition* or *guest_user:disk_partition*.		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Percent virtual disk busy: Indicates the percentage of elapsed time during which the disk is busy processing requests (i.e., reads or writes).	Percent	Comparing the percentage of time that the different disks are busy, an administrator can determine whether load is properly balanced across the different disks. The detailed diagnosis of the *Percent virtual disk busy* measure, if enabled, provides information such as the Process IDs executing on the disk, the Process names, the rate at which I/O read and write requests were issued by each of the processes, and the rate at which data was read from and written into the disk by each of the processes. In the event of excessive disk activity, the details provided in the detailed diagnosis page will enable users to figure out which process is performing the I/O operation that is keeping the disk busy. **The detailed diagnosis for this test is available for Windows guests only, and not Linux guests**.
	Percent reads from virtual disk: Indicates the percentage of elapsed time that the selected disk drive is busy servicing read requests.	Percent	
	Percent writes to virtual disk: Indicates the percentage of elapsed time that the selected disk drive is busy servicing write requests.	Percent	

	Virtual disk read time: Indicates the average time in seconds of a read of data from the disk.	Secs	
	Virtual disk write time: Indicates the average time in seconds of a write of data from the disk.	Secs	
	Avg. queue for virtual disk: Indicates the average number of both read and write requests that were queued for the selected disk during the sample interval.	Number	
	Current queue for virtual disk: The number of requests outstanding on the disk at the time the performance data is collected.	Number	This measure includes requests in service at the time of the snapshot. This is an instantaneous length, not an average over the time interval. Multi-spindle disk devices can have multiple requests active at one time, but other concurrent requests are awaiting service. This counter might reflect a transitory high or low queue length, but if there is a sustained load on the disk drive, it is likely that this will be consistently high. Requests experience delays proportional to the length of this queue minus the number of spindles on the disks. This difference should average less than two for good performance.
	Reads from virtual disk: Indicates the number of reads happening on a logical disk per second.	Reads/Sec	A dramatic increase in this value may be indicative of an I/O bottleneck on the guest.
	Data reads from virtual disk: Indicates the rate at which bytes are transferred from the disk during read operations.	KB/Sec	A very high value indicates an I/O bottleneck on the guest.
	Writes to virtual disk: Indicates the number of writes happening on a local disk per second.	Writes/Sec	A dramatic increase in this value may be indicative of an I/O bottleneck on the guest.

	Data writes to virtual disk: Indicates the rate at which bytes are transferred from the disk during write operations.	KB/Sec	A very high value indicates an I/O bottleneck on the guest.
	Disk service time: Indicates the average time that this disk took to service each transfer request (i.e., the average I/O operation time)	Secs	A sudden rise in the value of this measure can be attributed to a large amount of information being input or output. A consistent increase however, could indicate an I/O processing bottleneck.
	Disk queue time: Indicates the average time that transfer requests waited idly on queue for this disk.	Secs	Ideally, the value of this measure should be low.
	Disk I/O time: Indicates the average time taken for read and write operations of this disk.	Secs	The value of this measure is the sum of the values of the Disk service time and Disk queue time measures. A consistent increase in the value of this measure could indicate a latency in I/O processing.

3.4.1.1 Configuring Users for VM Monitoring

In order to enable the eG agent to connect to VMs in multiple domains and pull out metrics from them, the eG administrative interface provides a special page using which the different domain names, and their corresponding admin user names and admin passwords can be specified. To access this page, just click on the in any of the VM test configuration pages.

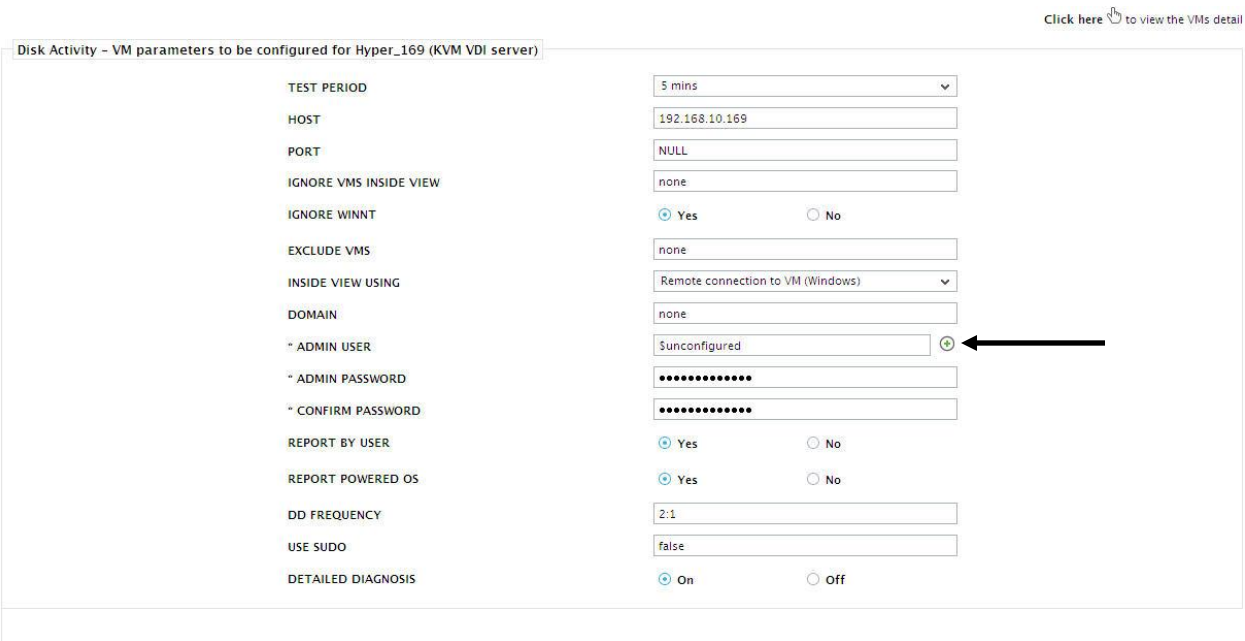


Figure 3.1: Configuring a VM test

Upon clicking, Figure 3.2 will appear, using which the VM user details can be configured.

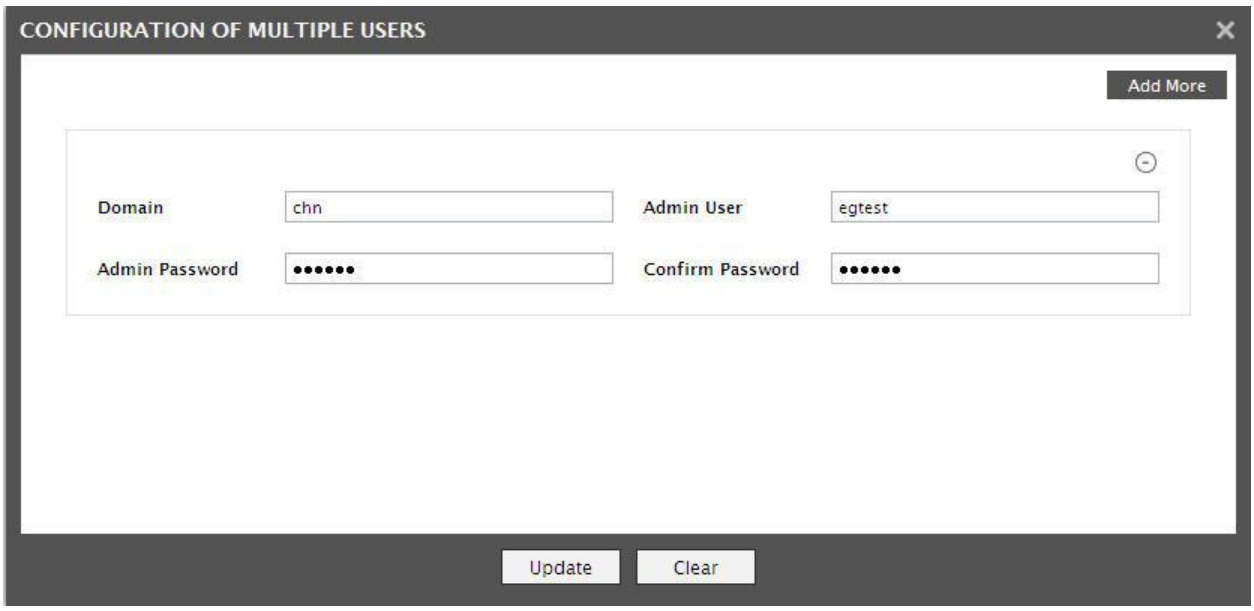


Figure 3.2: The VM user configuration page

To add a user specification, do the following:

1. First, provide the name of the **Domain** to which the VMs belong (see Figure 3.2). If one/more VMs do not belong to any domain, then, specify *none* here.
2. The eG agent must be configured with user privileges that will allow the agent to communicate with the VMs in a particular domain and extract statistics. If *none* is specified against **Domain**, then a local user account can be provided against **Admin User**. On the other hand, if a valid **Domain** name has been specified, then a domain administrator account can be provided in the **Admin User** text box. If key-based authentication is implemented

between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

3. The password of the specified **Admin User** should be mentioned in the **Admin Password** text box.
4. Confirm the password by retyping it in the **Confirm Password** text box.
5. To add more users, click on the **Add More** button in Figure 3.2. This will allow you to add one more user specification as depicted by Figure 3.3.

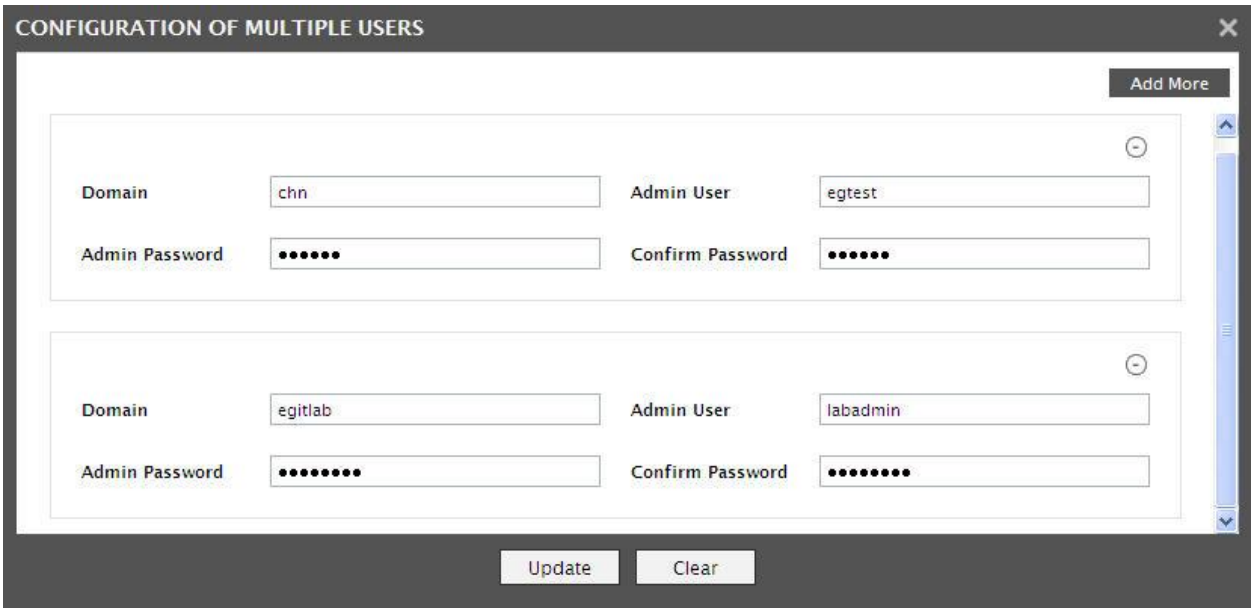


Figure 3.3: Adding another user

6. In some virtualized environments, the same **Domain** could be accessed using multiple **Admin User** names. For instance, to login to a **Domain** named *egitlab*, the eG agent can use the **Admin User** name *labadmin* or the **Admin User** name *jadmin*. You can configure the eG agent with the credentials of both these users as shown by Figure 3.4.

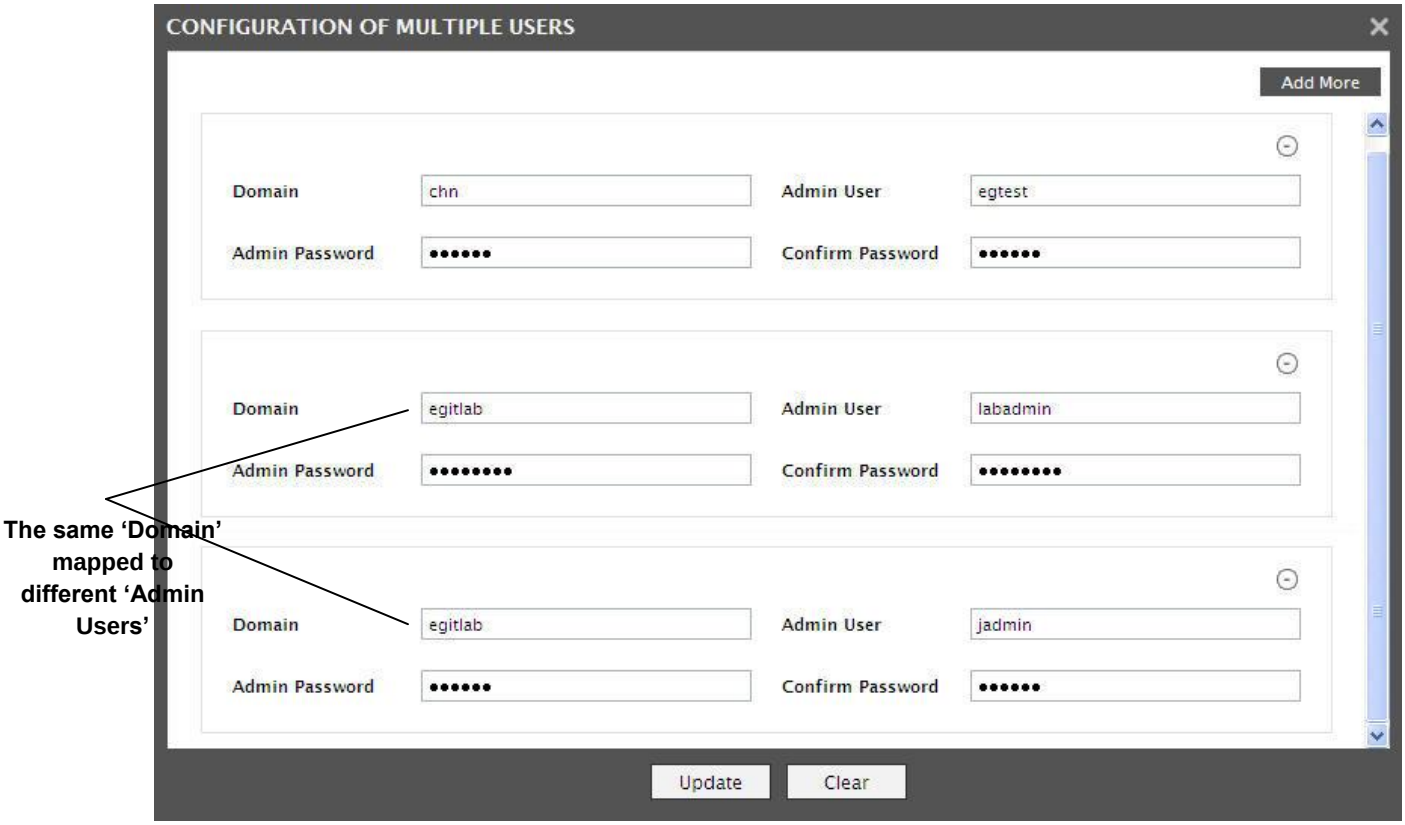


Figure 3.4: Associating a single domain with different admin users

When this is done, then, while attempting to connect to the domain, the eG agent will begin by using the first **Admin User** name of the specification. In the case of Figure 3.4, this will be *labadmin*. If, for some reason, the agent is unable to login using the first **Admin User** name, then it will try to login again, but this time using the second **Admin User** name of the specification - i.e., *jadmin* in our example (see Figure 3.4). If the first login attempt itself is successful, then the agent will ignore the second **Admin User** name.

7. To clear all the user specifications, simply click the **Clear** button in Figure 3.4.
8. To remove the details of a particular user alone, just click the ⊖ button in Figure 3.4.
9. To save the specification, just click on the **Update** button in Figure 3.4. 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 3.5).

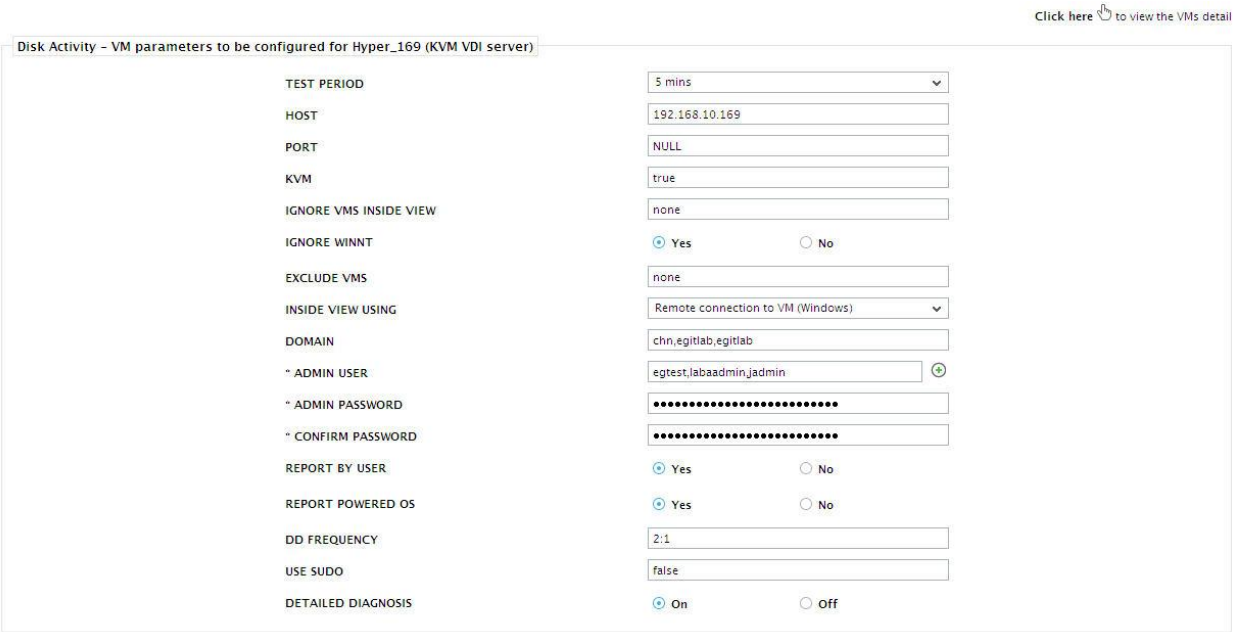


Figure 3.5: The test configuration page displaying multiple domain names, user names, and passwords

3.4.2 Disk Space - VM Test

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

Purpose	To measure the space usage of every disk partition on each guest of a KVM server
Target of the test	A KVM server host
Agent deploying the test	An internal agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

	➢ **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document. ➢ **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*. 9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 10. **REPORT POWERED OS - This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.** If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.		
Outputs of the test	One set of results for every combination of *virtual_guest:disk_partition* or *guest_user:disk_partition*.		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Total capacity: Indicates the total capacity of a disk partition; for the **Total** descriptor, this measure reports the sum of the total capacity of all disk partitions.	MB	

	Used space: Indicates the amount of space used in a disk partition; for the **Total** descriptor, this measure reports the sum of space used across all disk partitions.	MB	
	Free space: Indicates the current free space available for each disk partition of a system; for the **Total** descriptor, this measure reports the sum of the unused space in all disk partitions.	MB	
	Percent usage: Indicates the percentage of space usage on each disk partition of a system; for the **Total** descriptor, this measure reports the percentage of disk space used across all disk partitions.	Percent	A value close to 100% can indicate a potential problem situation where applications executing on the guest may not be able to write data to the disk partition(s) with very high usage.

3.4.3 System Details - VM Test

This test collects various metrics pertaining to the CPU and memory usage of every processor supported by a guest. The details of this test are as follows:

Purpose	To measure the CPU and memory usage of each guest of a KVM server
Target of the test	A KVM server host
Agent deploying the test	An internal agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights**. Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

11. **ENABLE MEMORY DIAGNOSIS** - By default, the **ENABLE MEMORY DIAGNOSIS** flag is set to **NO**, indicating that detailed diagnosis will not be available for the *Free memory in VM* measure reported by this test by default. If you want to view the detailed diagnosis of the *Free memory in VM* measure - i.e., to view the top 10 processes on the VM that are utilizing memory excessively - you can change this flag to **YES**.

12. **USE TOP FOR DD** – This parameter is applicable only for Linux platforms. By default, this parameter is set to **No**. This indicates that, by default, this test will report the detailed diagnosis of the *Virtual CPU utilization* measure for each processor being monitored by executing the `usr/bin/ps` command. In some environments, the detailed diagnosis may not be precisely displayed. In such cases, set the **USE TOP FOR DD** parameter to **Yes**. This will enable the eG agent to extract the exact detailed diagnosis of the *Virtual CPU utilization* measure by executing the `/usr/bin/top` command.

13. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *2:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

	14. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG manager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.
Outputs of the test	One set of results for every combination of *virtual_guest:processor* or *guest_user:processor*.

Measurements made by the test

Measurement	Measurement Unit	Interpretation
Virtual CPU utilization: This measurement indicates the percentage of CPU utilized by the processor.	Percent	A high value could signify a CPU bottleneck. The CPU utilization may be high because a few processes are consuming a lot of CPU, or because there are too many processes contending for a limited resource. The detailed diagnosis of this test reveals the top-10 CPU-intensive processes on the guest.
System usage of virtual CPU: Indicates the percentage of CPU time spent for system-level processing.	Percent	An unusually high value indicates a problem and may be due to too many system-level tasks executing simultaneously.
Run queue in VM: Indicates the instantaneous length of the queue in which threads are waiting for the processor cycle. This length does not include the threads that are currently being executed.	Number	A value consistently greater than 2 indicates that many processes could be simultaneously contending for the processor.
Blocked processes in VM: Indicates the number of processes blocked for I/O, paging, etc.	Number	A high value could indicate an I/O problem on the guest (e.g., a slow disk).

	Swap memory in VM: Denotes the committed amount of virtual memory. This corresponds to the space reserved for virtual memory on disk paging file(s).	MB	An unusually high value for the swap usage can indicate a memory bottleneck. Check the memory utilization of individual processes to figure out the process(es) that has (have) maximum memory consumption and look to tune their memory usages and allocations accordingly.
	Free memory in VM: Indicates the free memory available.	MB	A very low value of free memory is also an indication of high memory utilization on a guest. The detailed diagnosis of this measure, if enabled, lists the top 10 processes responsible for maximum memory consumption on the guest.
	Scan rate in VM: Indicates the memory scan rate.	Pages/Sec	A high value is indicative of memory thrashing. Excessive thrashing can be detrimental to guest performance.

Note

For multi-processor systems, where the CPU statistics are reported for each processor on the system, the statistics that are system-specific (e.g., run queue length, free memory, etc.) are only reported for the "Summary" descriptor of this test.

The detailed diagnosis capability of the *Virtual CPU utilization* measure, if enabled, provides a listing of the top 10 CPU-consuming processes (see Figure 3.6). In the event of a Cpu bottleneck, this information will enable users to identify the processes consuming a high percentage of CPU time. The users may then decide to stop such processes, so as to release the CPU resource for more important processing purposes.

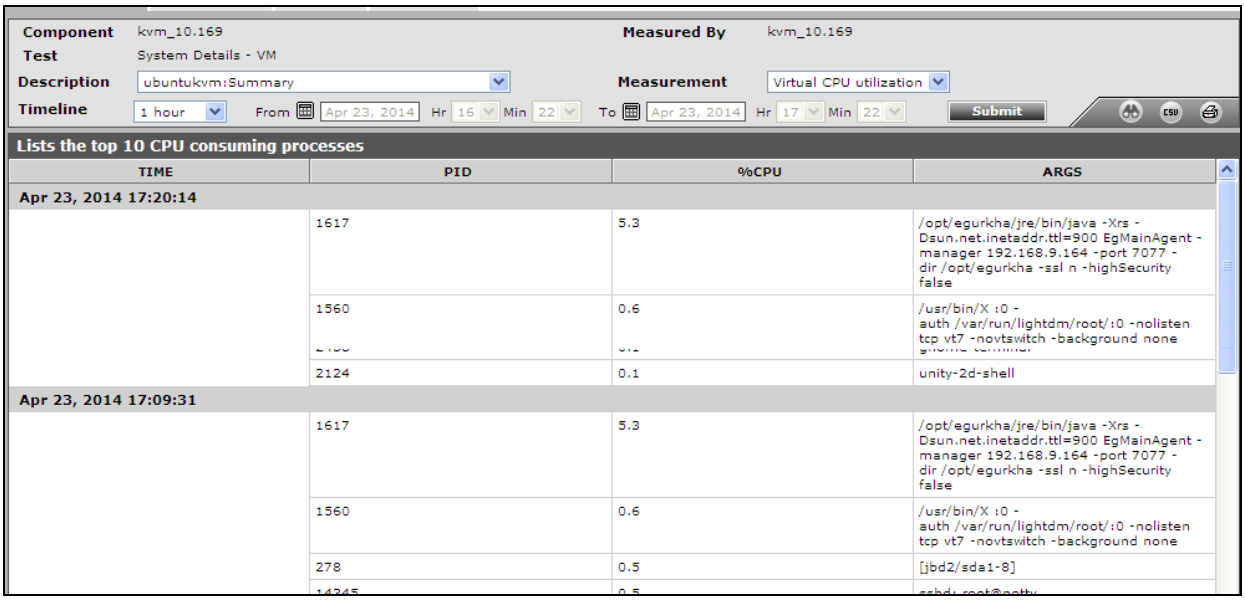


Figure 3.6: The top 10 CPU consuming processes

Note

While instantaneous spikes in CPU utilization are captured by the eG agents and displayed in the **Measures** page, the detailed diagnosis will not capture/display such instantaneous spikes. Instead, detailed diagnosis will display only a consistent increase in CPU utilization observed over a period of time.

3.4.4 Uptime - VM Test

In most virtualized environments, it is essential to monitor the uptime of VMs hosting critical server applications in the infrastructure. By tracking the uptime of each of the VMs, administrators can determine what percentage of time a VM has been up. Comparing this value with service level targets, administrators can determine the most trouble-prone areas of the virtualized infrastructure.

In some environments, administrators may schedule periodic reboots of their VM. By knowing that a specific VM has been up for an unusually long time, an administrator may come to know that the scheduled reboot task is not working on a VM.

The Uptime - VM test included in the eG agent monitors the uptime of each VM on a KVM server.

Purpose	To monitor the uptime of each VM on a KVM server
Target of the test	A KVM server host
Agent deploying the test	An internal agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

11. **REPORTMANAGERTIME** – If this flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such VMs will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

12. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *2:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

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

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

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

Outputs of the test	One set of results for each VM discovered on the KVM server being monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Has the VM been rebooted?: Indicates whether this guest has been rebooted during the last measurement period or not.	Boolean	If this measure shows 1, it means that the guest was rebooted during the last measurement period. By checking the time periods when this metric changes from 0 to 1, an administrator can determine the times when this guest was rebooted.
	Uptime of the VM during the last measure period: Indicates the time period that the guest has been up since the last time this test ran.	Secs	If the guest has not been rebooted during the last measurement period and the agent has been running continuously, this value will be equal to the measurement period. If the VM was rebooted during the last measurement period, this value will be less than the measurement period of the test. For example, if the measurement period is 300 secs, and if the VM was rebooted 120 secs back, this metric will report a value of 120 seconds. The accuracy of this metric is dependent on the measurement period - the smaller the measurement period, greater the accuracy.
	Total uptime of the VM: Indicates the total time that the guest has been up since its last reboot.	Mins	Administrators may wish to be alerted if a VM has been running without a reboot for a very long period. Setting a threshold for this metric allows administrators to determine such conditions.

Note

If a value less than a minute is configured as the test period of the Uptime - VM test, then, the **Uptime of the VM during the last measure period** measure will report the value 0 for Unix VMs (only) until the minute boundary is crossed. For instance, if you configure the Uptime - VM test to run every 10 seconds, then, for the first 5 test execution cyles (i.e., 10 x 5 = 50 seconds), the **Uptime of the VM during the last measure period** measure will report the value 0 for Unix VMs; however, the sixth time the test executes (i.e, when test execution touches the 1 minute boundary), this measure will report the value 60 seconds for the same VMs. Thereafter, every sixth measurement period will report 60 seconds as the uptime of the Unix VMs. This is because, Unix-based operating systems report uptime only in minutes and not in seconds.

3.4.5 Windows Memory - VM Test

To understand the metrics reported by this test, it is essential to understand how memory is handled by the operating system. On any Windows system, memory is partitioned into a part that is available for user processes,

and another that is available to the OS kernel. The kernel memory area is divided into several parts, with the two major parts (called "pools") being a nonpaged pool and a paged pool. The nonpaged pool is a section of memory that cannot, under any circumstances, be paged to disk. The paged pool is a section of memory that can be paged to disk. (Just being stored in the paged pooldoesn't necessarily mean that something has been paged to disk. It just means that it has either been paged to disk or it could be paged to disk.) Sandwiched directly in between the nonpaged and paged pools (although technically part of the nonpaged pool) is a section of memory called the "System Page Table Entries," or "System PTEs."

This test tracks critical metrics corresponding to the System PTEs and the pool areas of kernel memory of each Windows virtual machine of a KVM server.

Purpose	Tracks critical metrics corresponding to the System PTEs and the pool areas of kernel memory of each Windows-based virtual guest of a KVM server
Target of the test	A KVM server host
Agent deploying the test	An internal agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights**. Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

	➢ **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document. ➢ **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*. 9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.** If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 11. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG manager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for each VM discovered on the KVM server being monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

test	**Free entries in system page table:** Indicates the number of page table entries not currently in use by the guest.	Number	The maximum number of System PTEs that a server can have is set when the server boots. In heavily-used servers, you can run out of system PTEs. You can use the registry to increase the number of system PTEs, but that encroaches into the paged pool area, and you could run out of paged pool memory. Running out of either one is bad, and the goal should be to tune your server so that you run out of both at the exact same time. Typically, the value of this metric should be above 3000.
	Page read rate in VM: Indicates the average number of times per second the disk was read to resolve hard fault paging.	Reads/Sec	
	Page write rate in VM: Indicates the average number of times per second the pages are written to disk to free up the physical memory.	Writes/Sec	
	Page input rate in VM: Indicates the number of times per second that a process needed to access a piece of memory that was not in its working set, meaning that the guest had to retrieve it from the page file.	Pages/Sec	
	Page output rate in VM: Indicates the number of times per second the guest decided to trim a process's working set by writing some memory to disk in order to free up physical memory for another process.	Pages/Sec	This value is a critical measure of the memory utilization on a guest. If this value never increases, then there is sufficient memory in the guest. Instantaneous spikes of this value are acceptable, but if the value itself starts to rise over time or with load, it implies that there is a memory shortage on the guest.
	Memory pool non-paged data in VM: Indicates the total size of the kernel memory nonpaged pool.	MB	The kernel memory nonpage pool is an area of guest memory (that is, memory used by the guest operating system) for kernel objects that cannot be written to disk, but must remain in memory as long as the objects are allocated. Typically, there should be no more than 100 MB of non-paged pool memory being used.

	Memory pool paged data in VM : Indicates the total size of the Paged Pool.	MB	If the Paged Pool starts to run out of space (when it's 80% full by default), the guest will automatically take some memory away from the System File Cache and give it to the Paged Pool. This makes the System File Cache smaller. However, the system file cache is critical, and so it will never reach zero. Hence, a significant increase in the paged pool size is a problem. This metric is a useful indicator of memory leaks in a guest. A memory leak occurs when the guest allocates more memory to a process than the process gives back to the pool. Any time of process can cause a memory leak. If the amount of paged pool data keeps increasing even though the workload on the guest remains constant, it is an indicator of a memory leak.

3.4.6 Windows Network Traffic - VM Test

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

Purpose	To measure the incoming and outgoing traffic through each Windows-based guest of a KVM server host
Target of the test	A KVM server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Outputs of the test: One set of results for every *Windows_virtual_guest:network_interface* combination or *Windows_VM_guest_user:network_interface* combination

Measurements made by the test

Measurement	Measurement Unit	Interpretation
Incoming traffic: Indicates the rate at which data (including framing characters) is received on a network interface.	Mbps	An abnormally high rate of incoming traffic may require additional analysis.
Outgoing traffic: Represents the rate at which data (including framing characters) is sent on a network interface.	Mbps	An abnormally high rate of outgoing traffic may require additional analysis.
Maximum bandwidth: An estimate of the capacity of a network interface.	Mbps	

	Bandwidth usage: Indicates the percentage of bandwidth used by a network interface.	Percent	By comparing the bandwidth usage with the maximum bandwidth of an interface, an administrator can determine times when the network interface is overloaded or is being a performance bottleneck.
	Output queue length: Indicates the length of the output packet queue (in packets).	Number	If this is longer than 2, delays are being experienced and the bottleneck should be found and eliminated if possible.
	Outbound packet errors: The number of outbound packets that could not be transmitted because of errors.	Number	Ideally, number of outbound errors should be 0.
	Inbound packet errors: The number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.	Number	Ideally, number of inbound errors should be 0.

If the Windows Network Traffic - VM test is not reporting measures for a VM, make sure that you have enabled the SNMP service for the guest.

3.4.7 Network Traffic - VM Test

This is an internal test that monitors the incoming and outgoing traffic through each guest on a KVM server.

Purpose	To measure the incoming and outgoing traffic through each guest on a KVM server
Target of the test	A KVM server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Outputs of the test: One set of results for every *virtual_guest:network_interface* combination

Measurements made by the test:

Measurement	Measurement Unit	Interpretation
Incoming network traffic: Indicates the rate of incoming traffic.	Pkts/Sec	An increase in traffic to the guest can indicate an increase in accesses to the guest (from users or from other applications) or that the guest is under an attack of some form.
Outgoing network traffic: Represents the rate of outgoing traffic.	Pkts/Sec	An increase in traffic from the guest can indicate an increase in accesses to the guest (from users or from other applications).

3.4.8 TCP - VM Test

This is an internal test that monitors the incoming and outgoing traffic through each guest on a KVM server.

Purpose	To measure statistics pertaining to the TCP layer of a guest
Target of the test	A KVM server host
Agent deploying the test	A remote agent
Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. Note: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

Outputs of the test: One set of results for each powered-on guest/logged-in user on the KVM server monitored

Measurements made by the test:

Measurement	Measurement Unit	Interpretation
Incoming connections to VM: Indicates the connections per second received by the guest.	Conns/Sec	A high value can indicate an increase in input load.
Outgoing connections to VM: Indicates the connections per second initiated by the guest.	Conns/Sec	A high value can indicate that one or more of the applications executing on the guest have started using a number of TCP connections to some other guest or host.

	Current connections to VM: Indicates the currently established connections.	Number	A sudden increase in the number of connections established on a guest can indicate either an increase in load to one or more of the applications executing on the guest, or that one or more of the applications are experiencing a problem (e.g., a slow down). On Microsoft Windows, the current connections metrics is the total number of TCP connections that are currently in the ESTABLISHED or CLOSE_WAIT states.
	Connection drops on VM: Indicates the rate of established TCP connections dropped from the TCP listen queue.	Conns/Sec	This value should be 0 for most of the time. Any non-zero value implies that one or more applications on the guest are under overload.
	Connection failures on VM: Indicates the rate of half open TCP connections dropped from the listen queue.	Conns/Sec	This value should be 0 for most of the time. A prolonged non-zero value can indicate either that the server is under SYN attack or that there is a problem with the network link to the server that is resulting in connections being dropped without completion.

3.4.9 TCP Traffic - VM Test

Since most popular applications rely on the TCP protocol for their proper functioning, traffic monitoring at the TCP protocol layer can provide good indicators of the performance seen by the applications that use TCP. The most critical metric at the TCP protocol layer is the percentage of retransmissions. Since TCP uses an exponential back-off algorithm for its retransmissions, any retransmission of packets over the network (due to network congestion, noise, data link errors, etc.) can have a significant impact on the throughput seen by applications that use TCP. This test monitors the TCP protocol traffic to and from a guest, and particularly monitors retransmissions.

Purpose	Monitors the TCP protocol traffic to and from each guest of a KVM server, and particularly measures the percentage of retransmission
Target of the test	A KVM server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

	➢ **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document. ➢ **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*. 9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.** If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 11. **SEGMENTS SENT MIN** - Specify the minimum threshold for the number of segments sent/transmitted over the network. The default value is 10; in this case, the test will compute/report the **Retransmit ratio from VM** measure only if more than 10 segments are sent over the network – i.e., if the value of the **Segments sent by VM** measure crosses the value 10. On the other hand, if the **Segments sent by VM** measure reports a value less than 10, then the test will not compute/report the **Retransmit ratio from VM** measure. This is done to ensure that no false alerts are generated by the eG Enterprise system for the **Retransmit ratio from VM** measure. You can change this minimum threshold to any value of your choice.
Outputs of the test	One set of results for each powered-on guest/currently logged-in user on the KVM server monitored

Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Segments received by VM: Indicates the rate at which segments are received by the guest.	Segments/Sec	

	Segments sent by VM: Indicates the rate at which segments are sent to clients or other guests.	Segments/Sec	
	Retransmits by VM: Indicates the rate at which segments are being retransmitted by the guest.	Segments/Sec	
	Retransmit ratio from VM: Indicates the ratio of the rate of data retransmissions to the rate of data being sent by the guest.	Percent	Ideally, the retransmission ratio should be low (< 5%). Most often retransmissions at the TCP layer have significant impact on application performance. Very often a large number of retransmissions are caused by a congested network link, bottlenecks at a router causing buffer/queue overflows, or by lousy network links due to poor physical layer characteristics (e.g., low signal to noise ratio). By tracking the percentage of retransmissions at a guest, an administrator can quickly be alerted to problem situations in the network link(s) to the guest that may be impacting the service performance.

3.4.10 Handles Usage - VM Test

This test monitors and tracks the handles opened by processes running in a target Windows virtual machine.

Purpose	Monitors and tracks the handles opened by processes running in a target Windows virtual machine
Target of the test	A KVM server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

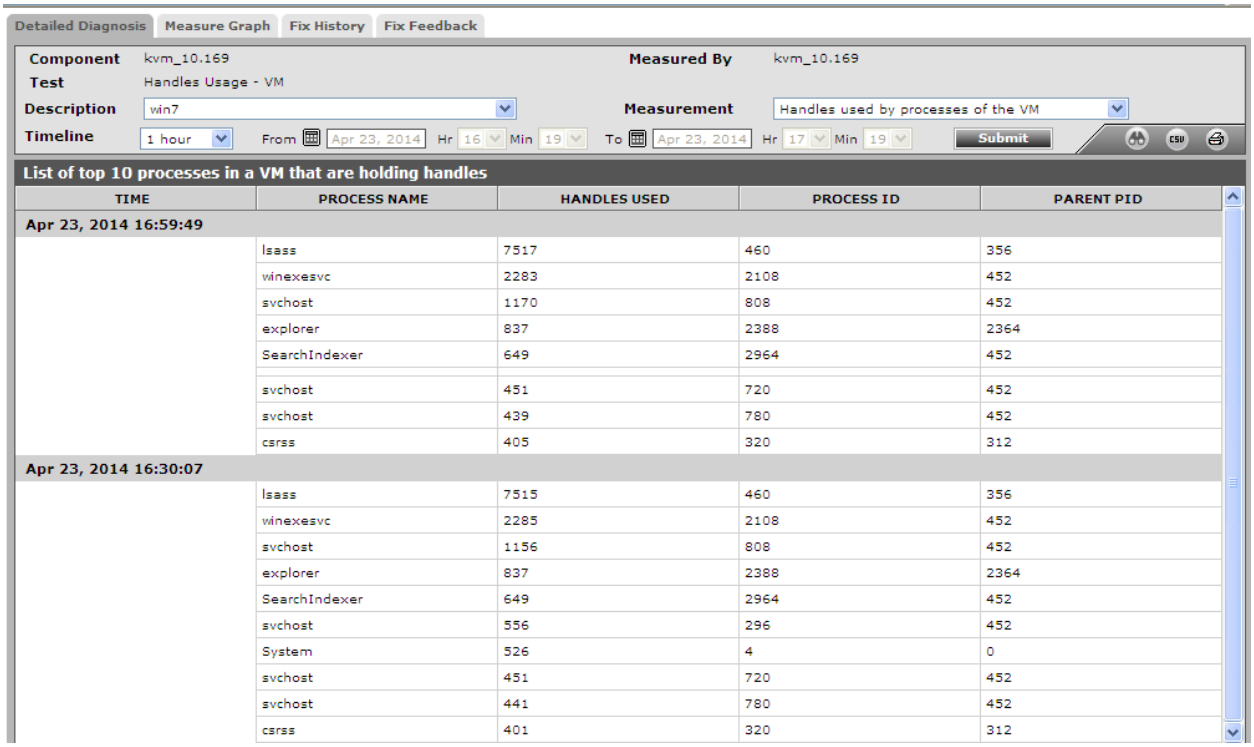
    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
11. **HANDLES GROWTH LIMIT** – This defines the upper limit of the handles opened by any process. By default, this parameter is set to 8000.
12. **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 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 mranager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Outputs of the test	One set of results for each powered-on guest/currently logged-in user on the KVM server monitored		
Measurements made by the test	Measurement	Measurement Unit	Interpretation
	Handles used by processes of the VM: Indicates the number of handles opened by various processes running in a target Windows virtual machine in the last measurement period.	Number	Use the detailed diagnosis of this measure to determine the top-10 processes in terms of number of handles opened. This information brings to light those processes with too many open handles. By closely tracking the handle usage of these processes over time, you can identify potential handle leaks.
	Processes using handles above limit in the VM: Indicates the number of processes that have opened the handles on or above the value defined in the input parameter - **HANDLES GROWTH LIMIT**.	Number	Using the detailed diagnosis of this measure, you can accurately isolate the process(es) that has opened more handles than the permitted limit. A high value of this measure indicates that too many processes are opening handles excessively. You might want to closely observe the handle usage of these processes over time to figure out whether the spike in usage is sporadic or consistent. A consistent increase in handle usage could indicate a handle leak.

The detailed diagnosis of the *Handles used by processes* measure, if enabled, lists the names of top-10 processes in terms of handle usage, the number of handles each process uses, the process ID, and the ID of the parent process.



TIME	PROCESS NAME	HANDLES USED	PROCESS ID	PARENT PID
Apr 23, 2014 16:59:49				
	lsass	7517	460	356
	winexesvc	2283	2108	452
	svchost	1170	808	452
	explorer	837	2388	2364
	SearchIndexer	649	2964	452
	svchost	451	720	452
	svchost	439	780	452
	csrss	405	320	312
Apr 23, 2014 16:30:07				
	lsass	7515	460	356
	winexesvc	2285	2108	452
	svchost	1156	808	452
	explorer	837	2388	2364
	SearchIndexer	649	2964	452
	svchost	556	296	452
	System	526	4	0
	svchost	451	720	452
	svchost	441	780	452
	csrss	401	320	312

Figure 3.19: The detailed diagnosis of the Handles used by processes measure

The detailed diagnosis of the *Processes using handles above limit in VM* measure, if enabled, lists the details of processes that are using more handles than the configured limit.

Time	Process Name	Handles used	Process ID	Parent PID
Jan 29, 2009 17:54:18	eGRSvc	62410	412	11512

Figure 3.20: The detailed diagnosis of the Processes using handles above limit in VM measure

3.4.11 Windows Services - VM Test

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

Purpose	Tracks the status (whether running or have stopped) of services executing on Windows virtual machines
Target of the test	A KVM server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
11. **IGNORESERVICES** - Provide a comma-separated list of services that need to be ignored while monitoring. When configuring a service name to exclude, make sure that you specify the **Display Name** of the service, and not the service **Name** you see in the **Services** window on your Windows VM.
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 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 mranager license should allow the detailed diagnosis capability
    - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Outputs of the test	One set of results for each powered-on guest/currently logged-in user on the KVM server monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	New automatic services started: Indicates the number of Windows services with startup type as *automatic*, which were running in the last measurement period.	Number	The detailed diagnosis of this measure lists the services (with startup type as *automatic*) that are running.
	New automatic services stopped: Indicates the number of Windows services with startup type as *automatic*, which were not running in the last measurement period.	Number	To know which services stopped, use the detailed diagnosis of this measure (if enabled).
	New manual services started: Indicates the number of Windows services with startup type as *manual*, which were running in the last measurement period.	Number	Use the detailed diagnosis of this measure to identify the *manual* services that are running.
	New manual services stopped: Indicates the number of Windows services with startup type as *manual*, which stopped running in the last measurement period.	Number	To identify the services that stopped, use the detailed diagnosis of this measure.

3.4.12 Memory Usage - VM Test

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

Purpose	Reports statistics related to the usage of the physical memory of the VMs
Target of the test	A KVM server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

	➢ **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document. ➢ **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*. 9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.** If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 11. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is 1:1. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY.** 12. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG mranager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.
Outputs of the test	One set of results for each powered-on guest/currently logged-in user on the KVM server monitored

Measurements made by the test	Measurement	Measurement Unit	Interpretation
	Total physical memory: Indicates the total physical memory of this VM.	MB	
	Used physical memory: Indicates the used physical memory of this VM.	MB	
	Free physical memory: Indicates the free physical memory of the VM.	MB	This measure typically indicates the amount of memory available for use by applications running on the target VM. On Unix operating systems (AIX and Linux), the operating system tends to use parts of the available memory for caching files, objects, etc. When applications require additional memory, this is released from the operating system cache. Hence, to understand the true free memory that is available to applications, the eG agent reports the sum of the free physical memory and the operating system cache memory size as the value of the *Free physical memory* measure while monitoring AIX and Linux guest operating systems.
	Physical memory utilized: Indicates the percent usage of physical memory by this VM.	Percent	Ideally, the value of this measure should be low. While sporadic spikes in memory usage could be caused by one/more rogue processes on the VM, a consistent increase in this value could be a cause for some serious concern, as it indicates a gradual, but steady erosion of valuable memory resources. If this unhealthy trend is not repaired soon, it could severely hamper VM performance, causing anything from a slowdown to a complete system meltdown. You can use the detailed diagnosis of this measure to figure out which processes on the VM are consuming memory excessively.

	Available physical memory: Indicates the amount of physical memory, immediately available for allocation to a process or for system use.	MB	Not all of the *Available physical memory* is *Free physical memory*. Typically, *Available physical memory* is made up of the Standby List, Free List, and Zeroed List. When Windows wants to trim a process' working set, the trimmed pages are moved (usually) to the Standby List. From here, they can be brought back to life in the working set with only a soft page fault (much faster than a hard fault, which would have to talk to the disk). If a page stays in the standby List for a long time, it gets freed and moved to the Free List. In the background, there is a low priority thread (actually, the only thread with priority 0) which takes pages from the Free List and zeros them out. Because of this, there is usually very little in the Free List. All new allocations always come from the Zeroed List, which is memory pages that have been overwritten with zeros. This is a standard part of the OS' cross-process security, to prevent any process ever seeing data from another. If the Zeroed List is empty, Free List memory is zeroed and used or, if that is empty too, Standby List memory is freed, zeroed, and used. It is because all three can be used with so little effort that they are all counted as "available". A high value is typically desired for this measure. **This measure will be available for Windows 2008 VMs only.**

	Modified memory: Indicates the amount of memory that is allocated to the modified page list.	MB	This memory contains cached data and code that is not actively in use by processes, the system and the system cache. This memory needs to be written out before it will be available for allocation to a process or for system use. Cache pages on the modified list have been altered in memory. No process has specifically asked for this data to be in memory, it is merely there as a consequence of caching. Therefore it can be written to disk at any time (not to the page file, but to its original file location) and reused. However, since this involves I/O, it is not considered to be Available physical memory. **This measure will be available for Windows 2008 VMs only.**
	Standby memory: Indicates the amount of memory assigned to the standby list.	MB	This memory contains cached data and code that is not actively in use by processes, the system and the system cache. It is immediately available for allocation to a process or for system use. If the system runs out of available free and zero memory, memory on lower priority standby cache page lists will be repurposed before memory on higher priority standby cache page lists. Typically, Standby memory is the aggregate of Standby Cache Core Bytes,Standby Cache Normal Priority Bytes, and Standby Cache Reserve Bytes. Standby Cache Core Bytes is the amount of physical memory, that is assigned to the core standby cache page lists. Standby Cache Normal Priority Bytes is the amount of physical memory, that is assigned to the normal priority standby cache page lists. Standby Cache Reserve Bytes is the amount of physical memory, that is assigned to the reserve standby cache page lists. **This measure will be available for Windows 2008 VMs only.**
	Cached memory: This measure is an aggregate of *Standby memory* and *Modified memory*.	MB	**This measure will be available for Windows 2008 VMs only.**

Note

While monitoring Linux/AIX guest operating systems, you may observe discrepancies between the value of the *Physical memory utilized* measure and the memory usage percentages reported per process by the detailed diagnosis of the same measure. This is because, while the *Physical memory utilized* measure takes into account the memory in the OS cache of the Linux/AIX VM, the memory usage percent that the detailed diagnosis reports per process does not consider the OS cache memory.

3.4.13 Page File - VM Test

When the load imposed by applications and services running on a VM nears the amount of RAM allocated for the VM, additional storage is necessary. The page file serves as the temporary store on disk for memory that cannot be accommodated in the physical RAM. Since it is frequently accessed for storing and retrieving data that is needed for virtual memory access by application, the location and sizing of the page files can have a critical impact on VM's performance. Ideally, the server operating system and the page file should be available on different drives for optimal performance. Splitting the page file across different drives can improve performance further. A rule of thumb in sizing the page file is to set the maximum size of the page file to 1.5 times the available RAM. While this works well for VMs with smaller physical memory, for other VMs, the optimal page file size has to be determined based on experience using the system and studying the typical workload. This test tracks the usage of each page file on the VMs.

Note

This test is applicable only for Windows VMs.

Purpose	Tracks the usage of each page file on the VMs
Target of the test	A KVM server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights**. Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

	➢ **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document. ➢ **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*. 9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.** If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 11. **REPORTTOTAL** - Set the **REPORTTOTAL** flag to **Yes** if you want the test to report total page file usage - i.e., the aggregate usage across multiple page files. In this case therefore, a **Total** descriptor will newly appear for this test in the eG monitoring console. 12. **REPORTTOTALONLY** - If both the **REPORTTOTAL** and **REPORTTOTALONLY** flags are set to **Yes**, then the test will report only the aggregate usage across multiple page files - in other words, the test will report values for the Total descriptor only. Likewise, if the **REPORTTOTAL** flag is set to **No**, and the **REPORTTOTALONLY** flag is set to **Yes**, then again, the test will report current usage for the **Total** descriptor only. However, if both the **REPORTTOTAL** and **REPORTTOTALONLY** flags are set to **No**, then the test will report individual usages only. Also, if the **REPORTTOTAL** flag is set to **Yes** and the **REPORTTOTALONLY** flag is set to **No**, then both the individual and **Total** usages will be reported.		
Outputs of the test	One set of results for each powered-on guest/currently logged-in user on the KVM server monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

test	**Current usage:** Indicates the current usage of this page file.	Percent	This metric should be less than 90%. If the page file does not have additional space, additional users/processes cannot be supported and system performance will suffer. To improve performance, consider resizing the page file. Microsoft Windows allows a minimum and maximum size of the page file to be specified. If the system has sufficient disk space, consider setting the page file to start out at the maximum size (by using the same value for the minimum and maximum sizes), so that system resources are not spent growing the page file size when there is a virtual memory shortage.

3.4.14 Domain Time Sync – VM Test

Time synchronization is one of the most important dependencies of windows. A time protocol is responsible for determining the best available time information and converging the clocks to ensure that a consistent time is maintained across systems. By default, windows support a tolerance of plus or minus five minutes for clocks. If the time variance exceeds this setting, clients will be unable to authenticate and in the case of domain controllers, replication will not occur. It implements a time synchronization system based on Network Time Protocol (NTP).

NTP is a fault-tolerant, highly scalable time protocol and it is used for synchronizing computer clocks by using a designated reference clock. A reference clock is some device or machinery that spits out the current time. The special thing about these things is accuracy. Reference clocks must be accurately following some time standard. NTP will compute some additional statistical values based on the current time reported by the reference clock, which will describe the quality of time it sees. Among these values are: offset (or phase), jitter (or dispersion), frequency error, and stability. Thus each NTP server will maintain an estimate of the quality of its reference clocks and of itself.

This test reports the time difference between the reference clock and that of the target environment, and thus helps assess the quality of time seen by the Windows VM. With the help of this test, you can also easily determine whether the reference time changed recently.

Note

This test reports metrics for Windows VMs only.

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

Purpose	Reports the time difference between the reference clock and that of the target environment, and thus helps assess the quality of time seen by the Windows VM
Target of the test	A KVM server host

Agent deploying the test	A remote agent
Configurable parameters for the test **Configurable parameters for the test**	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is NULL. 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on Windows NT operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default. 7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights**. Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the **eG VM Agent**, set the **INSIDE VIEW USING** flag to **eG VM Agent** (Windows). Once this is done, you can set the **DOMAIN, ADMIN USER,** and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain :** If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the DOMAIN field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

9. If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

	➢ **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple DOMAIN names, multiple ADMIN USER names and ADMIN PASSWORDs would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document. ➢ **If the inside view using flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to none. 10. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 11. **REPORT POWERED OS** - This flag becomes relevant only if the report by user flag is set to '**Yes**'. If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 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 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: 0. ➢ The eG mranager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.
Outputs of the test	One set of results for every user to each Windows virtual desktop on the KVM server being monitored

Measurements made by the test	Measurement	Measurement Unit	Interpretation
	NTP offset: Indicates the time difference between the local clock and the designated reference clock.	Secs	For a tiny offset, NTP will adjust the local clock; for small and larger offsets, NTP will reject the reference time for a while. In the latter case, the operating system's clock will continue with the last corrections effective while the new reference time is being rejected. After some time, small offsets (significantly less than a second) will be slewed (adjusted slowly), while larger offsets will cause the clock to be stepped (set anew). Huge offsets are rejected, and NTP will terminate itself, believing something very strange must have happened.

3.4.15 Disk Alignment – VM Test

Time synchronization is one of the most important dependencies of windows. A time protocol is responsible for determining the best available time information and converging the clocks to ensure that a consistent time is maintained across systems. By default, windows support a tolerance of plus or minus five minutes for clocks. If the time variance exceeds this setting, clients will be unable to authenticate and in the case of domain controllers, replication will not occur. It implements a time synchronization system based on Network Time Protocol (NTP).

NTP is a fault-tolerant, highly scalable time protocol and it is used for synchronizing computer clocks by using a designated reference clock. A reference clock is some device or machinery that spits out the current time. The special thing about these things is accuracy. Reference clocks must be accurately following some time standard. NTP will compute some additional statistical values based on the current time reported by the reference clock, which will describe the quality of time it sees. Among these values are: offset (or phase), jitter (or dispersion), frequency error, and stability. Thus each NTP server will maintain an estimate of the quality of its reference clocks and of itself.

This test reports the time difference between the reference clock and that of the target environment, and thus helps assess the quality of time seen by the Windows VM. With the help of this test, you can also easily determine whether the reference time changed recently.

This test reports metrics for Windows VMs only.

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

Purpose	Reports the time difference between the reference clock and that of the target environment, and thus helps assess the quality of time seen by the Windows VM
Target of the test	A KVM server host

Agent deploying the test	A remote agent
Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is NULL. 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. Note: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on Windows NT operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default. 7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**. Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights**. Refer to Section 2.4 for more details on

the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the **eG VM Agent**, set the **INSIDE VIEW USING** flag to **eG VM Agent** (Windows). Once this is done, you can set the **DOMAIN, ADMIN USER,** and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain :** If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the DOMAIN field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

     If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

   - **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple DOMAIN names, multiple ADMIN USER names and ADMIN PASSWORDs would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.

   - **If the inside view using flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to none.

9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM

	server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 10. **REPORT POWERED OS** - This flag becomes relevant only if the report by user flag is set to '**Yes**'. If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 11. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**. 12. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG mranager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of results for each disk partition on every Windows VM on a vSphere/ESX host being monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>Disk partition alignment status:
Indicates whether this disk partition is aligned or not.</td>
<td></td>
<td>If the partition is unaligned, this test reports the value Partition is not aligned. For an aligned partition, this test reports the value Partition is aligned.

The numeric values that correspond to the above-mentioned measure values are described in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Partition is aligned</td><td>100</td></tr>
<tr><td>Partition is not aligned</td><td>0</td></tr>
</table>

Note:

By default, this measure reports one of the Measure Values listed in the table above. The graph of this measure however will represent the disk alignment status using the numeric equivalents - 100 or 0.

If a partition is found to be misaligned, you can use the detailed diagnosis of this test to figure out the caption, device ID, logical partition name, and block size of the faulty partition.</td>
</tr>
</table>

4

Monitoring KVM servers with VMs Hosting Desktop Applications

In some environments, the virtual guests hosted on KVM VDI servers may be used to support desktop applications. Administrators of such virtual environments would want to know the following:

- How many desktops are powered on simultaneously on the ESX Server?
- Which users are logged on and when did each user login?
- How much CPU, memory, disk and network resources is each desktop taking?
- What is the typical duration of a user session?
- Who has the peak usage times?
- What applications are running on each desktop?
- Which ESX Server is a virtual guest running on?
- When was a guest moved from an ESX Server? Which ESX Server was the guest moved to?
- Why was the guest migrated? What activities on the ESX host caused the migration?

Using the *KVM VDI server* model (see Figure 4.1), administrators can find quick and accurate answers to all the queries above, and also receive a complete 'desktop view', which allows them to get up, close with the performance of every guest OS hosted by the KVM server and detect anomalies (if any) in its functioning.

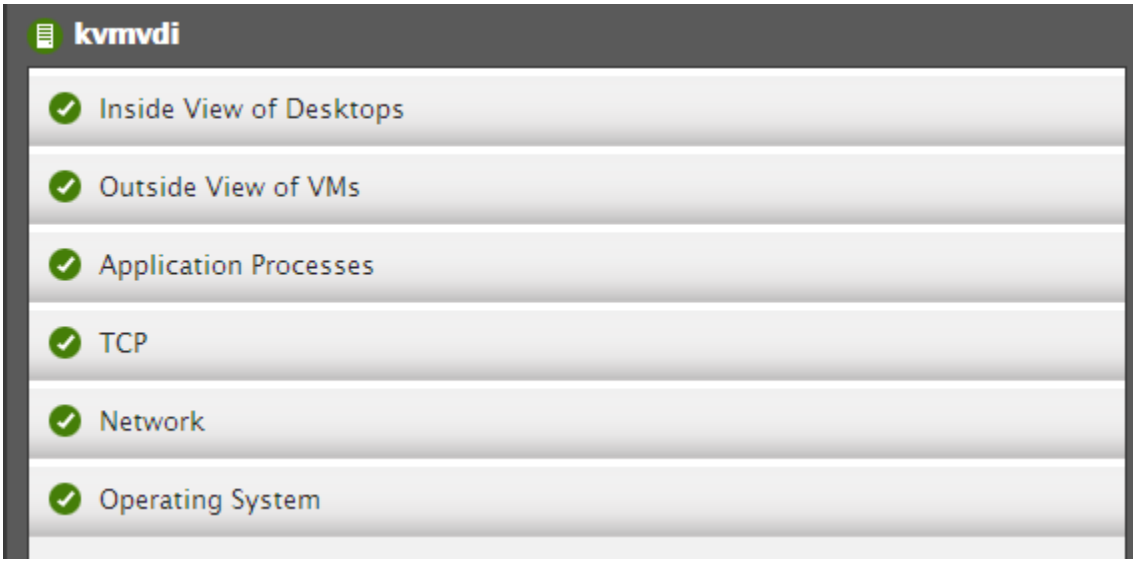


Figure 4.1: The layer model of the KVM VDI server

Each of the layers depicted by Figure 4.1 and the tests associated with the layers are discussed in detail in the sections that follow. Since the last 2 layers of the model have already been dealt with in the previous chapter and the **TCP** and **Application Processes** layer have been dealt in detail in the *Monitoring Unix and Windows server* document, this chapter will discuss the **Outside View of VMs** and the **Inside View of VMs** layers only.

4.1 The Outside View of VMs Layer

The **Outside View of VMs** provides the host operating system's view of the resource usage levels of each of the virtual guests hosted on it. Using the information reported by this test, administrators can:

- Determine which of the guests is taking up more resources (CPU, memory, network, or disk) than the others. This information can help with load balancing or capacity planning. For example, if one of the guests is receiving a very high rate of requests compared to the others, this guest may be a candidate for migration to another ESX server, so as to minimize the impact it has on the other guests on the current ESX server.
- Determine times when sudden or steady spikes in the physical resource utilization are caused by the guest machines

Know which guest systems at what times experienced heavy session loads or unexpected session logouts

Figure 4.2: The tests mapped to the Outside View of VMs layer

4.1.1 KVM VDI Logins Test

This test monitors the user logins to virtual machines and reports the total count of logins and logouts.

Purpose	Monitors the user logins to virtual machines and reports the total count of logins and logouts.
Target of the test	A KVM VDI server host
Agent deploying the test	An internal agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default. 7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).** Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the

eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

     If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the **ADMIN USER** text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the **ADMIN PASSWORD** if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

   - **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

	9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 10. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.** If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 11. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is 1:1. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY.** 12. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG mranager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.
II Outputs of the test	One set of results for the KVM VDI server that is to be monitored

Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Current sessions: Indicates the number of user sessions that are currently active across all the virtual machines.	Number	This is a good indicator of the session load on the VMs.
	New logins: Indicates the number of new logins to the guests.	Number	A consistent zero value could indicate a connection issue.

	Percent new logins: Indicates the percentage of current sessions that logged in during the last measurement period.	Percent	
	Sessions logging out: Indicates the number of sessions that logged out.	Number	If all the current sessions suddenly log out, it indicates a problem condition that requires investigation. The detailed diagnosis of this measure lists the sessions that logged out.

4.1.2 KVM Virtual Machines Test

Whenever users complaint of inaccessibility of their virtual machines, administrators need to promptly determine the reason for the same - is it because the VMs are not running currently? is it because they are not even registered? or is it because they have been moved to another server? The **KVM Virtual Machines** test provides administrators with this information. This test tracks the status and movement of each virtual machine on the target KVM server, reports the number and names of virtual machines in various states, and also captures the migration of virtual machines to other servers.

Purpose	Tracks the status and movement of each virtual machine on the target KVM server, reports the number and names of virtual machines in various states, and also captures the migration of virtual machines to other servers.		
Target of the test	A KVM VDI server host		
Agent deploying the test	An internal agent.		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** – The IP address of the KVM VDI server 3. **PORT** – The port at which the specified host listens		
Outputs of the test	One set of results for the KVM VDI server that is to be monitored		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Registered VMs: Indicates the total number of virtual desktops that have been registered with this KVM VDI server.	Number	The detailed diagnosis of this measure if enabled, lists each registered virtual machine, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.

	Running VMs: Indicates the total number of virtual desktops that are currently running.	Number	A high value is desired for this measure. The detailed diagnosis of this measure if enabled, lists each virtual machine that is currently running, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	VMs not running: Indicates the number of virtual desktops that are currently not running.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that is currently not running, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	Resource blocked VMs: Indicates the number of virtual machines that are currently blocked.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that is blocked, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	Crashed VMs: Indicates the number of virtual desktops that currently crashed.	Number	Ideally, the value of this measure should be 0. The detailed diagnosis of this measure if enabled, lists each virtual machine that crashed, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	No state VMs: Indicates the number of virtual desktops that are currently holding the status 'No state'.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that with the 'No state' status, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	Paused VMs: Indicates the number of virtual desktops that are currently paused.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that is currently paused, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	Added VMs: Indicates the number of virtual desktops that were newly added to the KVM server.	Number	This measure is a good indicator of the load on the KVM Server. The detailed diagnosis of this measure if enabled, lists each virtual machine that is added, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.

	Removed VMs: Indicates the number of virtual desktops that were removed from the KVM server.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine that is removed, the IP address of the virtual machine and the Operating system that is loaded on the virtual machine.
	VMs with users: Indicates the number of virtual desktops on which users are currently logged in.	Number	The detailed diagnosis of this measure if enabled, lists each virtual machine, the IP address of the virtual machine, the Operating system that is loaded on the virtual machine and the user who is logged into the virtual machine. **Note that this measure will be available only for the KVM VDI server.**
	VMs without users: Indicates the number of virtual desktops without any users logged in.	Number	**Note that this measure will be available only for the KVM VDI server.**

4.1.3 KVM VM Details Test

This test monitors the amount of the physical server's resources that each virtual machine on a KVM server is taking up. Using the metrics reported by this test, administrators can determine which virtual machine is taking up most CPU, which virtual machine is generating the most network traffic, which virtual machine is over-utilizing memory, which virtual machine has the maximum disk activity, etc.

Purpose	Monitors the amount of the physical server's resources that each virtual machine on a KVM server is taking up. Using the metrics reported by this test, administrators can determine which virtual machine is taking up most CPU, which virtual machine is generating the most network traffic, which virtual machine is over-utilizing memory, which virtual machine has the maximum disk activity, etc.		
Target of the test	A KVM VDI server host		
Agent deploying the test	An internal agent		
Configurable parameters for the test	1. **TEST PERIOD** - How often should the test be executed 2. **HOST** – The IP address of the KVM VDI server 3. **PORT** – The port at which the specified **HOST** listens		
Outputs of the test	One set of results for each virtual desktop of the KVM VDI server that is to be monitored		
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

<table>
<tr>
<td>test</td>
<td>VM state:
Indicates the current status of this virtual desktop.</td>
<td></td>
<td>The numeric values that correspond to each of the Measure Values that this test can take are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Blocked</td><td>0</td></tr>
<tr><td>Running</td><td>1</td></tr>
<tr><td>Crashed</td><td>2</td></tr>
<tr><td>Nostate</td><td>3</td></tr>
<tr><td>Paused</td><td>4</td></tr>
<tr><td>Shutdown</td><td>5</td></tr>
<tr><td>Shutoff</td><td>6</td></tr>
</table>
Note:
By default, this measure reports one of the Measure Values listed in the table above. The graph of this measure however, represents the status of each VM using the numeric equivalents - ‘0’ to ‘6’.</td>
</tr>
<tr>
<td></td>
<td>Is VM persistent?:
Indicates whether/not the configuration of this virtual desktop is persistent.</td>
<td></td>
<td>The numeric values that correspond to each of the Measure Values that this test can take are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Transient</td><td>0</td></tr>
<tr><td>Yes</td><td>1</td></tr>
<tr><td>Error</td><td>2</td></tr>
</table>
Note:
By default, this measure reports one of the Measure Values listed in the table above. The graph of this measure however, represents whether the configuration of this virtual desktop is persistent or not using the numeric equivalents - ‘0’ to ‘2’.</td>
</tr>
<tr>
<td></td>
<td>Physical CPU utilization:
Indicates the percentage of CPU utilized by this virtual desktop.</td>
<td>Percent</td>
<td>A very high value of this measure indicates that the virtual desktop is currently utilizing high memory resources.</td>
</tr>
</table>

	Virtual CPUs: Indicates the number of virtual CPUs that are allocated to this virtual desktop.	Number	
	Allocated memory: Indicates the amount of memory that is currently allocated to this virtual desktop.	MB	
	Used memory: Indicates the amount of memory that is used by this virtual desktop.	MB	A low value is desired for this measure.
	Free memory: Indicates the amount of memory that is available for use by this virtual desktop.	MB	A high value is desired for this measure. The memory that is used for reclaimable cache is not considered as free memory.
	Memory utilization: Indicates the percentage of memory that is currently utilized by this virtual desktop.	Percent	A high value for this measure indicates that the VM is currently running short of memory resources. Comparing the value of this measure across the VMs will help you identify the VM that is using the maximum memory resources.
	Memory swap-in: Indicates the amount of memory that is being swapped in by the server from the disk for this virtual desktop.	MB	
	Memory swap-out: Indicates the amount of memory that is being swapped to the disk by the server for this virtual desktop.	MB	
	Page faults: Indicates the number of page faults that occurred for the threads matching all processes.	Number	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.
	Unused memory: Indicates the amount of memory that is completely left unused in this virtual desktop.	MB	The value of this measure is the sum total of the *Free memory* and the memory that is used for reclaimable caches.

	Available memory: Indicates the amount of memory that is currently available in this virtual desktop.	MB	
	Balloon memory: Indicates the amount of balloon memory that is currently available for use in this virtual desktop.	MB	Memory ballooning is a virtual memory management technique used to free unused memory. Having multiple virtual machines (VMs) on a single physical server requires virtual memory management techniques to control resource sharing and to prevent shortages. Some processor chipsets use hardware to offload a portion of the virtual memory management work by creating two layers of page tables, the data structure that provides the mapping between virtual addresses and physical addresses. The layers, however, make it difficult for the hypervisor to see a VM's memory contents, how much memory that VM requires or whether the VM is consuming too much memory. Balloon drivers, which are installed in each VM, transfer the memory shortage from the host (where the shortage exists) to the VM. The hypervisor alerts the balloon driver of low memory instances and instructs it to inflate, which locks a set of unused memory in the VM. The hypervisor can then reassign the physical memory to another VM. This swap activity can potentially impact performance depending upon the amount of memory to recoup and/or the quality of the storage IOPS delivered to the VM. In a VMware environment, the balloon driver only activates when memory becomes scarce, so it’s best to have no ballooning activity at all. In a Windows Server environment, the balloon driver allocates RAM to the VM on-demand.
	RSS memory: Indicates the amount of resident memory that is allocated to the process of this virtual desktop.	MB	The resident set size is the portion of a process's memory that is held in RAM. The rest of the memory exists in swap or the filesystem (never loaded or previously unloaded parts of the executable).

	Disk errors: Indicates the number of errors that occurred during the disk reads/disk writes of this virtual desktop.	Number	Ideally, the value of this measure should be zero. Use the detailed diagnosis of this measure to figure out the nature of the errors and the disk on which the errors had occurred.
	Data reads: Indicates the rate at which data is read from the disk of this virtual desktop.	MB/sec	A high value of this measure indicates that the disk is experiencing high I/O activity. The detailed diagnosis of this measure if enabled, lists the name of the disk and the rate at which data is read from this disk.
	Read requests: Indicates the number of read requests handled by the disk of this virtual desktop.	MB/sec	The detailed diagnosis of this measure if enabled, lists the name of the disk and the number of requests handled.
	Data writes: Indicates the rate at which data is written to the disk of this virtual desktop.	MB/sec	The detailed diagnosis of this measure if enabled, lists the name of the disk and the rate at which data is written to the disk.
	Write requests: Indicates the number of write requests handled by the disk of this virtual desktop.	MB/sec	The detailed diagnosis of this measure if enabled, lists the name of the disk and the number of write requests handled by the disk.
	Data transmitted: Indicates the rate at which data is transmitted from this virtual desktop.	Mbps	A high value for this measure indicates that the data transmission is high for this VM. The detailed diagnosis of this measure if enabled, lists the name of the network interface through which data is transmitted and the rate at which data is transmitted.
	Packets transmitted: Indicates the rate at which packets are transmitted from this virtual desktop.	Packets/sec	A high value for this measure indicates that the data transmission is high for this VM. The detailed diagnosis of this measure if enabled, lists the name of the network interface through which the packets are transmitted and the rate at which the packets are transmitted.
	Data dropped during transmission: Indicates the number of data packets that were dropped during transmission.	Number	The detailed diagnosis of this measure if enabled, lists the name of the network interface that dropped the data and the number of data packets dropped.
	Errors during transmission: Indicates the number of errors encountered by this virtual desktop during transmission.	Number	Ideally, the value of this measure should be zero. The detailed diagnosis of this measure if enabled, lists the name of the network interface and the number of errors that were enocuntered.

	Data received: Indicates the rate at which data is received on this virtual desktop.	Mbps	The detailed diagnosis of this measure if enabled, lists the name of the network interface and the rate at which data was received.
	Packets received: Indicates the rate at which data packets were received by this virtual desktop.	Packets/sec	The detailed diagnosis of this measure if enabled, lists the name of the network interface and the rate at which the data packets were received.
	Data dropped during reception: Indicates the number of data packets that were dropped during reception by this virtual desktop.	Packets/sec	Ideally, the value of this measure should be zero. The detailed diagnosis of this measure if enabled, lists the name of the network interface and the number of data packets that were dropped during reception.
	Errors during reception: Indicates the number of errors encountered during data reception by this virtual desktop.	Number	Ideally, the value of this measure should be zero. The detailed diagnosis of this measure if enabled, lists the name of the network interface and the number of errors encountered during data reception.
	Allocated size: Indicates the cumulative allocated size of the disks of this virtual desktop.	MB	The detailed diagnosis of this measure if enabled, lists the name of each disk and the size allocated to each disk.
	Physical size: Indicates the physical size of this virtual desktop.	MB	The detailed diagnosis of this measure if enabled, lists the name of each disk and the physical size that is available in each disk.
	Logical size: Indicates the current logical size of this virtual desktop.	MB	The detailed diagnosis of this measure if enabled, lists the name of each disk and the logical size of each disk.
	Free physical size: Indicates the physical size of this virtual desktop that is currently free.	MB	A high value is desired for this measure. The detailed diagnosis of this measure if enabled, lists the name of each disk and the physical size that is currently free.
	Percentage of physical size utilized: Indicates the percentage of space that is already utilized by this virtual desktop.	Percent	A value close to 100% indicates that the VM is currently running out of physical space. The detailed diagnosis of this measure if enabled, lists the name of each disk and the percentage of space utilized by each disk of the VM.

4.1.4 VDI Applications Test

This test discovers the applications executing on the virtual desktops and reports the availability and resource-usage of each of the desktop applications.

Purpose	Monitors the user logins to virtual machines and reports the total count of logins and logouts.
Target of the test	A KVM VDI server host
Agent deploying the test	A remote agent
Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default. 7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).**

8. Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the

   eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **DOMAIN**, **ADMIN USER**, **ADMIN PASSWORD**, and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

     If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

- **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.
- **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.
11. **REPORT POWERED OS - This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**
12. If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.
13. IS_SHOW_ALL_APPS - To ensure that the test monitors only specific applications executing on the desktops and not all of them, set the **IS_SHOW_ALL_APPS** flag to **No**. Once this is done, then, you need to configure those applications that you want to exclude from the monitoring scope of this test. For this purpose, follow the steps given below:
    - Edit the **eg_tests.ini** file (in the {EG_INSTALL_DIR}\manager\config directory).
    - In the **[EXCLUDE_APPLICATIONS]** section of the file, you will find an entry of the following format:

      VmgApplicationTest={Comma-separated list of applications to be excluded}
    - To the comma-separated application list that pre-exists, append the applications that you want to monitor. For instance, if your test need not monitor notepad.exe, and powerpnt.exe, then, your entry should be:

      VmgApplicationTest=.....,notepad.exe,powerpnt.exe

      Note that the exact application names should be provided, but the extensions (for instance, .exe) can be dispensed with.
    - Finally, save the file.

    On the other hand, if you want to monitor all the applications, then, set the **IS_SHOW_ALL_APPS** flag to Yes.

	14. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: ➢ The eG mranager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.		
Outputs of the test	One set of outputs for every distinct application executing on the virtual desktops		
Measurements of the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Processes running: Indicates the number of instances of this application that is currently executing across all virtual desktops on the target host operating system.	Number	
	CPU usage: Indicates the percentage of physical CPU resources utilized by this application across the guest VMs.	Percent	A very high value of this measure is a cause for concern, as it indicates excessive CPU usage by a single application. This in turn would cause other desktop applications to contend for limited physical resources, thus degrading the performance of those applications and that of the virtual server as a whole.`
	Memory usage: Indicates the percentage of physical memory resources utilized by this application across the guest VMs.	Percent	A very high value of this measure is a cause for concern, as it indicates excessive memory usage by a single application. This in turn would cause other desktop applications to contend for limited physical memory resources, thus degrading the performance of those applications and that of the virtual server as a whole.
	CPU used: Indicates the physical CPU (in Mhz) used up by this application.	Mhz	

Since the remaining test mapped to the **Outside View of VMs** layer - namely, the **VM Connectivity** test has been discussed already, let us move to the **Inside View of VMs** layer.

4.2 The Inside View of Desktops layer

The **Outside View of VMs** layer provides an "external" view of the different VM guests - the metrics reported at this layer are based on what the VMware host is seeing about the performance of the individual guests. However, an external view of the VM guest operating system and its applications may not be sufficient. For instance, suppose one of the disk partitions of the guest operating system has reached capacity. This information cannot be gleaned from host operating system. Likewise, bottlenecks such as a longer process run queue or a higher disk queue length are more visible using an internal monitor. Internal monitoring (from within the guest operating system) also provides details about the resource utilization of different application(s) or processes.

The tests mapped to the **Virtual Desktop** layer provide an "internal" view of the workings of each of the guests - these tests send probes into each of the guest operating systems to analyze how well each guest utilizes the resources that are allocated to it, and how well it handles user sessions, TCP traffic, and network loading.

By default however, clicking on the **Virtual Desktop** layer, does not list the associated tests. Instead, Figure 4.1 appears, which displays the current state of all virtual desktops that have been configured on the monitored ESX host.

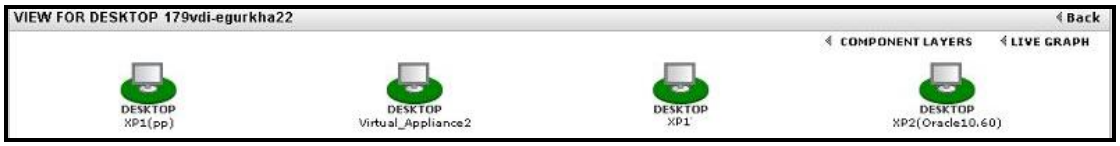


Figure 4.1: The current state of the desktops configured on the KVM VDI server host that is monitored

To return to the layer model of the *KVM VDI* server and view the tests mapped to the **Virtual Desktop** layer, click on the **COMPONENT LAYERS** link in Figure 4.1. The tests depicted by Figure 4.2 then appears.

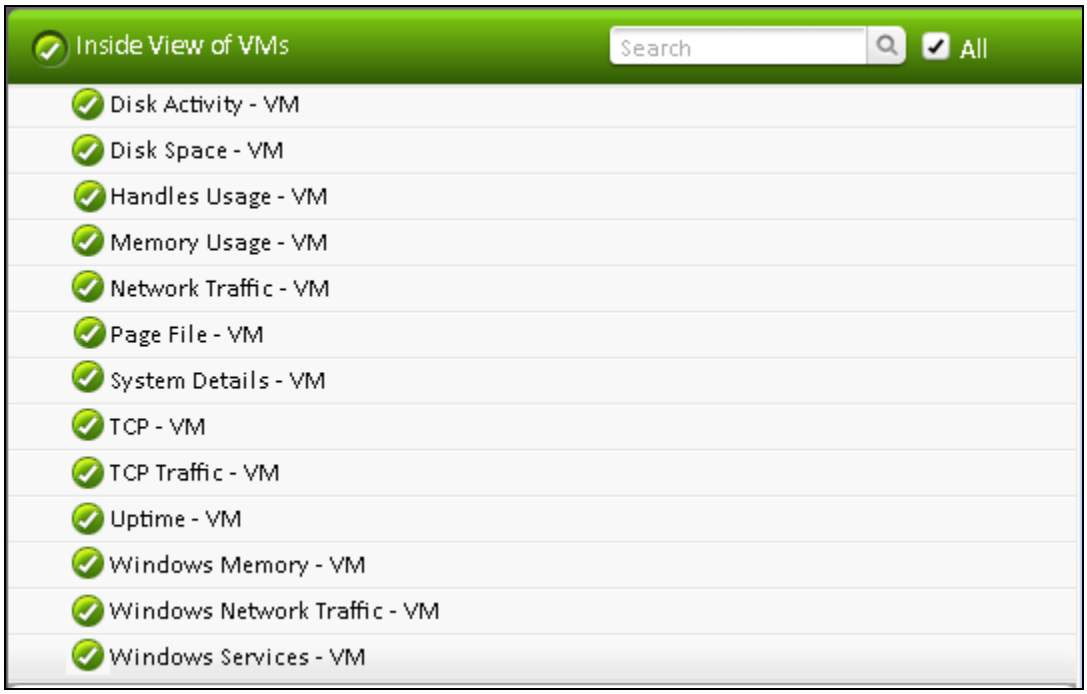


Figure 4.2: The tests associated with the Inside View of Desktops layer of a KVM VDI server

Almost all the tests depicted by Figure 4.2 have already been dealt with in the section on the *KVM server* model. The tests that are specific to the *KVM VDI server* model are alone discussed below.

4.2.1 Terminal to Desktop Connection Test

A Virtual Desktop Infrastructure (VDI) is a shared environment in which multiple users connect to desktops hosted by virtual machines executing on a KVM host from remote terminals using the Remote Desktop Protocol (RDP). One of the key factors influencing user experience in such an environment is the latency seen by the users when connecting to a virtual desktop. High network latencies or packet losses during transmission can cause significant slow-downs in request processing by the desktop. Hence, monitoring latencies between the virtual desktop and individual client terminals is important.

The Terminal to Desktop Connection test is executed by the eG agent on a KVM host. This test auto-discovers the virtual desktops on the KVM host, the users who are currently logged on to each of the virtual desktops, and the IP address from which they are connecting to the virtual desktops. For each user, the test monitors the quality of the link between the client and the virtual desktop.

Using this test, an administrator can identify user sessions that are being impacted by high latencies or by excessive packet drops. In some cases, a virtual desktop may regard a user session as active, even though the network link connecting the user terminal to the virtual desktop has failed. The **Terminal to Desktop Connection** test alerts administrators to such situations.

This test will work only on Windows VMs.

Purpose	Monitors the user logins to virtual machines and reports the total count of logins and logouts.
Target of the test	A KVM VDI server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. Note: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default. 7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).** 8. Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the

eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

     If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

   - **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

	10. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 11. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.** If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 12. **REPORTUNAVAILABILITY** – By default, this flag is set to **No**. This implies that, by default, the test will not report the unavailability of network connection between a user terminal and a virtual desktop. In other words, if the *Packet loss* measure of this test registers the value *100%* for any user, then, by default, this test will not report any measure for that user; under such circumstances, the corresponding user name will not appear as a descriptor of this test. You can set this flag to **Yes**, if you want the test to report and alert you to the unavailability of network connection between a user terminal and a virtual desktop. 13. **PACKETSIZE** - The size of packets used for the test (in bytes) 14. **PACKETCOUNT** - The number of packets exchanged between the virtual desktop and the user terminal during the test 15. **TIMEOUT** - How long after transmission should a packet be deemed lost (in seconds) 16. **PACKETINTERVAL** - Represents the interval (in milliseconds) between successive packet transmissions during the execution of this test.
Outputs of the test	One set of outputs for every user currently connected to the virtual desktop

Measurements of the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Number of sessions: Indicates the current number of sessions for a particular user	Number	The value 0 indicates that the user is not currently connected to the virtual desktop.
	Average delay: Indicates the average delay between transmission of a request by the agent on a virtual desktop and 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 virtual desktop.

	Minimum delay: Indicates the minimum delay between transmission of a request by the agent on a virtual desktop and receipt of the response back from the user terminal.	Secs	A significant increase in the minimum round-trip time is often a sure sign of a poor link between the desktop and a user's terminal.
	Packet loss: Indicates the percentage of packets lost during data exchange between the virtual desktop and the user terminal.	Percent	Comparing the value of this measure across users will enable administrators to quickly and accurately identify users who are experiencing slowdowns because of poor performance on the network links between their terminals and the virtual desktop.

Note

- If the same user is connecting to the virtual desktop from multiple client terminals, the value of the *Number of sessions*, *Average delay*, and *Packet loss* measures will be averaged across all the sessions of that user. The *Minimum delay* measure, on the other hand, will display the least value reported for *Minimum delay* across all the sessions of that user.
- When a user logs out, the number of sessions will be reduced by 1. If the number of user sessions becomes 0, the corresponding entry for that user in the eG user interface will be removed after a short period of time.

4.2.2 Domain Time Sync – VM Test

Time synchronization is one of the most important dependencies of windows. A time protocol is responsible for determining the best available time information and converging the clocks to ensure that a consistent time is maintained across systems. By default, windows support a tolerance of plus or minus five minutes for clocks. If the time variance exceeds this setting, clients will be unable to authenticate and in the case of domain controllers, replication will not occur. It implements a time synchronization system based on Network Time Protocol (NTP).

NTP is a fault-tolerant, highly scalable time protocol and it is used for synchronizing computer clocks by using a designated reference clock. A reference clock is some device or machinery that spits out the current time. The special thing about these things is accuracy. Reference clocks must be accurately following some time standard. NTP will compute some additional statistical values based on the current time reported by the reference clock, which will describe the quality of time it sees. Among these values are: offset (or phase), jitter (or dispersion), frequency error, and stability. Thus each NTP server will maintain an estimate of the quality of its reference clocks and of itself.

This test reports the time difference between the reference clock and that of the target environment, and thus helps assess the quality of time seen by the Windows VM. With the help of this test, you can also easily determine whether the reference time changed recently.

Note

This test reports metrics for Windows VMs only.

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

Purpose	Reports the time difference between the reference clock and that of the target environment, and thus helps assess the quality of time seen by the Windows VM
Target of the test	A KVM VDI server host
Agent deploying the test	A remote agent
Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is NULL. 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. Note While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box.

6. **IGNORE WINNT** - By default, the eG agent does not support the inside view for VMs executing on Windows NT operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default.

7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the **INSIDE VIEW USING** flag is set to **Remote connection to VM (Windows)**.

   Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights**. Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the **eG VM Agent**, set the **INSIDE VIEW USING** flag to **eG VM Agent** (Windows). Once this is done, you can set the **DOMAIN, ADMIN USER,** and **ADMIN PASSWORD** parameters to *none*.

8. **DOMAIN, ADMIN USER, ADMIN PASSWORD,** and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the DOMAIN within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain :** If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the ADMIN USER field and the corresponding password in the ADMIN PASSWORD field. Confirm the password by retyping it in the CONFIRM PASSWORD text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

     If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based*

	Authentication refer to Section 4.3.2 of this document. ➢ **If the guests belong to different domains** - In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple DOMAIN names, multiple ADMIN USER names and ADMIN PASSWORDs would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the Click here hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document. ➢ **If the inside view using flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the domain, admin user, and admin password parameters to none. 9. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*. 10. **REPORT POWERED OS** - This flag becomes relevant only if the report by user flag is set to '**Yes**'. If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently. 11. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**. 12. **DETAILED DIAGNOSIS** - To make diagnosis more efficient and accurate, the eG suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: 0. ➢ The eG mranager license should allow the detailed diagnosis capability ➢ Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.
Outputs of the test	One set of results for every user to each Windows virtual desktop on the KVM server being monitored

Measurements made by the test	Measurement	Measurement Unit	Interpretation
	NTP offset: Indicates the time difference between the local clock and the designated reference clock.	Secs	For a tiny offset, NTP will adjust the local clock; for small and larger offsets, NTP will reject the reference time for a while. In the latter case, the operating system's clock will continue with the last corrections effective while the new reference time is being rejected. After some time, small offsets (significantly less than a second) will be slewed (adjusted slowly), while larger offsets will cause the clock to be stepped (set anew). Huge offsets are rejected, and NTP will terminate itself, believing something very strange must have happened.

4.2.3 Browser Activity – VM Test

When a user complains of a virtual desktop slowdown, administrators will have to instantly figure out if that VM is experiencing a resource crunch, and if so, which process/application on the desktop is contributing to it. One of the common reasons for CPU/memory contentions and handle leaks on a virtual desktop is web browsing! If a user to a virtual desktop browses resource-intensive web sites, it is bound to result in over-usage of the resources allocated to that VM, which in turn degrades the performance of not just that VM but even the other VMs on that host. While the **System Details – VM** test can lead administrators to the exact browser application that is consuming the CPU/memory resources of the VM excessively, it does not provide visibility into the precise websites that were been browsed when the resource contention occurred. This is where the **Browser Activity – VM** test helps. For each web browser that is being accessed by a user per virtual desktop, this test reports how every browser uses the allocated CPU, memory, and disk resources and reveals the number and URLs of the web sites that are being accessed using each browser. This way, the test not only points administrators to resource-hungry browsers, but also indicates which web sites were being accessed using that browser.

Purpose	For each web browser that is being accessed by a user per virtual desktop, this test reports how every browser uses the allocated CPU, memory, and disk resources and reveals the number and URLs of the web sites that are being accessed using each browser
Target of the test	A KVM server host
Agent deploying the test	A remote agent

Configurable parameters for the test	1. **TESTPERIOD** - How often should the test be executed 2. **HOST** - The host for which the test is to be configured. 3. **PORT** - The port at which the host listens. By default, this is *NULL.* 4. **EXCLUDE VMS** - Administrators of some virtualized environments may not want to monitor some of their less-critical VMs - for instance, VM templates - both from 'outside' and from 'inside'. The eG agent in this case can be configured to completely exclude such VMs from its monitoring purview. To achieve this, provide a comma-separated list of VMs to be excluded from monitoring in the **EXCLUDE VMS** text box. Instead of VMs, VM name patterns can also be provided here in a comma-separated list. For example, your **EXCLUDE VMS** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside and outside views of all VMs on a virtual host by default. By providing a comma-separated list of VMs/VM name patterns in the **EXCLUDE VMS** text box, you can make sure the eG agent stops collecting 'inside' and 'outside' view metrics for a configured set of VMs. 5. **IGNORE VMS INSIDE VIEW** - Administrators of some high security VMware environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to **not obtain the 'inside view' of such 'inaccessible' VMs** using the **IGNORE VMS INSIDE VIEW** parameter. Against this parameter, you can provide a comma-separated list of VM names, or VM name patterns, for which the inside view need not be obtained. For instance, your **IGNORE VMS INSIDE VIEW** specification can be: **xp,*lin*,win*,vista*. Here, the * (asterisk) is used to denote leading and trailing spaces (as the case may be). By default, this parameter is set to *none* indicating that the eG agent obtains the inside view of all VMs on a KVM host by default. **Note**: While performing VM discovery, the eG agent will not discover the operating system of the VMs configured in the **IGNORE VMS INSIDE VIEW** text box. 6. **IGNORE WINNT** - By default, the eG agent does not support the *inside view* for VMs executing on **Windows NT** operating systems. Accordingly, the **IGNORE WINNT** flag is set to **Yes** by default. 7. **INSIDE VIEW USING** - By default, this test communicates with every VM remotely and extracts "inside view" metrics. Therefore, by default, the inside view using flag is set to **Remote connection to VM (Windows).** 8. Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 2.4 for more details on the **eG VM Agent**. To ensure that the "inside view" of Windows VMs is obtained using the

eG VM Agent, set the inside view using flag to **eG VM Agent (Windows)**. Once this is done, you can set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

9. **DOMAIN**, **ADMIN USER**, **ADMIN PASSWORD**, and **CONFIRM PASSWORD** - By default, this test connects to each virtual guest remotely and attempts to collect "inside view" metrics. In order to obtain a remote connection, the test must be configured with user privileges that allow remote communication with the virtual guests. The first step towards this is to specify the **DOMAIN** within which the virtual guests reside. The admin user and admin password will change according to the domain specification. Discussed below are the different values that the domain parameter can take, and how they impact the admin user and admin password specifications:

   - **If the VMs belong to a single domain** : If the guests belong to a specific domain, then specify the name of that domain against the domain parameter. In this case, any administrative user in that domain will have remote access to all the virtual guests. Therefore, an administrator account in the given domain can be provided in the **ADMIN USER** field and the corresponding password in the **ADMIN PASSWORD** field. Confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

   - **If the guests do not belong to any domain (as in the case of Linux/Solaris guests):** In this case, specify "none" in the **DOMAIN** field, and specify a local administrator account name in the **ADMIN USER** below.

     Prior to this, you need to ensure that the same local administrator account is available or is explicitly created on each of the virtual machines to be monitored. Then, proceed to provide the password of the **ADMIN USER** against **ADMIN PASSWORD**, and confirm the password by retyping it in the **CONFIRM PASSWORD** text box.

     If key-based authentication is implemented between the eG agent and the SSH daemon of a Linux guest, then, in the admin user text box, enter the name of the user whose **<USER_HOME_DIR>** (on that Linux guest) contains a **.ssh** directory with the *public key file* named **authorized_keys**. The admin password in this case will be the *passphrase* of the *public key*; the default *public key file* that is bundled with the eG agent takes the password *eginnovations*. Specify this as the admin password if you are using the default private/public key pair that is bundled with the eG agent to implement key-based authentication. On the other hand, if you are generating a new public/private key pair for this purpose, then use the *passphrase* that you provide while generating the pair. For the detailed procedure on *Implementing Key-based Authentication* refer to Section 4.3.2 of this document.

   - **If the guests belong to different domains -** In this case, you might want to provide multiple domain names. If this is done, then, to access the guests in every configured domain, the test should be configured with the required user privileges; this implies that along with multiple **DOMAIN** names, multiple **ADMIN USER** names and **ADMIN PASSWORD**s would also have to be provided. To help administrators provide these user details quickly and easily, the eG administrative interface embeds a special configuration page. To access this page, simply click on the *Click here* hyperlink that appears just above the parameters of this test in the test configuration page. To know how to use the special page, refer to Section 3.4.1.1 of this document.

   - **If the INSIDE VIEW USING flag is set to 'eG VM Agent (Windows)'** - In this case, the inside view can be obtained without domain administrator privileges. Therefore, set the **DOMAIN**, **ADMIN USER**, and **ADMIN PASSWORD** parameters to *none*.

10. **REPORT BY USER** - For the *KVM server* monitoring model, the **REPORT BY USER** flag is set to **No** by default, indicating that by default, the guest operating systems on the KVM server are identified using the hostname specified in the operating system. On the other hand, while monitoring KVM VDI environments, this flag is set to **YES** by default; this implies that in case of VDI servers, by default, the guests will be identified using the login of the user who is accessing the guest OS. In other words, in VDI environments, this test will, by default, report measures for every *username_on_virtualmachinename*.

11. **REPORT POWERED OS** - **This flag becomes relevant only if the REPORT BY USER flag is set to 'Yes'.**

    If the **REPORT POWERED OS** flag is set to **Yes** (which is the default setting), then this test will report measures for even those VMs that do not have any users logged in currently. Such guests will be identified by their *virtual machine name* and not by the *username_on_virtualmachinename*. On the other hand, if the **REPORT POWERED OS** flag is set to **No**, then this test will not report measures for those VMs to which no users are logged in currently.

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

Outputs of the test: One set of results for every user to each Windows virtual desktop on the KVM server being monitored

Measurements made by the test

Measurement	Measurement Unit	Interpretation
Running browser instances: Indicates the number of instances of this browser currently running on this virtual desktop.	Number	Use the detailed diagnosis of this measure to know how much resources were utilized by each instance of a browser, so that the resource-hungry instance can be isolated.

	Recent web sites: Indicates the number of websites that were accessed using this browser on this virtual desktop during the last measurement period.	Number	Use the detailed diagnosis of this measure to know which web sites are being accessed using a browser.
	CPU utilization: Indicates the percentage CPU usage of this browser on this virtual desktop.	Percent	Compare the value of this measure across browsers to know which browser consumed the maximum CPU on a desktop. If the value of this measure is close to 100% on that desktop, it indicates excessive CPU usage by the browser. You may then want to use the detailed diagnosis of the *Recent web sites* measure to know which web sites are being accessed using that browser, which caused CPU usage to soar.
	Memory used: Indicates the percent usage of memory by this browser on this virtual desktop.	Percent	Compare the value of this measure across browsers to know which browser consumed the maximum memory on a desktop. If the value of this measure is close to 100% on that desktop, it indicates excessive memory usage by the browser. You may then want to use the detailed diagnosis of the *Recent web sites* measure to know which web sites are being accessed using that browser, which caused CPU usage to soar.
	Handles used: Indicates the number of handles opened by this browser on this virtual desktop.	Number	Compare the value of this measure across browsers to know which browser opened the maximum number of handles on a desktop. If the value of this measure consistently increases on that desktop, it indicates that the corresponding browser is leaking memory. You may then want to use the detailed diagnosis of the *Recent web sites* measure to know which web sites are being accessed using that browser, which caused the memory leak.
	Disk reads: Indicates the rate at which this browser read from the disks supported by this virtual desktop.	KB/Sec	A high value for these measures indicates that the browser is generating high disk I/O. You may then want to use the detailed diagnosis of the *Recent web sites* measure of this browser to know which web sites on the browser are responsible for the high disk I/O.
	Disk writes: Indicates the rate at which this browser read from the disks of this virtual desktop.	KB/Sec	

	Disk IOPS: Indicates the rate of read and write operations performed by this browser on the disks of this virtual desktop.	Operations/Sec	A high value for this measure indicates that the browser is generating high disk I/O. You may then want to use the detailed diagnosis of the *Recent web sites* measure of this browser to know which web sites on the browser are responsible for the high disk I/O.
	Page faults: Indicates the rate at which page faults by the threads executing in this browser are occurring on this virtual desktop.	Faults/Sec	Ideally, the value of this measure should be low. A high value for a browser is a cause for concern. You may then want to use the detailed diagnosis of the *Recent web sites* measure of this browser to know which web sites on the browser are responsible for page faults.

The detailed diagnosis of the *Running browser instances* measure reveals the process ID of each browser instance that is currently running on the virtual desktop and the resource usage of each instance. This way, you can easily and accurately identify the instance that is consuming resources excessively.

List of browser instances and their performance

TIME	PROCESSID	CPUUTIL(%)	MEMUTIL(%)	HANDLES COUNT	DISK IO READ(KB/SEC)	DISK IO WRITE(KB/SEC)	DISK IOPS(OPERATIONS/SEC)	PAGE FAULTS(FAULTS/SEC)	WEBSITE TITLE	
Oct 25, 2013 18:41:10										
	4404	0	0.7658	544	0	0	0	0	Yahoo India - Windows Internet Explorer	-
	4188	0	0.4282	527	0	0	0	0	-	-
	6132	0	0.465	450	0	0	0	0	Google - Windows Internet Explorer	https:/ gws_rd=

Figure 4.3: The detailed diagnosis of the Running browser instances measure

The detailed diagnosis of the *Recent web sites* measure reveals the names and URLs of the web sites that are being accessed using a browser.

Details of websites that are last browsed

TIME	WEBSITE TITLE	WEBSITE URL
Oct 25, 2013 18:41:10		
	Google - Windows Internet Explorer	https://www.google.co.in/?gws_rd=cr&ei=DBhmUpWMC4nYrQez3IDYBQ
	Yahoo India - Windows Internet Explorer	-

Figure 4 4: The detailed diagnosis of the Recent web sites measure

4.3 Troubleshooting

4.3.1 Troubleshooting the Failure of the eG Remote Agent to Obtain the 'Inside View' of a Windows VM

If the eG remote agent is unable to obtain the inside view of a Windows VM, then, first check whether the agent is able to connect to the problem VM. The steps for performing this check will vary depending upon the operating system on which the remote agent executes.

Note

The steps discussed below apply only when the following conditions are fulfilled:

- The KVM servers being monitored should be configured only with Windows VMs;
- All the Windows VMs should belong to a single domain only - i.e., the inside view tests for the target KVM server should be configured with a single **ADMIN USER** and **ADMIN PASSWORD** only.

If the eG agent is operating on a Unix host, then the following steps will apply:

1. Login to the Unix host as eG install user.
2. Go to the shell prompt and switch to the **/opt/egurkha/lib** directory.
3. Set the **CLASSPATH** by issuing the following commands at the prompt:

   CLASSPATH=.:eg_agent.jar:eg_util.jar:$CLASSPATH

   export CLASSPATH
4. Next, set the JRE path by issuing the following commands:

   PATH=/opt/egurkha/jre/bin:$PATH

   export PATH
5. Finally, issue the following command to try connecting to the Windows VM:

   java EgWinConnect -user <username> -password <password> -domain <domainname> -IP <IP_address>

   In this command:

   - Substitute the **<username>** and **<password>** variables with the values that you have configured for the **ADMIN USER** and **ADMIN PASSWORD** parameters (respectively) for all inside view tests. While the admin user value can be easily obtained from the test configuration page in the eG administrative interface, the admin password cannot be so obtained, as it would be in an encrypted format in the user interface. To know the password therefore, do the following:
     - Open the **eg_agents.ini** file in the **/opt/egurkha/gent/config** directory on the agent host.
     - Look for entries related to the inside view tests that the agent has executed on the other Windows VMs on the target vSphere/ESX server.

- These entries will typically include a **-admin_password** parameter, which will be followed by the decrypted admin password.
- Make a note of the decrypted password, and substitute the **<password>** variable in the **java EgWinConnect** command with it.

- Next, substitute the **<domainname>** variable with the value that you passed to the domain parameter in the test configuration page, and the **<IP_address>** with the IP address of the problematic Windows VM.
- If this command fails, it is a clear indication that the remote agent is unable to communicate with the Windows VM. You will then have to investigate the reasons for the same and fix them in order to ensure that the agent is able to obtain the "inside view" of that VM.

If the eG agent is operating on a Windows host, then the following steps will apply:

1. Login to the Windows host on which the eG agent is executing.
2. Go to the command prompt and switch to the **<EG_INSTALL_DIR>\lib** directory.
3. Issue the following command to set the path to the **<EG_INSTALL_DIR>\JRE\bin** directory.

   set path=<EG_INSTALL_DIR>\JRE\bin

4. Set the **CLASSPATH** by issuing the following command at the prompt:

   set classpath=.<EG_INSTALL_DIR>\lib\eg_agent.jar;<EG_INSTALL_DIR>\lib\eg_util.jar

5. Finally, issue the following command to try connecting to the Windows VM:

   java EgWinConnect -user <username> -password <password> -domain <domainname> -IP <IP_address>

   In this command:

   - Substitute the **<username>** and **<password>** variables with the values that you have configured for the **ADMIN USER** and **ADMIN PASSWORD** parameters (respectively) for all inside view tests. While the admin user value can be easily obtained from the test configuration page in the eG administrative interface, the admin password cannot be so obtained, as it would be in an encrypted format in the user interface. To know the password therefore, do the following:
     - Open the **eg_agents.ini** file in the **<EG_INSTALL_DIR>\agent\config** directory on the agent host.
     - Look for entries related to the inside view tests that the agent has executed on the other Windows VMs on the target vSphere/ESX server.
     - These entries will typically include a **-admin_password** parameter, which will be followed by the decrypted admin password.
     - Make a note of the decrypted password, and substitute the **<password>** variable in the **java EgWinConnect** command with it.
   - Next, substitute the **<domainname>** variable with the value that you passed to the domain parameter in the test configuration page, and the **<IP_address>** with the IP address of the problematic Windows VM.

   A sample command is given below:

   C:\eGurkha\lib>java EgWinConnect -user eguser -password C13120CB9E5D4B1423419897808BAE65 -domain mas -ip 192.168.10.216

6. If the command is successful, the following output will appear:

```
*******************************************************************************
Attempt to connect and execute for 192.168.10.216
Output is [[Ok, , , , , Windows IP Configuration, , , , , , , , , Ethernet adapt
er Local Area Connection 2:, , , , , ,    Connection-specific DNS Suffix  . : ,
, ,    IP Address. . . . . . . . . . . . : 192.168.10.216, , ,    Subnet Mask .
. . . . . . . . . . . : 255.255.255.0, , ,    Default Gateway . . . . . . . . . :
192.168.10.2, , , , EgDone 0(0x0)], []]
*******************************************************************************
```

7. The command may fail when it encounters one of the following errors. The reasons for these errors and the recommended resolution for the same have been provided below.

Error	Reason	Fix
Counldn't connect to \\<IP_of_Windows_VM>\ADMIN$ **Logon failure: unknown user name or bad password**	Occurs if the inside-view tests have been configured with an incorrect **DOMAIN**, invalid **ADMIN USER** name, or a wrong **ADMIN PASSWORD**.	Reconfigure the tests with the valid credentials of a *domain administrator*
Counldn't connect to \\<IP_of_Windows_VM>\ADMIN$ **The network path was not found.**	➢ Can occur if the **ADMIN$** share has not been enabled on the target Windows VM; ➢ Can occur if the Windows firewall is blocking connection to the VM, or if File/Print Sharing has not been enabled yet.	➢ Enable the ADMIN$ share on the target VM. ➢ Provide domain administrator with full access to the ADMIN$ share. ➢ Try connecting to the Windows VM remotely; if the problem persists: • Reconfigure the Windows firewall to allow communication between the remote agent and the Windows VM; • Reconfigure the Windows firewall to allow File/Print Sharing
Couldn't copy service to \\<IP_of_Windows_VMs>\ADMIN$ **Access is denied.**	➢ Can occur if the **ADMIN$** share exists but the *domain administrator* does not have permission to access the shared folder; ➢ Can occur if the **ADMIN$** share exists, but the user with full access to the shared folder is not the *domain administrator*.	Provide *domain administrator* with full access to the **ADMIN$** share.

4.3.2 Troubleshooting the Failure of the eG Remote Agent to Connect to or Report Measures for Linux Guests

By default, the eG agent uses secure shell (SSH) to connect to Linux guests, and collect performance metrics from them. Password Authentication is the default method for SSH connections in eG Enterprise. If the eG agent fails to

report measures for a Linux guest or is unable to connect to a guest, it could imply that the Linux guest does not support SSH or that password authentication is not supported by the SSH daemon running on the Linux guest. Under such circumstances, you can perform either of the following:

- Enable Password Authentication in the SSH daemon on the Linux guest; or,
- Implement Key-Based Authentication between eG agent and the SSH daemon of the Linux guest.

If you pick option (1), then follow the steps given below to enable password authentication:

- Login to the Linux guest to be monitored.
- Edit the **sshd_config** file in the **/etc/ssh** directory.
- Check whether the **Password Authentication** flag in the **sshd_config** file is set to **no**. If so, set it to **yes**.
- Then, save the file and restart/signal the SSH daemon (eg., using **kill -1 <sshd_config PID>**).

On the contrary, if you choose to enable key-based authentication [i.e, option (2)], then you will have to generate a public/private key pair. A public/private key pair is available in the **<EG_INSTALL_DIR>\agent\sshkeys** directory (on Windows; on Unix, this will be **/opt/egurkha/agent/sshkeys**) of the eG agent. While the private key is available in the file named **id_rsa**, the public key is contained within the file **authorized_keys**. You now have the option to proceed with the default keys or generate a different key pair. If you decide to go with the keys bundled with the eG agent, do the following:

- To enable key-based authentication, the private key should remain in the **<EG_INSTALL_DIR>\agent\sshkeys** directory (on Windows; on Unix, this will be **/opt/egurkha/agent/sshkeys**), and the public key should be copied to each of the Linux guests to be monitored. To achieve this, first login to the Linux guest to be monitored as the eG user.
- Create a directory named **.ssh** in the **<USER_HOME_DIR>** on the guest operating system, using the command: **mkdir ~/.ssh**.
- Next, copy the **authorized_keys** file from the **<EG_INSTALL_DIR>\agent\sshkeys** directory (on Windows; on Unix, this will be **/opt/egurkha/agent/sshkeys**) on the eG remote agent host to the **<USER_HOME_DIR>/.ssh** directory on the Linux guest.

  Make sure that the permission of the **.ssh** directory and the **authorized_keys** file is **700**.

- Finally, on the eG manager host, edit the **<EG_INSTALL_DIR>\manager\config\eg_tests.ini** file. Against the **EgJavaSSHKeyFile** parameter, enter: **agent/sshkeys/id_rsa.pub**, and save the file.

On the other hand, if you want to generate a new key pair, then do the following:

- Login to any Linux host in your environment (even a Linux VM) as an eG user.
- From the **<USER_HOME_DIR>**, execute the command: **ssh-keygen -t rsa**. Upon executing the command, you will be requested to specify the full path to the file to which the key is to be saved. By default, a directory named **.ssh** will be created in the **<USER_HOME_DIR>**, to which the key pair will be saved. To go with the default location, simply press **Enter**.

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/egurkha/.ssh/id_rsa):
```

- Next, you will be prompted to provide a pass phrase. Provide any pass phrase of your choice.

```
Enter passphrase (empty for no passphrase): eginnovations
Enter same passphrase again: eginnovations
```

- If the key pair is created successfully, then the following messages will appear:

```
Your identification has been saved in /home/egurkha/.ssh/id_rsa.
Your public key has been saved in /home/egurkha/.ssh/id_rsa.pub.
The key fingerprint is:
09:f4:02:3f:7d:00:4a:b4:6d:b9:2f:c1:cb:cf:0e:e1 dclements@sde4.freshwater.com
```

- The messages indicate that the private key has been saved to a file named **id_rsa** in the **<USER_HOME_DIR>/.ssh**, and the public key has been saved to a file named **id_rsa.pub** in the same directory. Now, to enable key-based authentication, login to the Linux guest to be monitored as the eG user.
- Create a directory named **.ssh** in the **<USER_HOME_DIR>** on the guest operating system, using the command: **mkdir ~/.ssh**.
- Next, copy the **id_rsa.pub** file from the **<USER_HOME_DIR>/.ssh** directory on the Linux host to the **<USER_HOME_DIR>/.ssh** directory on the Linux guest.
- Ensure that the **id_rsa.pub** file on the Linux guest is renamed as **authorized_keys**.
- Repeat this procedure on every Linux guest to be monitored.
- Then, lock the file permissions down to prevent other users from being able to read the key pair data, using the following commands:

  chmod go-w ~/
  chmod 700 ~/.ssh
  chmod go-rwx ~/.ssh/*

- Finally, on the eG manager host, edit the **<EG_INSTALL_DIR>\manager\config\eg_tests.ini** file. Against the **EgJavaSSHKeyFile** parameter, enter: **agent/sshkeys/id_rsa.pub**, and save the file.

Instead of choosing between the authentication modes (Password or Key-based), you can also disable the usage of the Java SSH client, and use **plink** to connect to Linux guests. To achieve this, follow the steps given below:

- Edit the **eg_tests.ini** file in the **/opt/egurkha/manager/config** directory (on Unix; on Windows, this will be **<EG_INSTALL_DIR>\manager\config** directory).
- Set the **JavaSSHForVm** flag in the **[AGENT_SETTINGS]** section of the file to **false**; by default, this is set to **true** indicating that the eG agent uses Java SSH by default. By setting the flag to **false**, you can ensure that the eG agent does not use Java SSH, and instead uses the **plink** command to connect to Linux guests.
- The **plink** command exists in the **<EG_INSTALL_DIR>\lib\vmgfiles** directory (on Windows; on Unix, this will be **/opt/egurkha/lib/vmgfiles**) of the eG agent. To use the **plink** command, you need to explicitly configure the SSH keys, so that the eG agent is able to communicate with the Linux guests using SSH. To do this, follow the steps given below:
  - Go to the command prompt and switch to the directory containing the **plink** command.
  - Then, execute the **plink** command to connect to any of the Linux-based virtual machines on the vSphere host. The syntax for the **plink** command is as follows:

    plink -ssh <user>@<IP_of_target_host> <command>

    For example, assume that you want to connect to the virtual machine, **192.168.10.7**, as user **john** with password **john**, to know its hostname. The syntax of the **plink** command in this case will be:

plink -ssh john@192.168.10.7 hostname, where **hostname** is the command to be executed on the remote host for extracting its hostname.

- To ensure that you do not connect to an imposter host, **SSH2.x** presents you with a unique host key fingerprint for that host, and requests your confirmation to save the displayed host key to the cache.

```
The server's host key is not cached in the registry. You have no guarantee
that the server is the computer you think it is.
The server's rsa2 key fingerprint is:<host key>
If you trust this host, enter "y" to add the key to PuTTY's cache and carry
on connecting.
If you want to carry on connecting just once, without adding the key to the
cache, enter "n".
If you do not trust this host, press Return to abandon the connection.
Store key in cache? (y/n) y
```

Once you confirm the host key storage and provide the user's password to connect to the virtual guest, this message will not appear during your subsequent attempts to connect to any Linux-based virtual machine on the monitored vSphere/ESX host. In other words, the eG agent will be able to execute tests on all Linux guests on the target ESX host without any interruption. Therefore, press **y** to confirm key storage.

4.3.3 Troubleshooting the Failure of the eG Remote Agent to Connect to or Report Measures for Windows 2008 or Windows Vista VMs

The remote agent may not be able to connect to or collect inside view metrics from a VM running Windows 2008 or Windows Vista, if the **User Access Control** (UAC) feature is enabled on those VMs. In such a case, do the following on each of those VMs to enable the remote agent to connect to them:

1. Click **Start**
2. Type **REGEDIT**
3. Press **Enter**
4. In the left pane, browse to the following folder:

   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
5. Right-click on a blank area in the right pane.
6. Click **New**.
7. Click **DWORD Value**.
8. Type **LocalAccountTokenFilterPolicy**.
9. Double-click the item you just created,
10. Type **1** into the box that appears.
11. Click **OK**.
12. Restart the virtual machine.

5 Conclusion

This document has described in detail the monitoring paradigm used and the measurement capabilities of the eG Enterprise suite of products with respect to the **KVM servers**. For details of how to administer and use the eG Enterprise suite of products, refer to the user manuals.

We will be adding new measurement capabilities into the future versions of the eG Enterprise suite. If you can identify new capabilities that you would like us to incorporate in the eG Enterprise suite of products, please contact support@eginnovations.com. We look forward to your support and cooperation. Any feedback regarding this manual or any other aspects of the eG Enterprise suite can be forwarded to feedback@eginnovations.com.