Enabling Service Excellence

Configuring and Monitoring Oracle VDI

eG Enterprise v5.6

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

Table of Contents

Table of Figures

Chapter

1

Configuring and Monitoring the Oracle VirtualBox

Oracle VDI (Virtual Desktop Infrastructure) provides desktop virtualization to replace personal computers with virtual machines (VMs) on a server. Users can access these VMs though any RDP client, or through the web via Sun Secure Global Desktop (SGD).

Oracle Virtual Desktop Infrastructure is made up of four main components: virtualization platform, session management (Oracle VDI Core), desktop access clients, and storage.

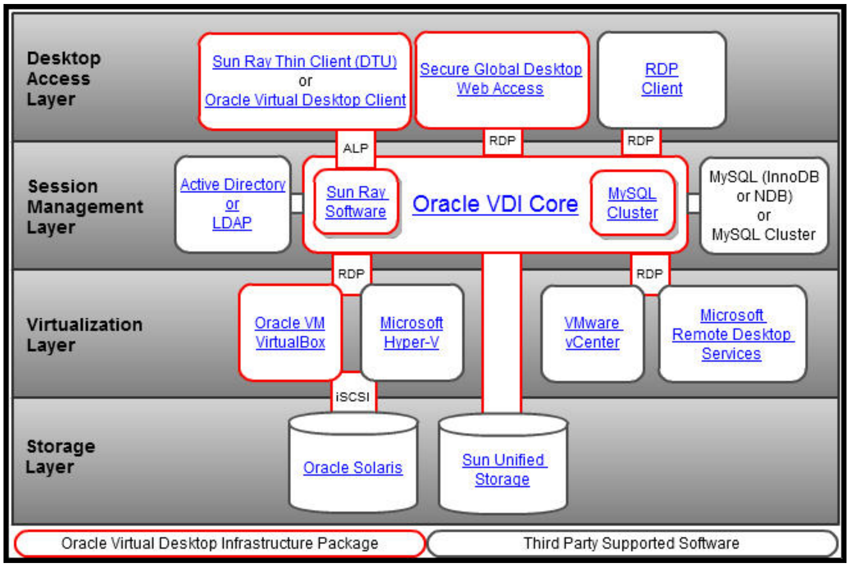


Figure 1.1: Architecture of the Oracle Virtual Desktop Infrastructure

The basis for the Oracle Virtual Desktop Infrastructure architecture is the virtualization platform. In addition to creating and storing virtual machines, the virtualization platform offers the core functionality needed for virtual desktop management such as starting, stopping, and snapshotting virtual machines. Oracle Virtual Desktop Infrastructure 3.2 supports Oracle VM VirtualBox (the Oracle VDI Hypervisor), VMware vCenter, Microsoft Hyper-V, and Microsoft Remote Desktop Services as virtualization platforms.

Oracle VM VirtualBox is cross-platform x86 virtualization software that extends the power of your existing computers to run multiple operating systems, on the same hardware, at the same time, and alongside your existing applications.

Oracle VM VirtualBox includes a hypervisor for the host platform, an application programming interface (API) and software development kit (SDK) for managing guest virtual machines, a command-line tool for managing guests locally, a web service for remote management of guests, a wizard-style graphical tool to manage guests, a graphical console for displaying guest applications on the local host, and a built-in Remote Desktop Protocol (RDP) server that provides complete access to a guest from a remote client. At the core is the hypervisor, implemented as a *ring 0* (privileged) kernel service. Figure 1.2 shows the relationships between all of these components.

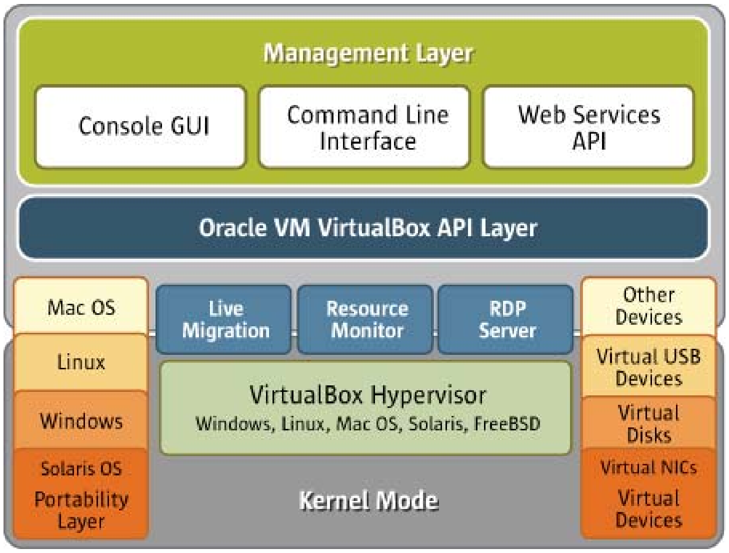


Figure 1.2: Architecture of an Oracle VirtualBox

The kernel service consists of a device driver named *vboxsrv*, which is responsible for tasks such as allocating physical memory for the guest virtual machine, and several loadable hypervisor modules for things like saving and restoring the guest process context when a host interrupt occurs, turning control over to the guest OS to begin execution, and deciding when VT-x or AMD-V events need to be handled.

The hypervisor does not get involved with the details of the guest operating system scheduling. Instead, those tasks are handled completely by the guest during its execution. The entire guest is run as a single process on the host system and will run only when scheduled by the host. If they are present, an administrator can use host resource controls such as scheduling classes and CPU caps or reservations to give very predictable execution of the guest machine.

Additional device drivers will be present to allow the guest machine access to other host resources such as disks, network controllers, and audio and USB devices. In reality, the hypervisor actually does little work. Rather, most of the interesting work in running the guest machine is done in the guest process. Thus the host's resource controls and scheduling methods can be used to control the guest machine behavior.

In addition to the kernel modules, several processes on the host are used to support running guests. All of these processes are started automatically when needed.

1.1 Challenges in Monitoring the Oracle VirtualBox

What makes monitoring an Oracle VDI infrastructure a challenge is the large number of virtual desktops that will typically be configured on the VirtualBoxes and the large number of users to the desktops. While it can be very difficult to keep track of which user is accessing which desktop on which VirtualBox, the live and automatic migration of desktops to other VirtualBoxes (if any) in the environment only compounds the problem. To make matters worse, the users to the VDI service demand from the virtual desktops the same quality of service that they are used to receiving from their physical desktops. This means, quick access, uninterrupted connectivity, and stable operations will be the criteria on which the user experience with the VDI service will be judged. Non-availability of a desktop when a user needs it, or slowdowns in desktop operations caused by a resource contention at the desktop-level or at the host-level may result in a deluge of user complaints and a bevy of dissatisfied users. To avoid this, service desk should be able to:

- Continuously monitor the user activity to the virtual desktops operating on a VirtualBox;
- Know which user accessed the desktop on which VirtualBox at what time;
- Track the powered-on state of desktops;
- Study the resource usage patterns of the virtual desktops to nail the root-cause of resource contentions - is it owing to a resource-hungry desktop? or a resource-starved host?;
- Promptly alert users to potential resource drains on desktops or a sudden change in the desktop state, much before users notice the difference!

1.2 eG's Solution to Oracle VirtualBox Monitoring

The specialized *Oracle VirtualBox* monitoring model addresses all the requirements outlined above and more! This 100%, web-based solution employs a single eG agent to perform detailed 'In-N-Out' monitoring of the virtual desktops operating on an Oracle VirtualBox.

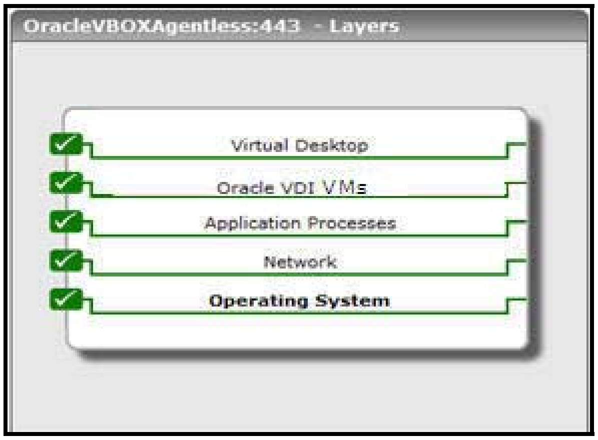


Figure 1.3: The layer model of Oracle VirtualBox

The metrics collected by this eG agent report on the percentage of the Oracle VirtualBox host's resources that each of 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, the applications they are accessing, and the resource usage of 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.

In addition, the same agent can also track the critical processes running on the Oracle VirtualBox host and their resource usage, the network connection to the host, and the TCP connectivity of the host, and thus report on the overall health of the host.

The agent is also capable of capturing and reporting on the live migration of desktops from one VirtualBox to another.

Based on the monitoring approach chosen, you can deploy this eG agent on the VirtualBox itself or on any remote Windows/Linux/Solaris host in the environment. Section 1.3 discusses both these deployment models in detail.

1.3 Agent Deployment Models

eG Enterprise allows administrators the flexibility to choose between the *agent-based* and *agentless* approaches to monitoring the *Oracle VirtualBox*.

1.3.1 The Agent-based Monitoring Approach

The agent-based approach **requires that the eG agent be installed on the host operating system of the VirtualBox.** Since Oracle VDI comes bundled with an Oracle VirtualBox that runs an Oracle Solaris operating system, you need to install a **Solaris eG agent** on the host. To know how to install an eG agent on Solaris, refer to the *eG Installation Guide*.

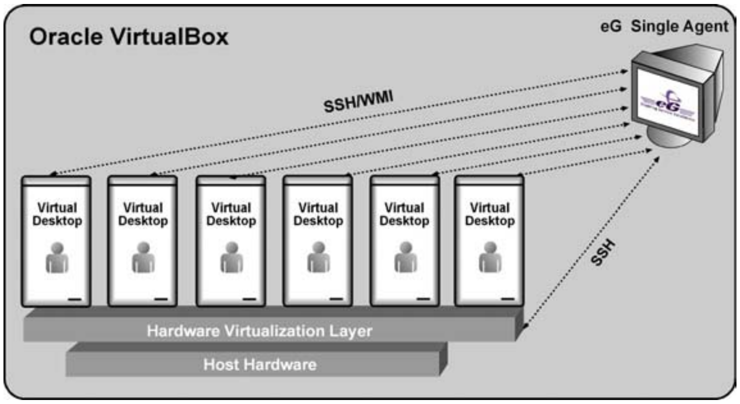


Figure 1.4: Agent-based monitoring of the Oracle VirtualBox

This agent should then be configured to communicate with the Oracle VirtualBox via SSH and run privileged Virtual Desktop Access (VDA) commands on the Oracle VirtualBox to determine the health of the host, to discover the IP address and operating system of each of the guests on the host, and to report how each guest has utilized the host's physical resources (i.e., *outside view*). For connecting to the target Oracle VirtualBox via SSH, the eG agent has to be configured with the credentials of a user with the required privileges. Also, to enable the eG agent to run VDA commands on the host, a **sudo** package has to be installed on the VirtualBox. To know how, refer to Section 1.4 below.

Once the guests are discovered, the eG agent remotely communicates with each guest using SSH/WMI (depending upon the operating system of the guest) to obtain the "inside view" of every guest. To establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting “inside view” metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM, which allows the eG agent on the service console to collect “inside view” metrics from the VMs **without domain administrator rights.** Refer to Section 1.5 for more details on the **eG VM Agent**.

1.3.2 The Agentless Monitoring Approach

The **agentless approach** to monitoring the Oracle VirtualBox involves the following:

- Deploying the eG agent on a remote system running Microsoft Windows or Linux or Solaris;
- Configuring the remote eG agent to communicate with the target VirtualBox via SSH;
- Configuring the remote eG agent to run Virtual Desktop Access (VDA) commands on the VirtualBox to perform guest discovery and to collect host-level and 'outside view' metrics; to run these commands, you need to install a **sudo** package on the VirtualBox - refer to Section 1.4 to know how.
- Configuring the remote eG agent to collect performance metrics from each of the guest VMs configured on the VirtualBox using SSH/WMI; by default, the eG agent uses SSH/WMI (depending upon the virtual OS to be monitored) to communicate remotely with the virtual machines on the VirtualBox (see Figure 1.5) and collect metrics. To establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting “inside view” metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent** on every Windows VM, which allows the eG agent to collect “inside view” metrics from the VMs **without domain administrator rights.** Refer to Section 1.5 for more details on the **eG VM Agent**.

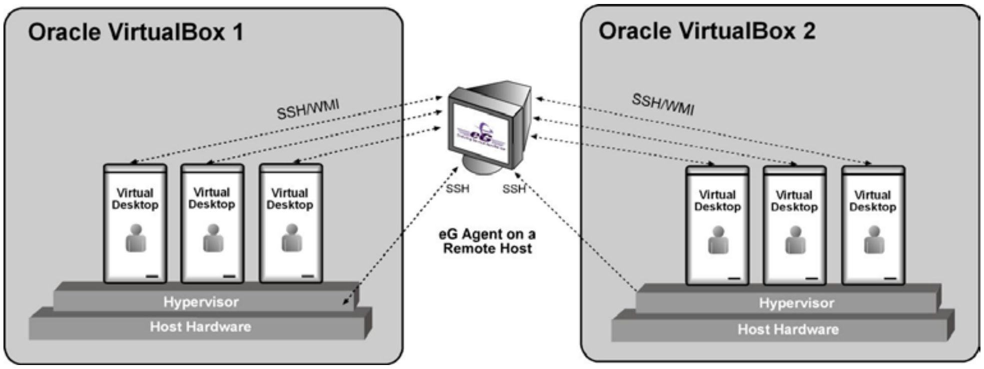


Figure 1.5: The Agentless approach to monitoring the Oracle VirtualBox

Regardless of the monitoring approach chosen, certain pre-requisites need to be fulfilled before attempting to monitor the Oracle VirtualBox. Section 1.4 below discusses these pre-requisites.

1.4 Pre-requisites for Monitoring the Oracle VirtualBox

This section details the pre-requisites that need to be fulfilled for monitoring an Oracle VirtualBox in an agent-based and an agentless manner.

1.4.1 Pre-requisites for Monitoring the Oracle VirtualBox in an Agent-based Manner

1.4.1.1 General Pre-requisites

Enable the eG agent to communicate with the eG manager port (default: 7077).

1.4.1.2 Pre-requisites for Auto-Discovering VMs and Obtaining their "Outside View"

- Ensure that the eG agent is able to connect to the target VirtualBox via SSH.
- Make sure that the SSH port (default: 22) is opened for communication between the eG agent and the Oracle VirtualBox.
- Configure all the tests that the eG agent executes with the name and password of a user who has the right to access the target Oracle VirtualBox via SSH.
- To enable the eG agent to run VDA commands on the VirtualBox, install the **sudo** package on the VirtualBox. To install this package, do the following:
    - Login to the Solaris system hosting the VirtualBox as a *root* user.
    - To download the **sudo** package, connect to the URL: http://sysinfo.bascomp.org/solaris/installing-sudo-on-solaris/
    - If the Solaris processor is Intel based, download the file **TCMsudo-1.8.2-i386.pkg.gz** from the web site mentioned above. On the other hand, if the Solaris host uses a SPARC processor instead, download the file **TCMsudo-1.8.2-sparc.pkg.gz** from the web site.
    - Download the chosen file to any location on the VirtualBox host (say, **/tmp**).
    - From the Solaris prompt, switch to the directory hosting the downloaded package and unzip the compressed package using the following command:

      gunzip <package_name>

      For instance:

      gunzip TCMsudo-1.8.2-sparc.pkg.gz
    - Then, install the package by issuing the following command at the prompt:

      pkgadd -d <package name>

      For instance:

      pkgadd –d TCMsudo-1.8.2-sparc.pkg
    - Once installation is complete, you will find that the package is installed in the **/usr/local/** folder on the Solaris host.
- All the tests run by the eG agent should be configured with the full path to the install directory of the **sudo** package;

1.4.1.3 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 1.5 of this document.
- Enable the eG 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)**.

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

- Ensure that the eG agent has IP connectivity to at least one of the network interfaces of the VMs.
- Typically, the Windows File and Print Sharing port is 139. Enable the eG agent to communicate with this port.
- 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 1.6.1 of this document for a step-by-step procedure to achieve this.
- To enable the eG 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 agent to communicate with the guest operating system. Refer to Section 1.6.2 of this document for a step-by-step procedure to achieve this.
- Set the **INSIDE VIEW USING** flag for all the "inside view" tests to **Remote connection to VM (Windows)**.
- For monitoring a Linux/Solaris VM, the SSH port (TCP port 22) must be enabled for communication between the eG agent and the VM being monitored.

1.4.2 Pre-requisites for Monitoring the Oracle VirtualBox in an Agentless Manner

1.4.2.1 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 (eg., *Windows Japanese*), then the remote agent monitoring such VMs should also run on a multi-byte operating system.

1.4.2.2 Pre-requisites for Auto-Discovering VMs and Obtaining their "Outside View"

- Ensure that the remote agent has IP connectivity to the target VirtualBox.
- Ensure that the remote agent can connect to the target VirtualBox via SSH.

- Configure all the tests that the remote agent executes with the name and password of a user who is privileged to access the VirtualBox via SSH.
- To enable the remote agent to run VDA commands on the VirtualBox, a **sudo** package has to be installed on the VirtualBox host; to know how to install the **sudo** package, refer to Section 1.4.1.2 above.
- After the **sudo** package is installed, perform the following steps on the VirtualBox host:
  - Login to the host as a *root* user;
  - At the command prompt of the host, issue the following command to create a new user:

    useradd –d /export/home/<username> –m <username>

    For instance:

    useradd –d /export/home/eguser –m eguser
  - Next, issue the following command to set a password for the above user:

    passwd <username>
  - When prompted to provide the password, specify the same.
  - Then, proceed to edit the **sudo** script by issuing the following command:

    usr/local/sbin/visudo
  - Add the following entry to the script:

    <username> ALL=NOPASSWD:/usr/bin/VBoxManage
- All the tests run by the eG agent should be configured with the full path to the install directory of the **sudo** package;

1.4.2.3 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 1.5 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)**.

1.4.2.4 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 1.6.1 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 1.6.2 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/Solaris VM, the SSH port (TCP port 22) must be enabled for communication between the remote agent and the VM being monitored.
- 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)**.

1.5 Configuring the eG 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**.

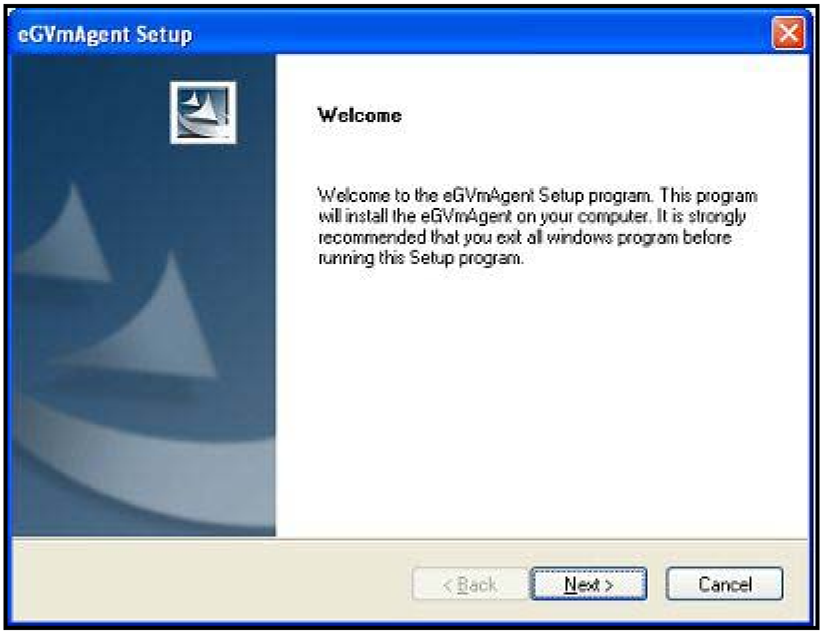
Users have multiple options to choose from when it comes to installing the eG VM Agent. These options have been discussed below:

- 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:

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**.

Figure 1.6 then appears. Click on the **Next** button in Figure 1.6 to continue.



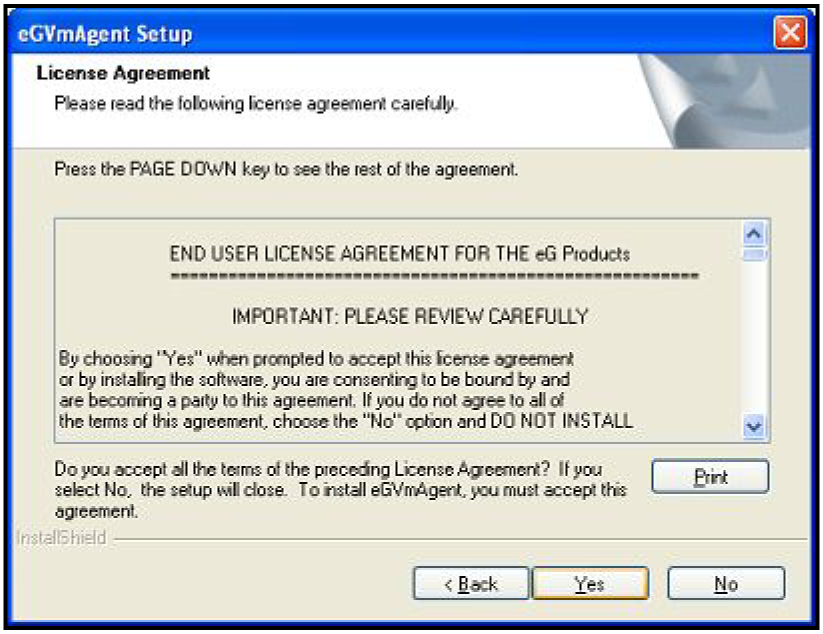
eGVmAgent Setup

Welcome

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

Figure 1.6: Welcome screen of the eG VM Agent installation wizard

1. When Figure 1.7 appears, click on **Yes** to accept the displayed license agreement.



eGVmAgent Setup

License Agreement

Please read the following license agreement carefully.

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

END USER LICENSE AGREEMENT FOR THE eG Products
==

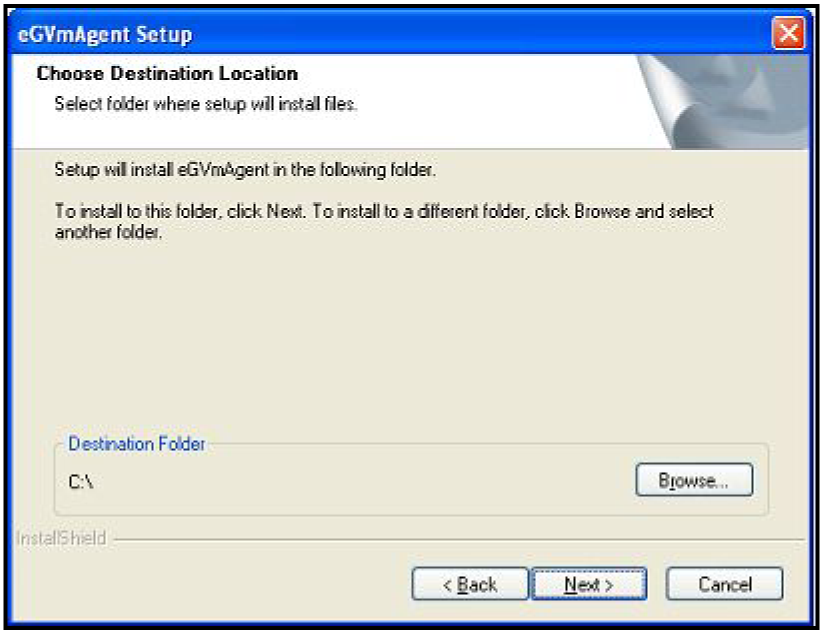
IMPORTANT: PLEASE REVIEW CAREFULLY

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

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

Figure 1.7: Accepting the license agreement

2. Use the **Browse** button in Figure 1.8 to indicate the location in which the agent should be installed, and click the **Next** button to proceed.



eGVmAgent Setup

Choose Destination Location

Select folder where setup will install files.

Setup will install eGVmAgent in the following folder.

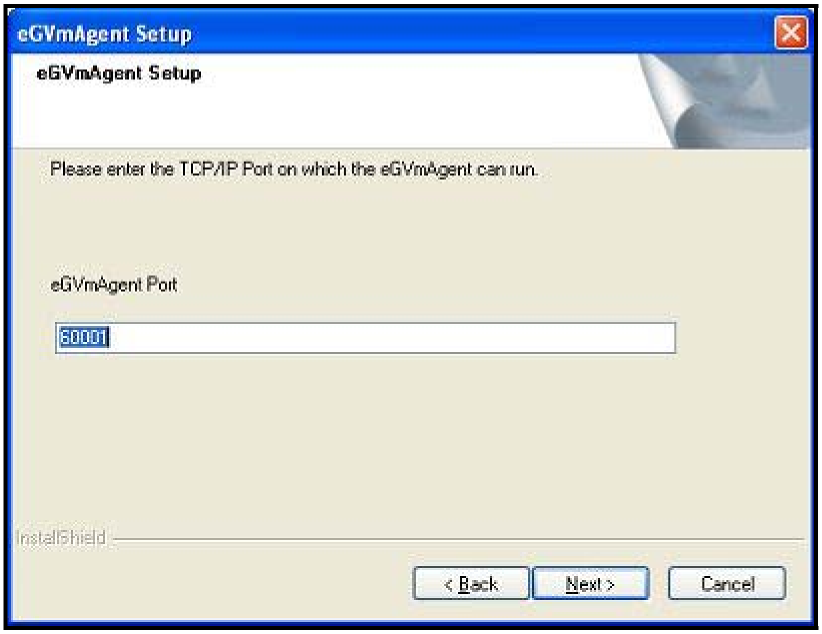
To install to this folder, click Next. To install to a different folder, click Browse and select another folder.

Destination Folder

C:\

Figure 1.8: Specifying the install directory of the eG VM Agent

3. 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 1.9 to proceed.



eGVmAgent Setup

eGVmAgent Setup

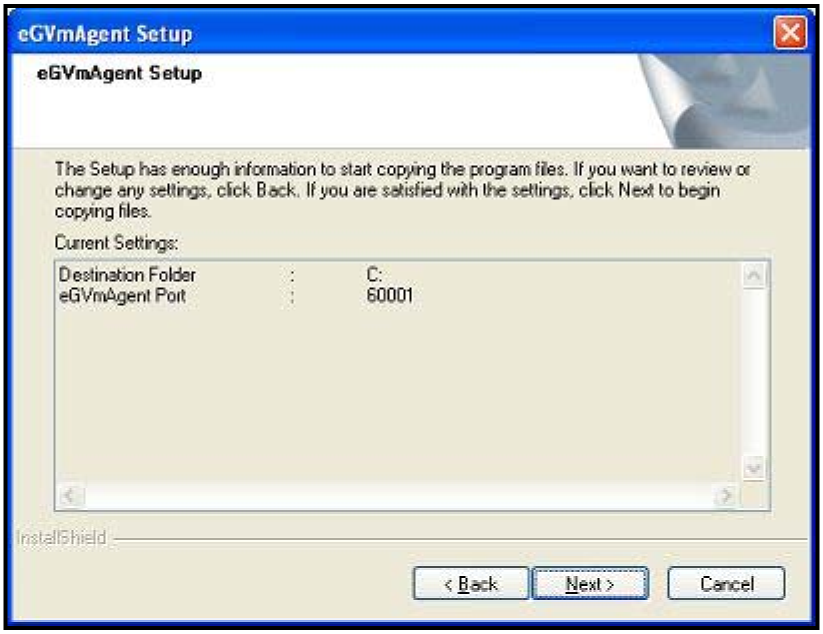
Please enter the TCP/IP Port on which the eGVmAgent can run.

eGVmAgent Port

60001

Figure 1.9: Specifying the VM agent port

4. A summary of your specifications then follows (see Figure 1.10). Click **Next** to proceed.



eGVmAgent Setup

eGVmAgent Setup

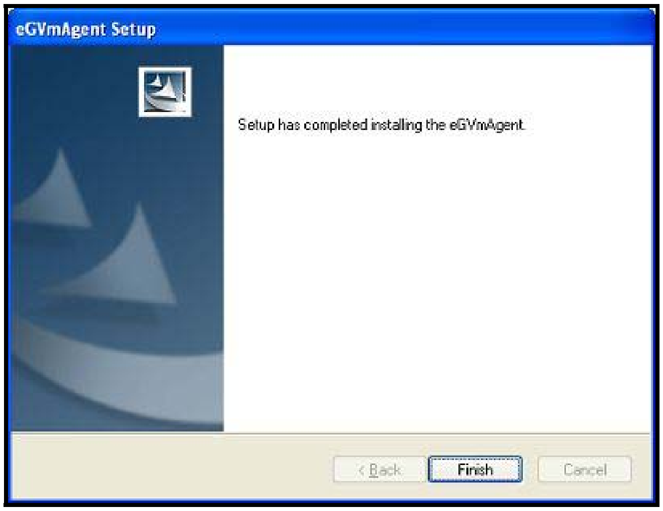
The Setup has enough information to start copying the program files. If you want to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files.

Current Settings:

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

Figure 1.10: A summary of your specifications

5. Finally, click the **Finish** button in Figure 1.11 to complete the installation.



eGVmAgent Setup

Setup has completed installing the eGVmAgent.

Figure 1.11: Finishing the installation

1.5.1 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 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.

1.5.2 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.

1.5.3 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 Enterprise Suite 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.

1.6 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 VirtualBox 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 (Windows)**.

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

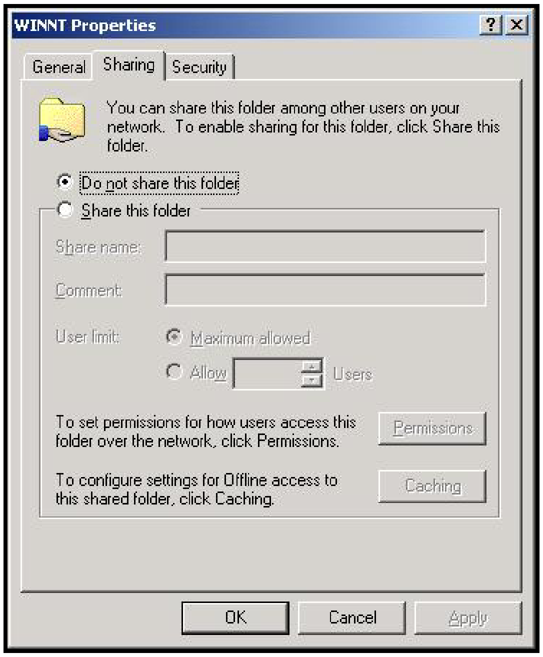
The sections to come discuss the procedure to be followed for fulfilling the 2 requirements above.

1.6.1 Enabling ADMIN$ Share Access on Windows Virtual Guests

1.6.1.1 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:

1. Open the Windows Explorer on the virtual machine, browse for the corresponding **Windows** directory in the C drive, right-click on it, and select the **Sharing** option from the shortcut menu.
2. If the **ADMIN$** share does not pre-exist on the Windows guest, then Figure 1.12 appears indicating the same.



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

● Do not share this folder

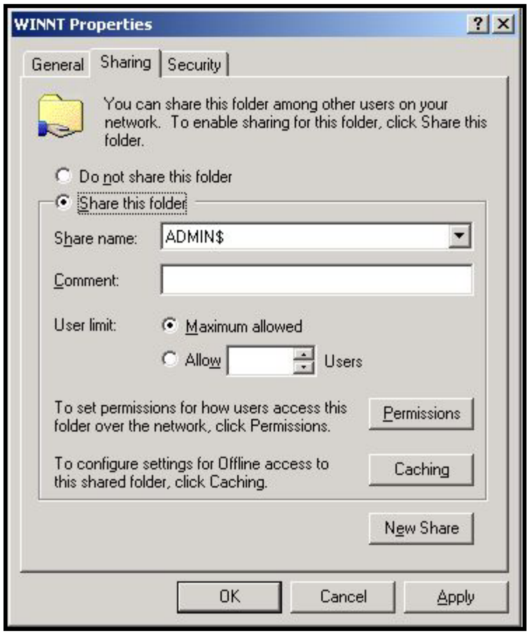
○ Share this folder

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

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

Figure 1.12: The ADMIN$ share does not exist

On the other hand, if the **ADMIN$** share pre-exists, Figure 1.13 appears. In such a case, first, remove the **ADMIN$** share by selecting the **Do not share this folder** option from Figure 1.13 and clicking the **Apply** and **OK** buttons. After this, you will have to repeat step 1 of this procedure to open Figure 1.12. Then, proceed as indicated by step 3 onwards.



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

○ Do not share this folder

● Share this folder

Share name: ADMIN$

Comment:

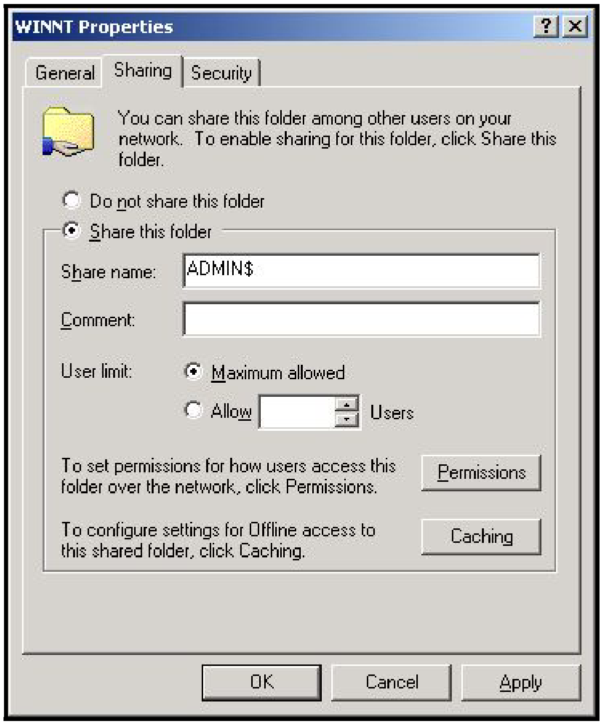
User limit: ● Maximum allowed ○ Allow Users

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

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

Figure 1.13: Admin$ share pre-exists

3. To create (or re-create) the **ADMIN$** share, select the **Share this folder** option from Figure 1.14, and provide **ADMIN$** share against the **Share name** text box (see Figure 1.14).



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

○ Do not share this folder

● Share this folder

Share name: ADMIN$

Comment:

User limit: ● Maximum allowed ○ Allow Users

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

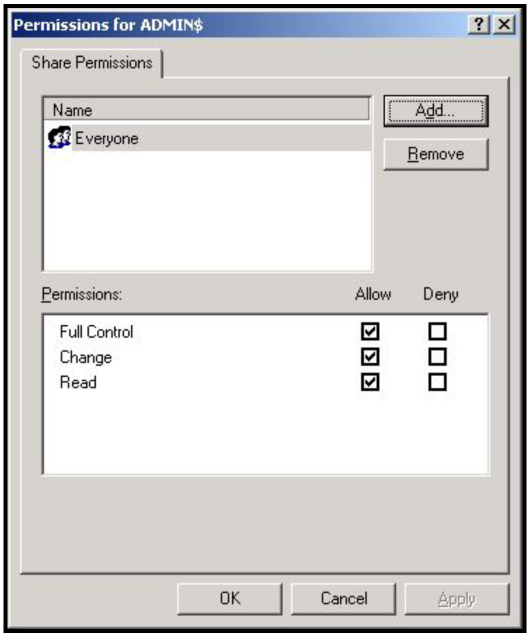
To configure settings for Offline access to this shared folder, click Caching.

Figure 1.14: Creating the ADMIN$ share

4. 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 1.14.

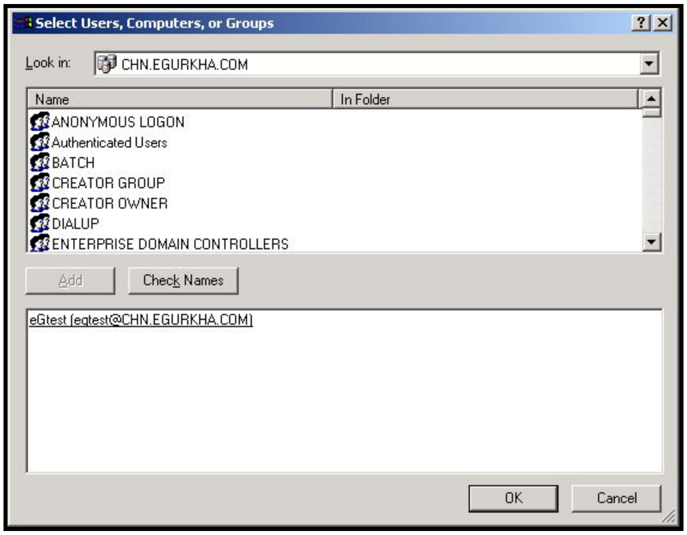
5. By default, the **ADMIN$** share can be accessed by **Everyone** (see Figure 1.15). To grant access rights to a specific administrative (local/domain) user, select the **Add** button in Figure 1.15. When Figure 1.16 appears, select the domain to search from the **Look in** list. The valid user accounts configured on the chosen domain then appear in the box below. From this box, choose the administrator's account and click on the **Add** button to add the chosen user account to the box below the **Add** button.



Name
Everyone

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

Figure 1.15: Clicking the Add button



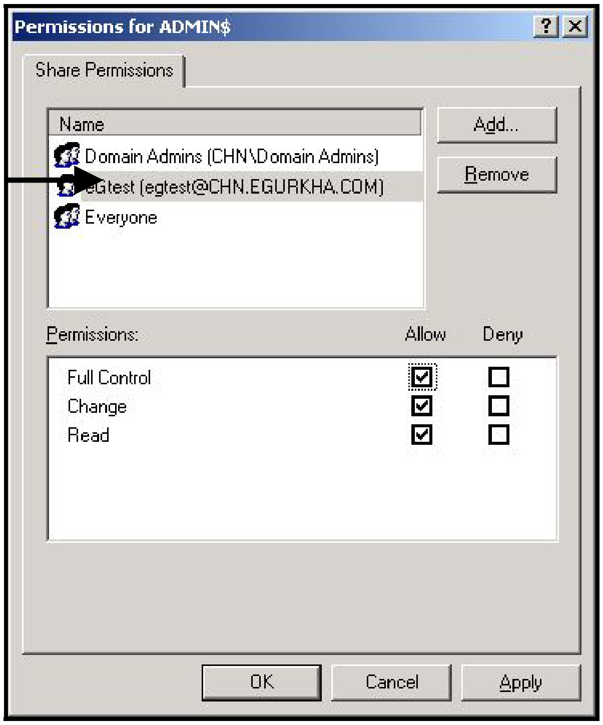
Look in: CHN.EGURKHA.COM

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

eGtest (egtest@CHN.EGURKHA.COM)

Figure 1.16: Selecting the administrative user to whom access rights are to be granted

6. Finally, click the **OK** button. You will then switch to Figure 1.17, where the newly added administrator account will appear.



Permissions for ADMIN$

Share Permissions

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

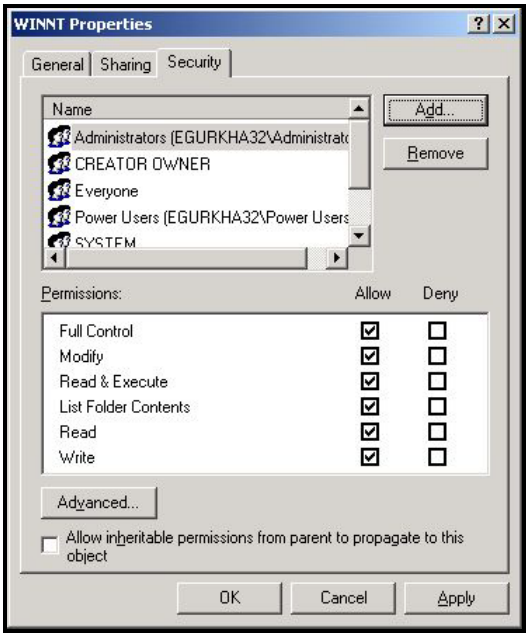
Permissions:	Allow	Deny
Full Control	☑	□
Change	☑	□
Read	☑	□

Figure 1.17: The administrator account granted access permissions

7. Select the newly added administrator account from Figure 1.17, and then, using the **Permissions** section, grant the administrator **Full Control**, **Change**, and **Read** permissions.

8. Finally, click the **Apply** and **OK** buttons in Figure 1.17 to register the changes.

9. Once you return to Figure 1.18, click on the **Security** tab to define the security settings for the **ADMIN$** share (see Figure 1.18).



WINNT Properties

General | Sharing | Security

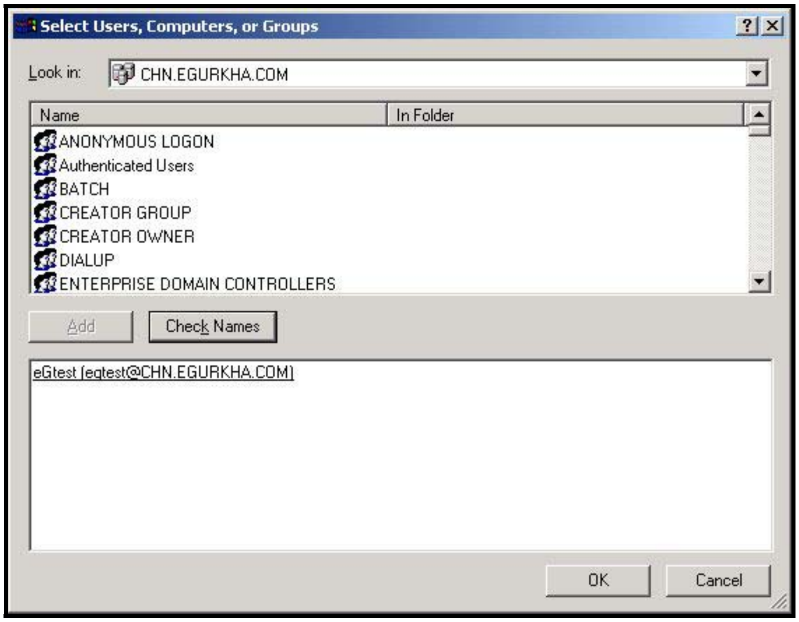
Name
Administrators (EGURKHA32\Administrato
CREATOR OWNER
Everyone
Power Users (EGURKHA32\Power Users
SYSTEM

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

□ Allow inheritable permissions from parent to propagate to this object

Figure 1.18: Defining the Security settings for the ADMIN$ share

10. 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 1.18, pick a domain from the **Look in** list of Figure 1.19, select the said administrator account from the domain users list below, and click the **Add** button (in Figure 1.19) to add the chosen account. Then, click the **OK** button in Figure 1.19.



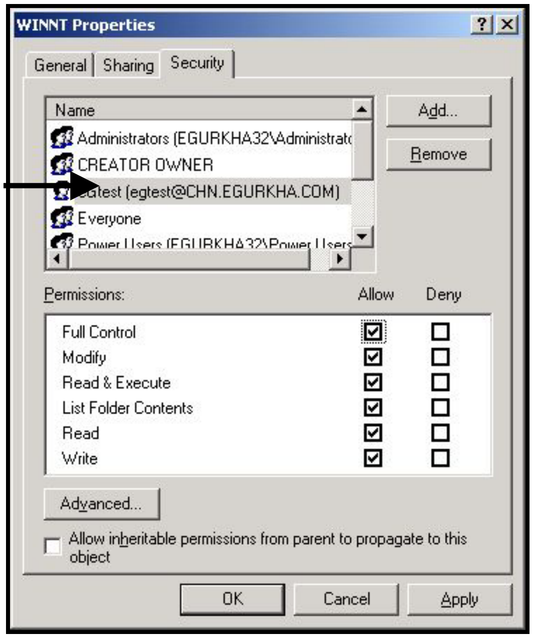
Look in: CHN.EGURKHA.COM

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

eGtest (egtest@CHN.EGURKHA.COM)

Figure 1.19: Adding the administrator account

11. This will bring you back to Figure 1.18, but this time, the newly added domain administrator account will be listed therein as indicated by Figure 1.20.



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

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

□ Allow inheritable permissions from parent to propagate to this object

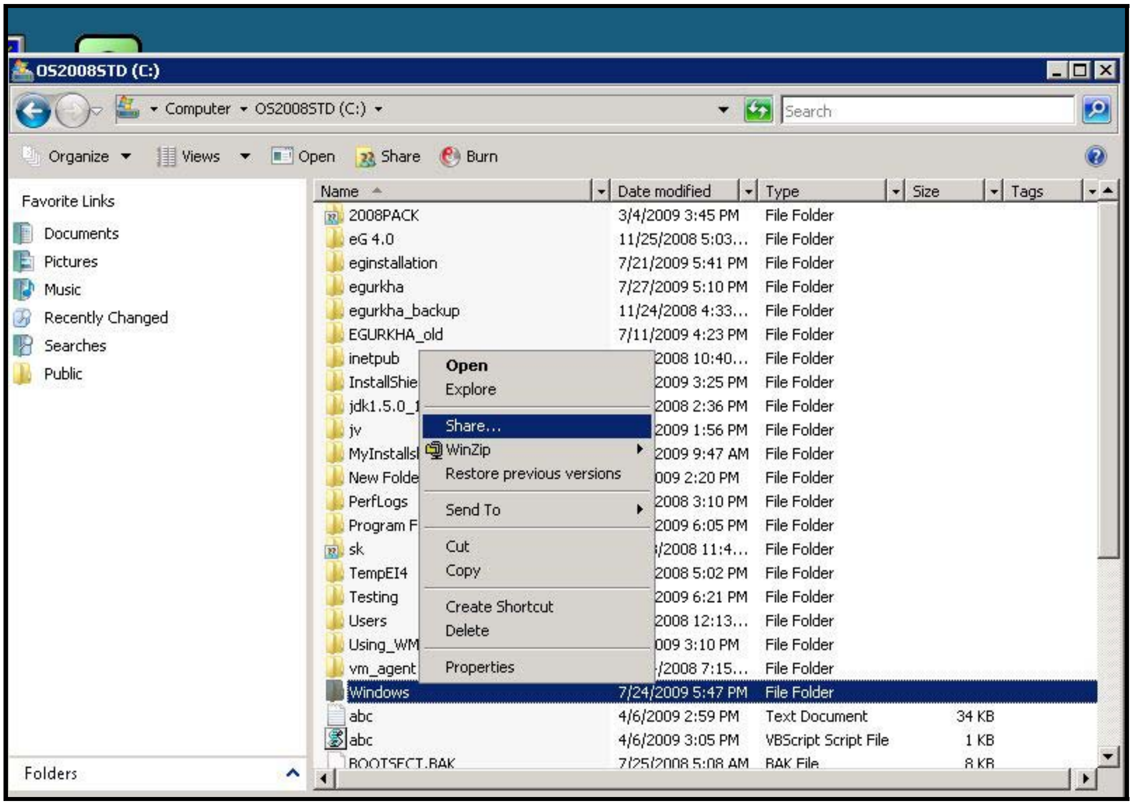
Figure 1.20: The Administrator account in the Security list

12. Finally, click the **Apply** and **OK** buttons in Figure 1.20.

1.6.1.2 Enabling ADMIN$ Share Access on Windows 2008 VMs

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

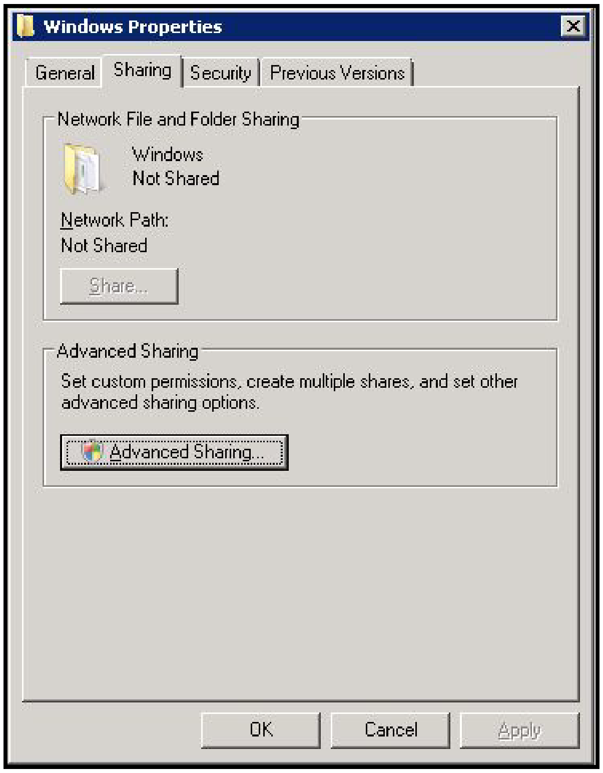
1. Open the Windows Explorer on the virtual machine, browse for the corresponding **Windows** directory in the C drive, right-click on it, and select the **Share** option from the shortcut menu.



Name	Date modified	Type	Size	Tags
2008PACK	3/4/2009 3:45 PM	File Folder		
eG 4.0	11/25/2008 5:03...	File Folder		
eginstallation	7/21/2009 5:41 PM	File Folder		
egurkha	7/27/2009 5:10 PM	File Folder		
egurkha_backup	11/24/2008 4:33...	File Folder		
EGURKHA_old	7/11/2009 4:23 PM	File Folder		
inetpub	2008 10:40...	File Folder		
InstallShie	2009 3:25 PM	File Folder		
jdk1.5.0_1	2008 2:36 PM	File Folder		
jv	2009 1:56 PM	File Folder		
MyInstallsl	2009 9:47 AM	File Folder		
New Folde	009 2:20 PM	File Folder		
PerfLogs	2008 3:10 PM	File Folder		
Program F	2009 6:05 PM	File Folder		
sk	/2008 11:4...	File Folder		
TempEI4	2008 5:02 PM	File Folder		
Testing	2009 6:21 PM	File Folder		
Users	2008 12:13...	File Folder		
Using_WM	009 3:10 PM	File Folder		
vm_agent	/2008 7:15...	File Folder		
Windows	7/24/2009 5:47 PM	File Folder		
abc	4/6/2009 2:59 PM	Text Document	34 KB	
abc	4/6/2009 3:05 PM	VBScript Script File	1 KB	
BOOTSECT.BAK	7/25/2008 5:08 AM	BAK File	8 KB	

Figure 1.21: Selecting the Share option from the shortcut menu

Figure 1.22 will then appear. Click on **Advanced Sharing** in Figure 1.22.



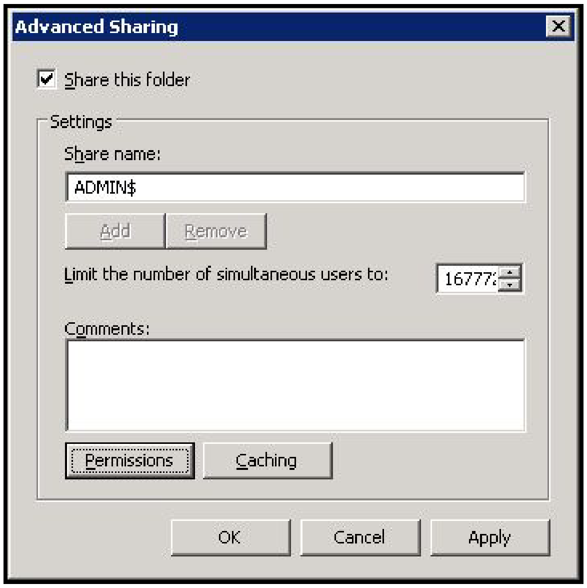
Windows
Not Shared

Network Path:
Not Shared

Set custom permissions, create multiple shares, and set other advanced sharing options.

Figure 1.22: Cicking on Advanced Sharing

Select the **Share this folder** check box in Figure 1.23 that appears, enter **ADMIN$** against **Share name**, and click on the **Permissions** button in Figure 1.23, to allow only a local/domain administrator to access the folder.



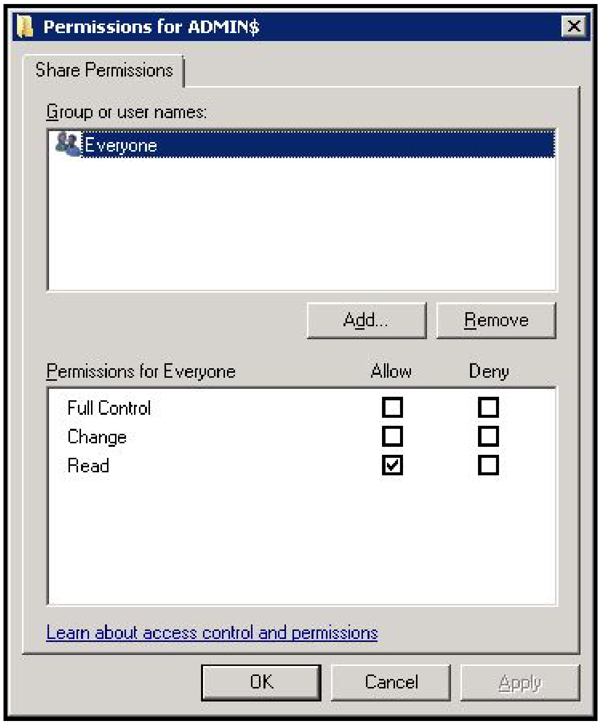
☑ Share this folder

Share name: ADMIN$

Limit the number of simultaneous users to: 16777:

Figure 1.23: Enabling the ADMIN$ share

- When Figure 1.24 appears, click on the **Add** button therein.



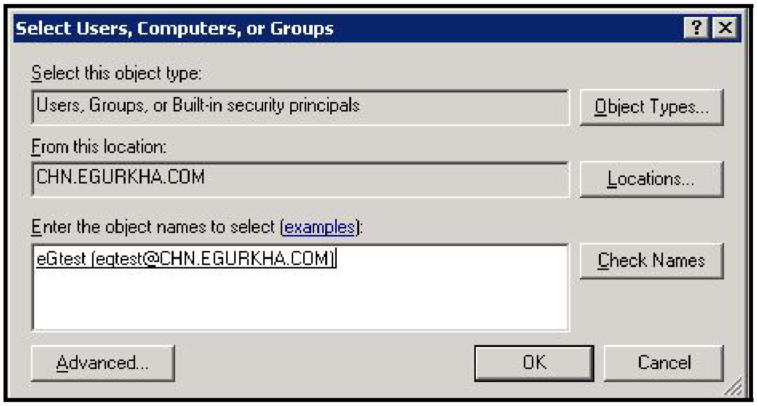
Group or user names:

Everyone

Permissions for Everyone	Allow	Deny
Full Control	□	□
Change	□	□
Read	☑	□

Figure 1.24: Clicking on the Add button

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 1.25. 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 1.25. Then, enter the name of the local/domain administrator in the **Enter the object names to select** text area, and click the **OK** button.



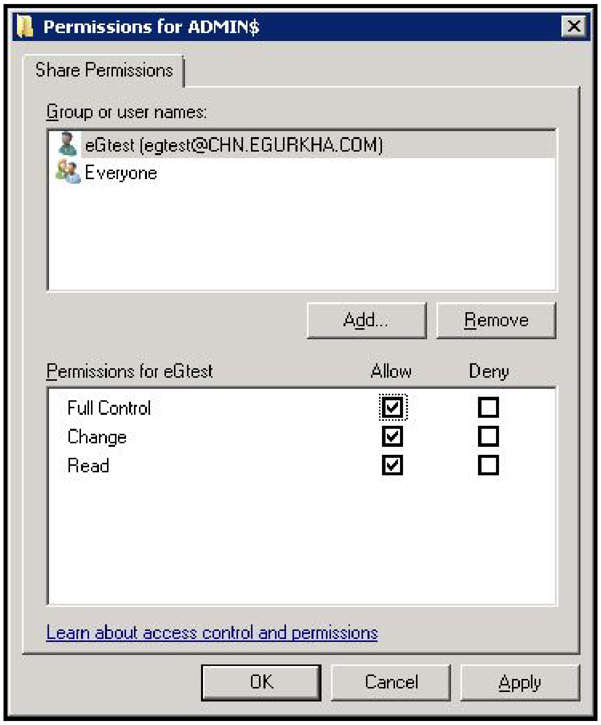
Select this object type:
Users, Groups, or Built-in security principals

From this location:
CHN.EGURKHA.COM

Enter the object names to select (examples):
eGtest (egtest@CHN.EGURKHA.COM)

Figure 1.25: Allowing a domain administrator to access the folder

The newly added user will be listed in the **Group or user names** section, as depicted by Figure 1.26. Select this user, and then, check all the three check boxes under **Allow** in the **Permissions for <user>** section in Figure 1.26. Then, click the **Apply** and **OK** buttons therein.



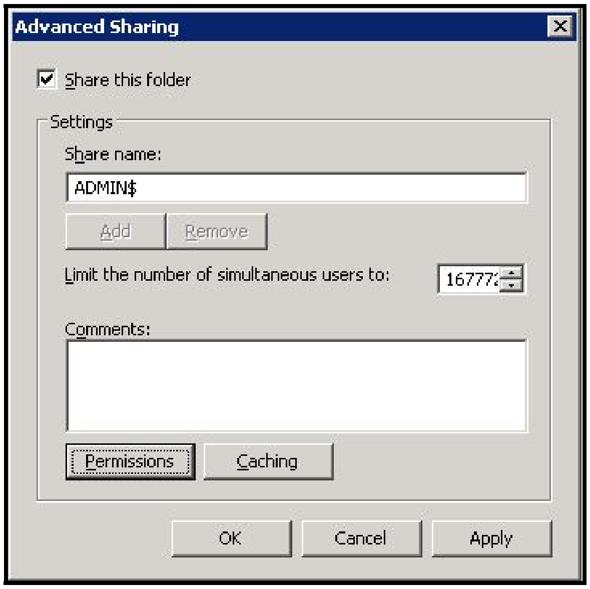
Group or user names:

- eGtest (egtest@CHN.EGURKHA.COM)
- Everyone

Permissions for eGtest	Allow	Deny
Full Control	☑	□
Change	☑	□
Read	☑	□

Figure 1.26: Allowing full access to the local/domain administrator

When Figure 1.27 appears, click on the **Apply** and **OK** buttons therein to register the changes.



☑ Share this folder

Share name: ADMIN$

Limit the number of simultaneous users to: 16777

Figure 1.27: Applying the changes

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

2. 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:

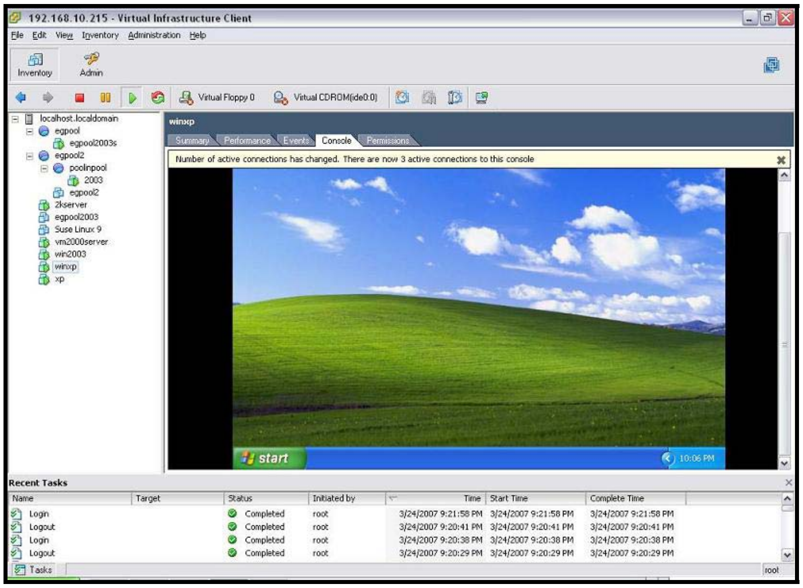
As with any change to the registry, ensure that the above-mentioned change is also performed with utmost care, so as to avoid problems in the functioning of the operating system.

1.6.2 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 1.28).



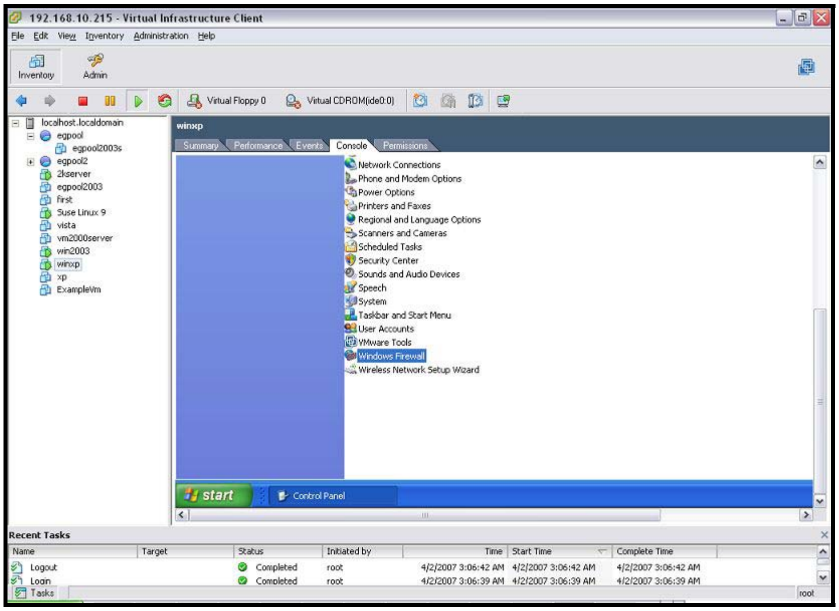
Number of active connections has changed. There are now 3 active connections to this console

Recent Tasks

Name	Target	Status	Initiated by	Time	Start Time	Complete Time
Login		Completed	root	3/24/2007 9:21:58 PM	3/24/2007 9:21:58 PM	3/24/2007 9:21:58 PM
Logout		Completed	root	3/24/2007 9:20:41 PM	3/24/2007 9:20:41 PM	3/24/2007 9:20:41 PM
Login		Completed	root	3/24/2007 9:20:38 PM	3/24/2007 9:20:38 PM	3/24/2007 9:20:38 PM
Logout		Completed	root	3/24/2007 9:20:29 PM	3/24/2007 9:20:29 PM	3/24/2007 9:20:29 PM

Figure 1.28: Selecting the guest OS

2. Follow the menu sequence: Start -> All Programs -> Control Panel (see Figure 1.29), and then double-click on the **Windows Firewall** option within.

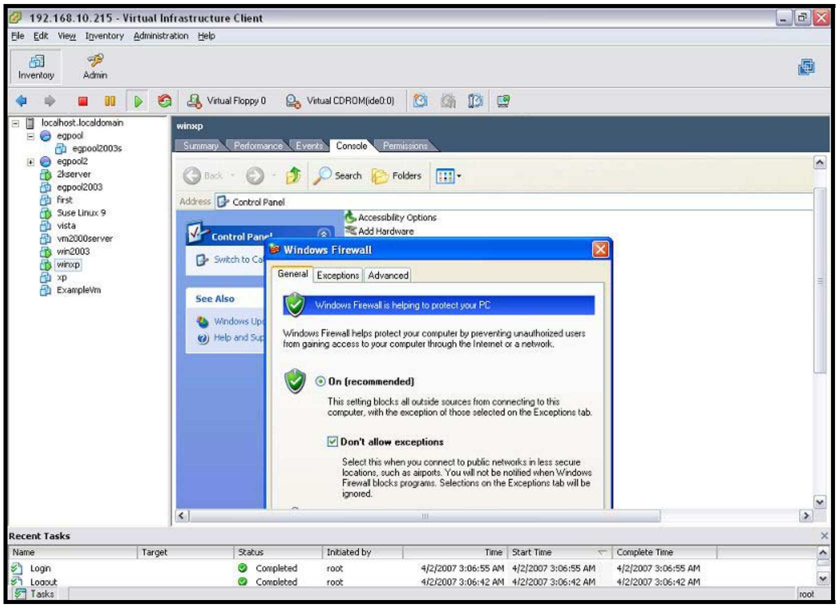


Recent Tasks

Name	Target	Status	Initiated by	Time	Start Time	Complete Time
Logout		Completed	root	4/2/2007 3:06:42 AM	4/2/2007 3:06:42 AM	4/2/2007 3:06:42 AM
Login		Completed	root	4/2/2007 3:06:39 AM	4/2/2007 3:06:39 AM	4/2/2007 3:06:39 AM

Figure 1.29: Opening the Windows Firewall

3. Figure 1.30 then appears, with the **General** tab selected by default.



Windows Firewall is helping to protect your PC

Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

● On (recommended)

This setting blocks all outside sources from connecting to this computer, with the exception of those selected on the Exceptions tab.

☑ Don't allow exceptions

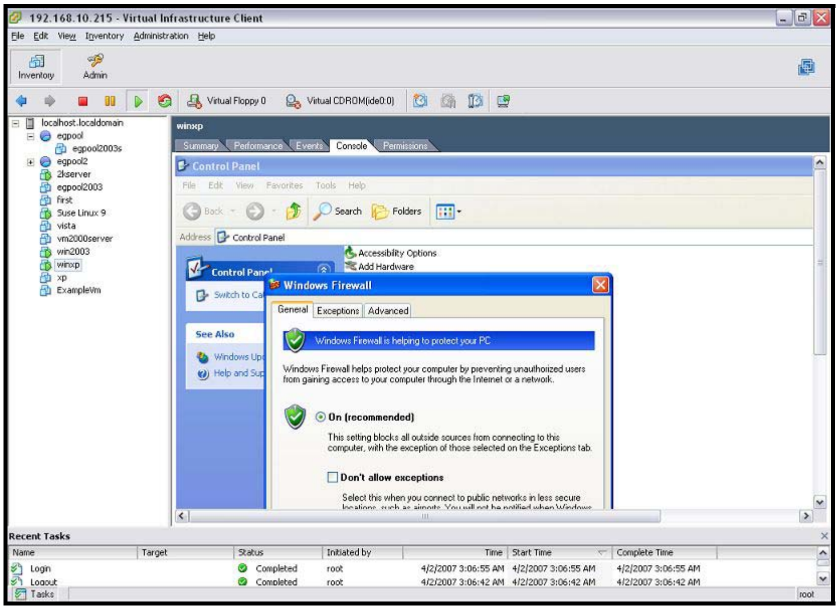
Select this when you connect to public networks in less secure locations, such as airports. You will not be notified when Windows Firewall blocks programs. Selections on the Exceptions tab will be ignored.

Recent Tasks

Name	Target	Status	Initiated by	Time	Start Time	Complete Time
Login		Completed	root	4/2/2007 3:06:55 AM	4/2/2007 3:06:55 AM	4/2/2007 3:06:55 AM
Logout		Completed	root	4/2/2007 3:06:42 AM	4/2/2007 3:06:42 AM	4/2/2007 3:06:42 AM

Figure 1.30: The General tab of the Windows Firewall dialog box

4. Deselect the **Don't allow exceptions** check box as indicated by Figure 1.31.



Windows Firewall is helping to protect your PC

Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

● On (recommended)

This setting blocks all outside sources from connecting to this computer, with the exception of those selected on the Exceptions tab.

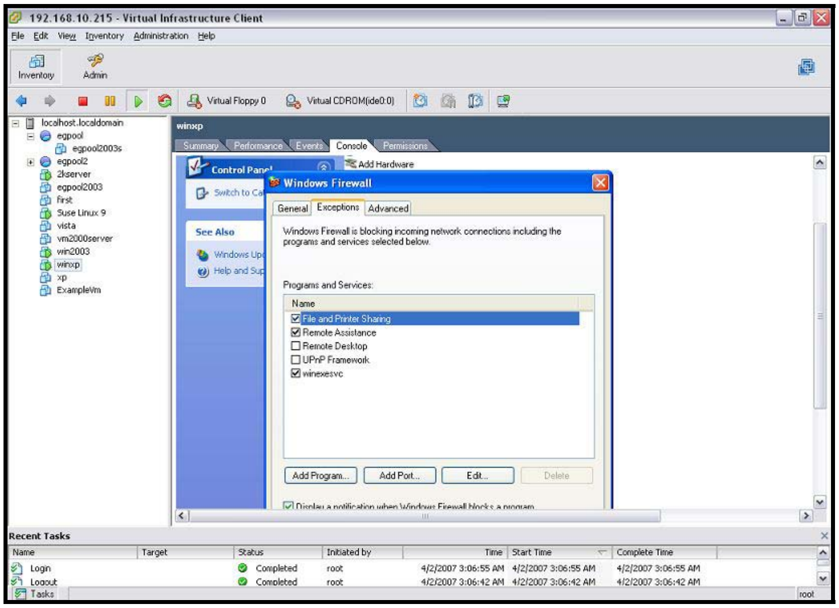
□ Don't allow exceptions

Select this when you connect to public networks in less secure locations, such as airports. You will not be notified when Windows

Name	Target	Status	Initiated by	Time	Start Time	Complete Time
Login		Completed	root	4/2/2007 3:06:55 AM	4/2/2007 3:06:55 AM	4/2/2007 3:06:55 AM
Logout		Completed	root	4/2/2007 3:06:42 AM	4/2/2007 3:06:42 AM	4/2/2007 3:06:42 AM

Figure 1.31: 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 1.32).



Windows Firewall is blocking incoming network connections including the programs and services selected below.

Programs and Services:

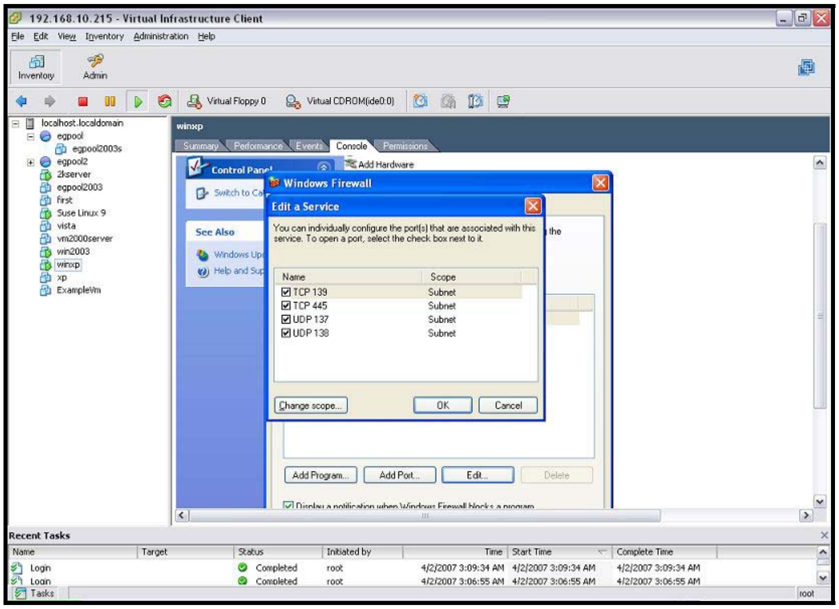
Name
☑ File and Printer Sharing
☑ Remote Assistance
□ Remote Desktop
□ UPnP Framework
☑ winexesvc

☑ Display a notification when Windows Firewall blocks a program

Name	Target	Status	Initiated by	Time	Start Time	Complete Time
Login		Completed	root	4/2/2007 3:06:55 AM	4/2/2007 3:06:55 AM	4/2/2007 3:06:55 AM
Logout		Completed	root	4/2/2007 3:06:42 AM	4/2/2007 3:06:42 AM	4/2/2007 3:06:42 AM

Figure 1.32: Enabling 'File and Printer Sharing'

6. Then, click the **Edit** button in Figure 1.33 to open the ports required for the agent-guest communication. Ensure that at least one of the listed TCP ports are enabled.



Edit a Service

You can individually configure the port(s) that are associated with this service. To open a port, select the check box next to it.

Name	Scope
☑ TCP 139	Subnet
☑ TCP 445	Subnet
☑ UDP 137	Subnet
☑ UDP 138	Subnet

Change scope... OK Cancel

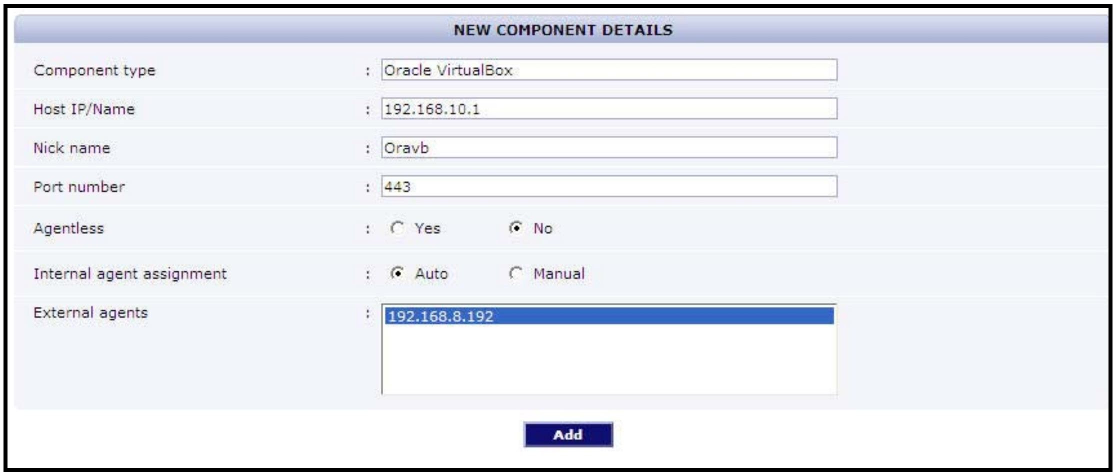
Figure 1.33: Opening ports

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

1.7 Administering the eG Manager to monitor the Oracle VirtualBox

To achieve this, do the following:

1. Log into the eG administrative interface.
2. eG Enterprise cannot automatically discover the *Oracle VirtualBox*. You need to manually add the component using the add/modify components page (see Figure Figure 1.34) that appears when the Infrastructure -> Components -> Add/Modify menu sequence is followed. Remember that components manually added are managed automatically.

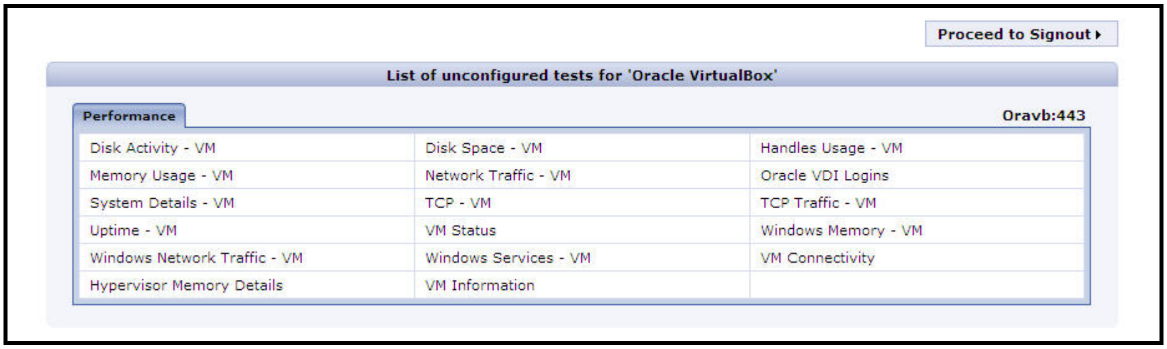

NEW COMPONENT DETAILS

Component type	:	Oracle VirtualBox
Host IP/Name	:	192.168.10.1
Nick name	:	Oravb
Port number	:	443
Agentless	:	○ Yes ● No
Internal agent assignment	:	● Auto ○ Manual
External agents	:	192.168.8.192

Add

Figure 1.34: Adding the Oracle VirtualBox component

3. Next, when you try to signout of the administrative interface, a list of Unconfigured tests will appear as shown in Figure 1.35.



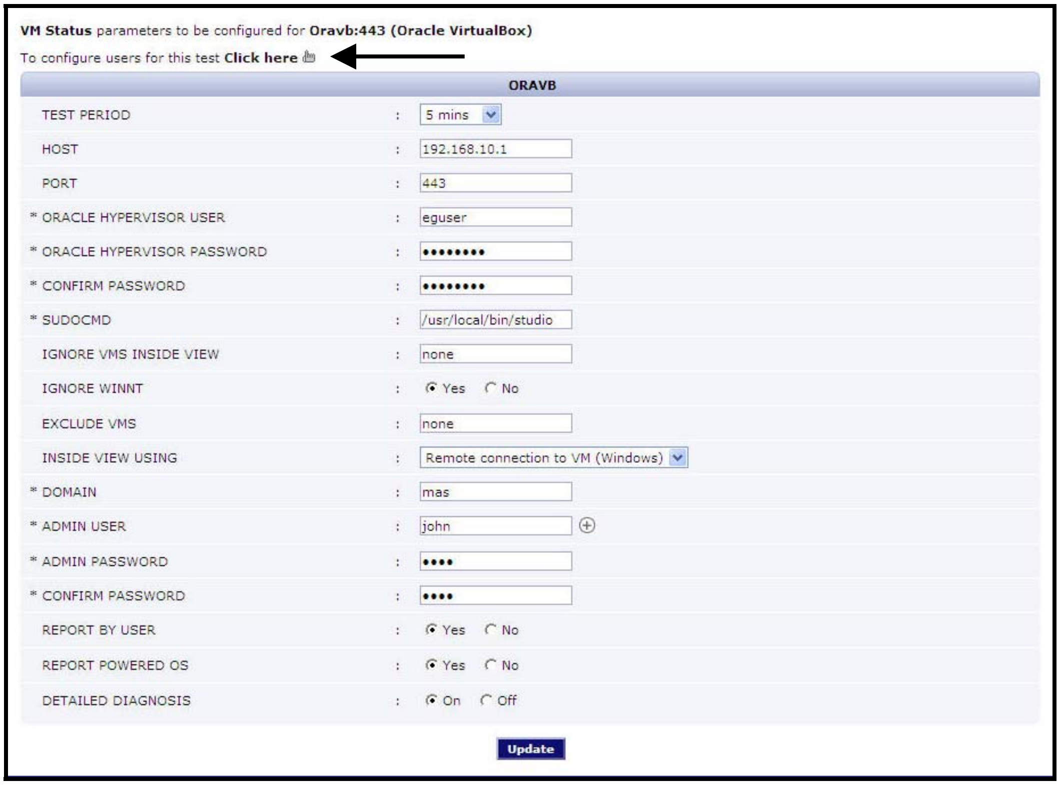
List of unconfigured tests for 'Oracle VirtualBox'

Performance | Oravb:443

Disk Activity - VM	Disk Space - VM	Handles Usage - VM
Memory Usage - VM	Network Traffic - VM	Oracle VDI Logins
System Details - VM	TCP - VM	TCP Traffic - VM
Uptime - VM	VM Status	Windows Memory - VM
Windows Network Traffic - VM	Windows Services - VM	VM Connectivity
Hypervisor Memory Details	VM Information	

Figure 1.35: List of Unconfigured tests for the Oracle VirtualBox

4. Now try to configure the VM Status test as shown in Figure 1.36.



VM Status parameters to be configured for **Oravb:443 (Oracle VirtualBox)**

To configure users for this test **Click here**

ORAVB

TEST PERIOD	:	5 mins
HOST	:	192.168.10.1
PORT	:	443
* ORACLE HYPERVISOR USER	:	eguser
* ORACLE HYPERVISOR PASSWORD	:	••••••••
* CONFIRM PASSWORD	:	••••••••
* SUDOCMD	:	/usr/local/bin/studio
IGNORE VMS INSIDE VIEW	:	none
IGNORE WINNT	:	● Yes ○ No
EXCLUDE VMS	:	none
INSIDE VIEW USING	:	Remote connection to VM (Windows)
* DOMAIN	:	mas
* ADMIN USER	:	john
* ADMIN PASSWORD	:	••••
* CONFIRM PASSWORD	:	••••
REPORT BY USER	:	● Yes ○ No
REPORT POWERED OS	:	● Yes ○ No
DETAILED DIAGNOSIS	:	● On ○ Off

Figure 1.36: Configuring the VM Status test

5. Specify the following as shown in Figure 1.36.

   - **TEST PERIOD** - How often should the test be executed
   - **HOST** - The host for which the test is to be configured
   - **PORT –** Refers to the port used by the specified **HOST**.
   - **ORACLE HYPERVISOR USER** - Specify the name of the user who has the right to access the VirtualBox via SSH.

- **ORACLE HYPERVISOR PASSWORD** - Provide the password of the **ORACLE HYPERVISOR USER**.
- **CONFIRM PASSWORD** - Confirm the password by retyping it here.
- **SUDOCMD** - This test executes certain privileged VDA (Virtual Desktop Access) commands to pull out the desired metrics from the VirtualBox. To enable the test to run these commands, you first need to install a **sudo** package on the VirtualBox host. The procedure for installing this package is detailed in Section 1.4.1.2 of this document. Once the package is installed, you need to specify the full path to the install directory of the **sudo** package in the **SUDOCMD** text box.
- **IGNORE VMS INSIDE VIEW**- Administrators of some high security virtualized environments might not have permissions to internally monitor one/more VMs. The eG agent can be configured to ignore such 'inaccessible' VMs using the **IGNORE VMS INSISDE 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 VirtualBox 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.

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

  Typically, to establish this remote connection with Windows VMs in particular, eG Enterprise requires that the eG agent be configured with domain administrator privileges. In high-security environments, where the IT staff might have reservations about exposing the credentials of their domain administrators, this approach to extracting "inside view" metrics might not be preferred. In such environments therefore, eG Enterprise provides administrators the option to deploy a piece of software called the **eG VM Agent (Windows)** on every Windows VM; this VM agent allows the eG agent to collect "inside view" metrics from the Windows VMs **without domain administrator rights.** Refer to Section 1.5 of *Monitoring the Oracle VirtualBox* document 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*.

- **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:

  1. **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.

  2. **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 2.4 of *Monitoring the Oracle VirtualBox* document.

  3. **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 1.8 of this document.

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

- **REPORT BY USER** - While monitoring a VirtualBox, this flag is set to **Yes** by default, indicating that by default, the guest operating systems on the VirtualBox are identified using the login of the user who is accessing the guest operating systems. If this flag is set to **No**, then the guests will be identified using the host name specified of the guest operating system.

- **REPORT POWERED OS** - This flag is set to **Yes** by default. This flag is closely related to the **REPORT BY USER** flag. As already mentioned, the **REPORT BY USER** flag is set to **Yes** by default. In this case, the default **Yes** status of the **REPORT POWERED OS** flag implies that this test will report measures for even those VMs that do not have any users logged in, as long as the VM is powered-on. Such guests will be identified by their *virtual machine name* and not the *user name*. If the status of the **REPORT POWERED OS** flag is changed to **No**, then this test will not report measures for those powered-on VMs to which no users are logged in currently. If the **REPORT BY USER** flag is set to **No**, then the eG Enterprise system will disregard the status of the **REPORT POWERED OS** flag (be it **Yes** or **No**). In other words, this test will continue to report measures for every powered-on VM on the server.

- **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:

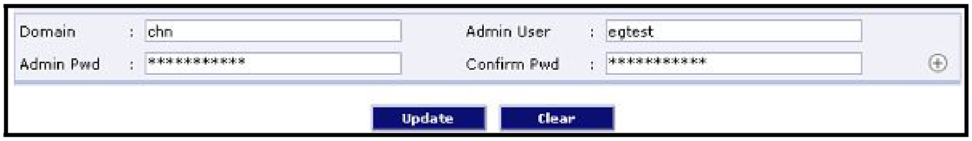
  - The eG manager license should allow the detailed diagnosis capability
  - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

6. Finally click on the **Update** button and signout of the administrative interface.

1.8 Configuring Users for VM Monitoring

In order to enable the eG agent to connect to VMs in multiple domains and pull out metrics from them, the eG administrative interface provides a special page using which the different **DOMAIN** names, and their corresponding **ADMIN USER** names and **ADMIN PASSWORDS** can be specified. To access this page, just click on the **Click here** hyperlink in any of the VM test configuration pages as shown in Figure 1.36.

Upon clicking, Figure 1.37 will appear, using which the VM user details can be configured.

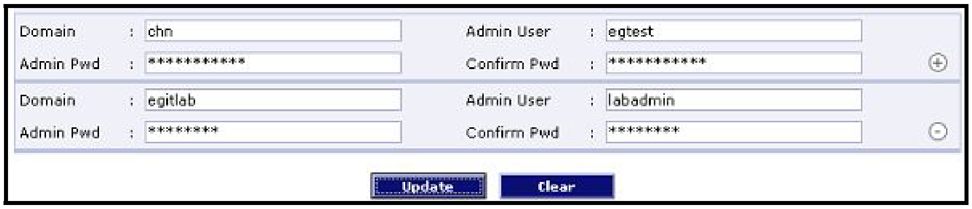


Domain	:	chn	Admin User	:	egtest
Admin Pwd	:	***********	Confirm Pwd	:	***********

Figure 1.37: The VM user configuration page

To add a user specification, do the following:

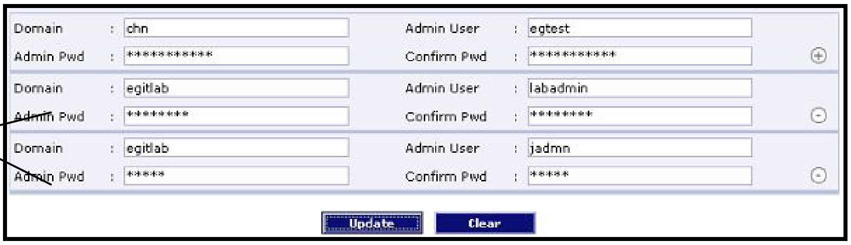
- First, provide the name of the **Domain** to which the VMs belong (see Figure 1.37). If one/more VMs do not belong to any domain, then, specify *none* here.
- The eG agent must be configured with user privileges that will allow the agent to communicate with the VMs in a particular domain and extract 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 1.10 of this document.
- The password of the specified **Admin User** should be mentioned in the **Admin Pwd** text box.
- Confirm the password by retyping it in the **Confirm Pwd** text box.
- To add more users, click on the ⊕ button in Figure 1.37. This will allow you to add one more user specification as depicted by Figure 1.38.



Domain	:	chn	Admin User	:	egtest
Admin Pwd	:	***********	Confirm Pwd	:	***********
Domain	:	egitlab	Admin User	:	labadmin
Admin Pwd	:	********	Confirm Pwd	:	********

Figure 1.38: Adding another user

- In some virtualized environments, the same **Domain** could be accessed using multiple **Admin User** names. For instance, to login to a **Domain** named *egitlab*, the eG agent can use the **Admin User** name *labadmin* or the **Admin User** name *jadmn*. You can configure the eG agent with the credentials of both these users as shown by Figure 1.39.



Domain	: chn	Admin User	: egtest
Admin Pwd	: ***********	Confirm Pwd	: ***********
Domain	: egitlab	Admin User	: labadmin
Admin Pwd	: ********	Confirm Pwd	: ********
Domain	: egitlab	Admin User	: jadmn
Admin Pwd	: *****	Confirm Pwd	: *****

The same 'Domain' mapped to different 'Admin Users'

Figure 1.39: 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 1.39, 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 1.39). If the first login attempt itself is successful, then the agent will ignore the second **Admin User** name.

- To clear all the user specifications, simply click the **Clear** button in Figure 1.39.
- To remove the details of a particular user alone, just click the ⊖ button in Figure 1.39.
- To save the specification, just click on the **Update** button in Figure 1.39. This will lead you back to the test configuration page as shown in Figure 1.36, where you will find the multiple domain names, user names, and passwords listed against the respective fields.

1.9 Monitoring the Oracle VirtualBox

To monitor the Microsoft *Oracle VirtualBox*, do the following:

1. Login as a monitor / supermonitor user.
2. Click on the components option in the menu bar, and select the Servers option from the components menu.
3. From the **component list** page, click on the *Oracle VirtualBox* component for which you wish to view measurements.

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

By default, the eG agent uses secure shell (SSH) to connect to Linux guests, and collect performance metrics from them. Password Authentication is the default method for SSH connections in eG Enterprise. If the eG agent fails to report measures for a Linux guest or is unable to connect to a guest, it could imply that the Linux guest does not support SSH or that password authentication is not supported by the SSH daemon running on the Linux guest. Under such circumstances, you can perform either of the following:

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

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

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

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

- To enable key-based authentication, the private key should remain in the **<EG_INSTALL_DIR>\agent\sshkeys** directory (on Windows; on Unix, this will be **/opt/egurkha/agent/sshkeys**), and the public key should be copied to each of the Linux guests to be monitored. To achieve this, first login to the Linux guest to be monitored as the eG user.
- Create a directory named **.ssh** in the **<USER_HOME_DIR>** on the guest operating system, using the command: **mkdir ~/.ssh**.
- Next, copy the **authorized_keys** file from the **<EG_INSTALL_DIR>\agent\sshkeys** directory (on Windows; on Unix, this will be **/opt/egurkha/agent/sshkeys**) on the eG remote agent host to the **<USER_HOME_DIR>/.ssh** directory on the Linux guest.
- Make sure that the permission of the **.ssh** directory and the **authorized_keys** file is **700**.
- Finally, on the eG manager host, edit the **<EG_INSTALL_DIR>\manager\config\eg_tests.ini** file. Against the **EgJavaSSHKeyFile** parameter, enter: **agent/sshkeys/id_rsa.pub**, and save the file.

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

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

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

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

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

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

```
Your identification has been saved in /hom
```

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

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

  chmod go-w ~/
  chmod 700 ~/.ssh
  chmod go-rwx ~/.ssh/*
- Finally, on the eG manager host, edit the **<EG_INSTALL_DIR>\manager\config\eg_tests.ini** file. Against the **EgJavaSSHKeyFile** parameter, enter: **agent/sshkeys/id_rsa.pub**, and save the file.

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

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

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

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

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

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

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

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

Chapter

2

Configuring and Monitoring the Oracle VDI Broker

Oracle VDI (Virtual Desktop Infrastructure) provides desktop virtualization to replace personal computers with virtual machines (VMs) on a server. Users can access these VMs though any RDP client, or through the web via Sun Secure Global Desktop (SGD).

This chapter deals with the steps involved in the configuring and monitoring the *Microsoft Sharepoint 2007* servers.

2.1 Pre-requisites for Monitoring the Oracle VDI Broker in an Agent-based Manner

In case of an agent-based approach, the following pre-requisites need to be fulfilled to enable the eG agent to run the VDA commands:

A **sudo** package has to be installed on the broker host; to install this package, do the following:

- Login to the Solaris system hosting the broker as a *root* user.
- To download the **sudo** package, connect to the URL: http://sysinfo.bascomp.org/solaris/installing-sudo-on-solaris/
- If the Solaris processor is Intel based, download the file **TCMsudo-1.8.2-i386.pkg.gz** from the web site mentioned above. On the other hand, if the Solaris host uses a SPARC processor instead, download the file **TCMsudo-1.8.2-sparc.pkg.gz** from the web site.
- Download the chosen file to any location on the broker host (say, **/tmp**).
- From the Solaris prompt, switch to the directory hosting the downloaded package and unzip the compressed package using the following command:

  gunzip <package_name>

  For instance:

  gunzip TCMsudo-1.8.2-sparc.pkg.gz
- Then, install the package by issuing the following command at the prompt:

  pkgadd -d <package name>

  For instance:

pkgadd –d TCMsudo-1.8.2-sparc.pkg

- Once installation is complete, you will find that the package is installed in the **/usr/local/** folder on the Solaris host.

All the tests run by the eG agent should be configured with the full path to the install directory of the **sudo** package;

2.2 Pre-requisites for Monitoring the Oracle VDI Broker in an Agentless Manner

In case of an agentless approach, the following pre-requisites need to be fulfilled to enable the eG agent to run the VDA commands:

A **sudo** package has to be installed on the broker host; to know how to install the **sudo** package, refer to Section 1.1 above.

After the **sudo** package is installed, perform the following steps on the broker host:

- Login to the broker host as a *root* user;
- At the command prompt of the host, issue the following command to create a new user:

  useradd –d /export/home/<username> –m <username>

  For instance:

  useradd –d /export/home/eguser –m eguser
- Next, issue the following command to set a password for the above user:

  passwd <username>
- When prompted to provide the password, specify the same.
- Then, proceed to edit the **sudo** script by issuing the following command:

  usr/local/sbin/visudo
- Add the following entries to the script:

```
<username> ALL=NOPASSWD:/opt/SUNWvda/sbin/vda
<username> ALL=NOPASSWD:/usr/sbin/cacaoadm
<username> ALL=NOPASSWD:/opt/SUNWvda/sbin/vda-db-status
<username> ALL=NOPASSWD:/opt/SUNWvda/sbin/vda-webadmin
```

All the tests run by the eG agent should be configured with the full path to the install directory of the **sudo** package;

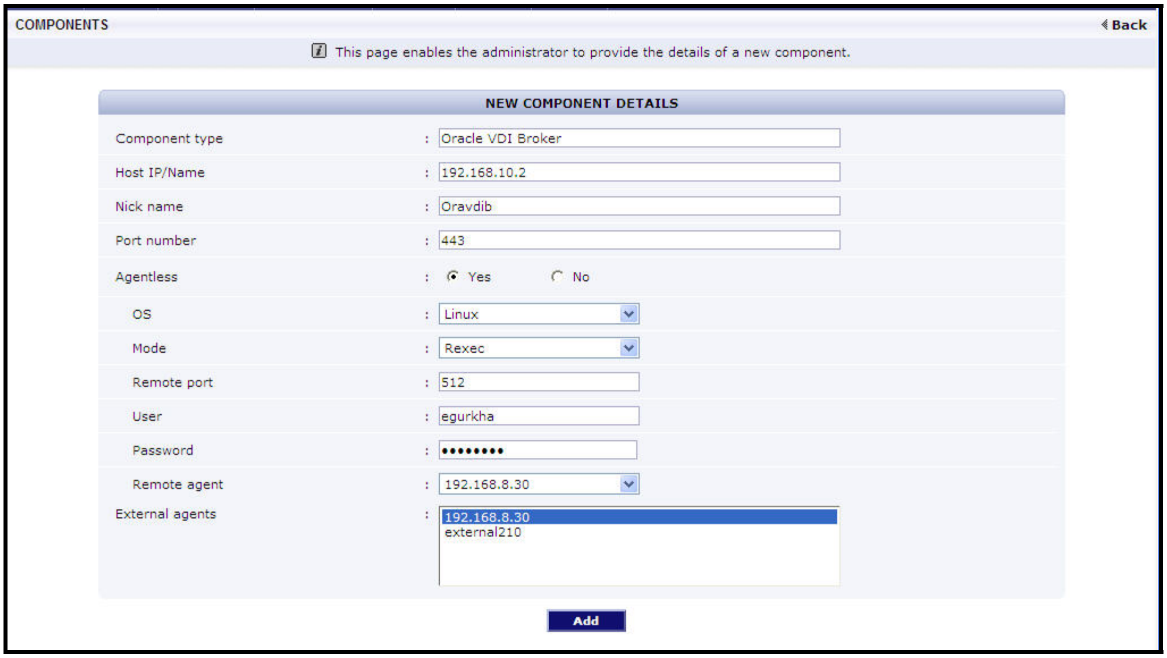
Once these pre-requisites are fulfilled, the eG agent will use the **sudo** package to run the VDA commands and extract the measures for the tests.

2.3 Administering the eG Manager to monitor an Oracle VDI Broker

To achieve this, do the following:

1. Log into the eG administrative interface.
2. eG Enterprise cannot automatically discover the *Oracle VDI Broker*. You need to manually add the component using the add/modify components page (see Figure 2.1) that appears

when the Infrastructure -> Components -> Add/Modify menu sequence is followed. Remember that components manually added are managed automatically. You can either follow the agentless approach or agent based approach to do so. If you set the **Agentless** flag to **Yes**, then select the **OS** as **Linux** and **Mode** as **Rexec** besides providing the required User and password fields.



COMPONENTS

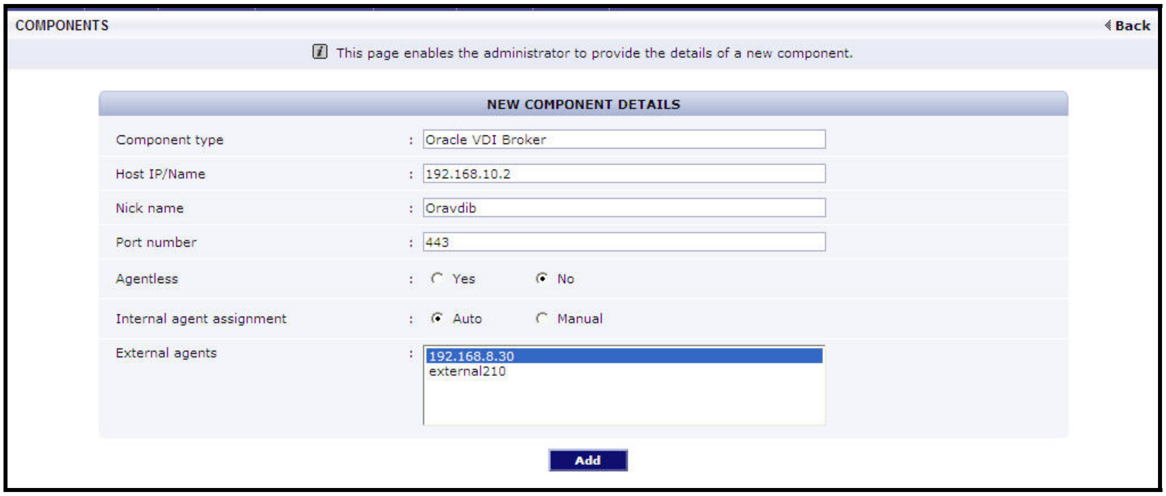
This page enables the administrator to provide the details of a new component.

NEW COMPONENT DETAILS

Component type	:	Oracle VDI Broker
Host IP/Name	:	192.168.10.2
Nick name	:	Oravdib
Port number	:	443
Agentless	:	● Yes ○ No
OS	:	Linux
Mode	:	Rexec
Remote port	:	512
User	:	egurkha
Password	:	••••••••
Remote agent	:	192.168.8.30
External agents	:	192.168.8.30 external210

Figure 2.1: Adding the Oracle VDI Broker in an agentless approach

3. Otherwise you can set the **Agentless** flag to **No** and add the component as shown in Figure 2.2



COMPONENTS

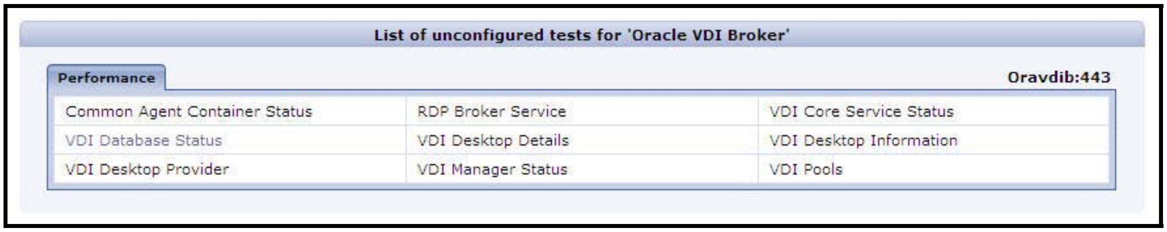
This page enables the administrator to provide the details of a new component.

NEW COMPONENT DETAILS

Component type	:	Oracle VDI Broker
Host IP/Name	:	192.168.10.2
Nick name	:	Oravdib
Port number	:	443
Agentless	:	○ Yes ● No
Internal agent assignment	:	● Auto ○ Manual
External agents	:	192.168.8.30 external210

Figure 2.2: Adding the Oracle VDI Broker in an agent based approach

4. Upon adding the component, try to signout of the administrative interface. A list of unconfigured tests will appear as shown in Figure 2.3.



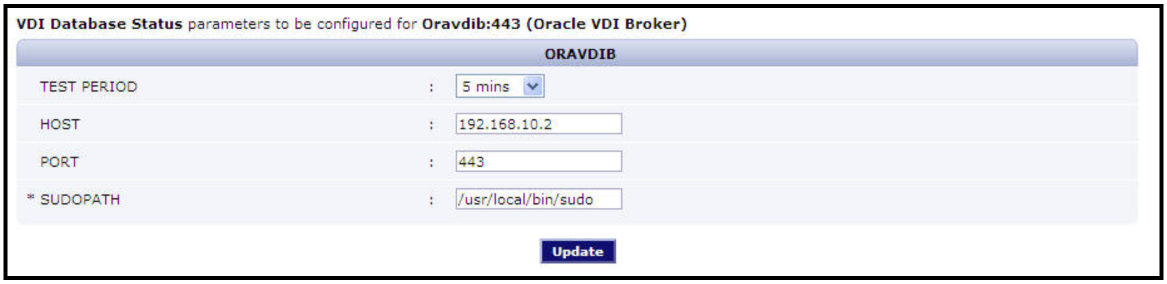
List of unconfigured tests for 'Oracle VDI Broker'

Oravdib:443

Common Agent Container Status	RDP Broker Service	VDI Core Service Status
VDI Database Status	VDI Desktop Details	VDI Desktop Information
VDI Desktop Provider	VDI Manager Status	VDI Pools

Figure 2.3: List of Unconfigured tests for the Oracle VDI Broker

5. Now configure the VDI Database Status test as shown in Figure 2.4.



VDI Database Status parameters to be configured for **Oravdib:443 (Oracle VDI Broker)**

ORAVDIB

TEST PERIOD	:	5 mins
HOST	:	192.168.10.2
PORT	:	443
* SUDOPATH	:	/usr/local/bin/sudo

Figure 2.4: Configuring the VDI Database Status test

6. Specify the following in Figure 2.4

   - **TEST PERIOD** - How often should the test be executed
   - **HOST** - The host for which the test is to be configured
   - **PORT –** Refers to the port used by the specified **HOST**.
   - **SUDOPATH** - This test executes certain privileged VDA (Virtual Desktop Access) commands to pull out the desired metrics from the broker. To enable the test to run these commands, you first need to install a **sudo** package on the broker host. The procedure for installing this package is detailed in Section 1.1 of this document. Once the package is installed, you need to specify the full path to the install directory of the **sudo** package in the **SUDOPATH** text box.

7. Finally, click on the **Update** button to register the changes and signout of the administrative interface.

2.4 Monitoring the Oracle VDI broker

To monitor the *Oracle VDI Broker*, do the following:

1. Login as a monitor / supermonitor user.
2. Click on the components option in the menu bar, and select the Servers option from the components menu.
3. From the **component list** page, click on the *Microsoft Oracle VDI Broker* component for which you wish to view measurements.

Chapter

3

Conclusion

This document has described in detail the steps for configuring and monitoring the **Oracle VDI**. 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.