User's Manual

VMware™ ESX Server™

Version 1.0

vmware™

VMware, Inc.

3145 Porter Drive, Bldg. F
Palo Alto, CA 94304
www.vmware.com

Please note that you will always find the most up-to-date technical documentation on our Web site at http://www.vmware.com/support/.

The VMware Web site also provides the latest product updates.

Table of Contents

1

Introduction to VMware ESX Server

Introducing VMware ESX Server

Partition Your Servers for Mainframe-Class Service. The Logical Way.

As an information technology professional, you constantly face the challenge of doing more with less. You must respond to a growing number of requests for server applications across a daunting variety of hardware and software platforms. And because running these applications on the same server may cause instability, you are continually buying new servers and integrating them into your environment. This further complicates your management task.

And while you're adding servers, you must still deliver performance, high availability and security without missing a beat — all the while holding the line on IT budgets and staffing. Sound familiar?

Now You Can Think Inside the Box

VMware™ ESX Server™ enables you to build and manage a fast-changing enterprise computing infrastructure using Intel computers safely, reliably and cost effectively.

What's our secret? VMware is the only company to virtualize the Intel hardware architecture, which enables us for the first time to bring mainframe-class virtual machine technology, memory-based virtual networking and advanced workload management to Intel computers. As a result, VMware ESX Server dramatically reduces the number of servers you need to support new applications and lets you manage the servers you use more efficiently. ESX Server also accelerates development, testing and deployment while increasing reliability and security.

How does it work? With ESX Server, you can create multiple virtual machines that run simultaneously on a single physical computer. Each virtual machine is configured with its own operating system, applications and network identity. And each can use any program that runs on a physical server. Even applications that require unusual operating system patches, device drivers or network configurations can be run in a virtual machine.

Our memory-based virtual networking lets you achieve speeds comparable to Gigabit Ethernet between virtual machines without adding additional hardware.

For applications in which you need additional resources temporarily or in which you must guarantee service levels over time, VMware server software enables you to dynamically allocate resources across virtual machines at any level of granularity.

And you can remotely manage virtual machines individually or as a group from any desktop with an Internet connection.

Consolidate Servers To Maximize ROI

As you well know, the cost of adding new servers to your environment is much more than just the price of a new machine. Floor and rack space are probably at a premium in your organization. And uninterruptible power and air conditioning aren't free either.

Most costly is the time required from your system administrators to get each new server installed and configured, then perform regular backup services and manage resources across servers to levels you have guaranteed.

In addition to being able to run many virtual machines on a single physical server, each VMware virtual machine has all the attributes of a physical machine, including isolation from other virtual machines. This ensures full security, fault protection and service level guarantees. And VMware virtual machines offer high-speed network and disk throughput, which means that many applications run at near-native performance.

But the benefits of using ESX Server really add up when you consolidate your applications onto highly scalable, highly reliable, enterprise-class servers using virtual machines. Instead of 20 low-cost servers for 20 applications — and the reliability and management challenges that come with them — you can consolidate them all onto a single multiprocessor server with high availability features such as RAID storage and hot-swappable components.

Simplify Development and Testing

If you develop or test server software, you know what a chore it is to configure and maintain the servers needed to represent all the platforms you support.

ESX Server provides an ideal platform for hosting development and test servers. Rather than dedicating a physical server to each environment, you can run test servers in virtual machines. A VMware virtual machine can be configured to match exactly the hardware, operating system, networking and installed applications you need in your build and test environments. Virtual machines can be booted up and powered down independently. And if you need it, test runs can be allocated additional hardware resources for the duration of the run.

Once configured, a VMware virtual machine can be saved in a virtual disk file. By maintaining your preconfigured virtual disk files on a file server, you can quickly download and start exactly the test environment you need.

VMware disk-in-a-file features are especially useful for build and test work. If virtual disks are set to nonpersistent mode, you can instantly recover from corrupted files by simply restarting the virtual machine. And our undoable disk feature makes it easy to modify a baseline virtual server with new software and commit the changes to a new virtual disk when you are satisfied with them.

Fast Track Deployment to Efficiently Scale Your Infrastructure

Another complex and costly activity is rolling out a new application or upgrading existing ones. Your users may be distributed around the world and chances are they use a variety of diverse hardware. This makes it difficult to automate or even simplify the deployment process.

Because ESX Server masks the differences in the underlying physical computers, you now have a standard platform not only for developing and testing your applications but also for deploying them across a diverse mix of hardware. Because VMware virtual machines are self-contained in just a few files, you can easily duplicate and install them, even at remote locations.

Using VMware scripting capabilities, you can even integrate deployment into a Web-based system, allowing your users to install new applications safely and securely themselves.

Improve Availability With Low-Cost Hot Standby and Instant Start Up

As network applications become more important to your organization, maximizing server uptime is increasingly critical. It’s no longer sufficient to be alerted to a failed server by a user complaint or a pager message sent after the server has failed to respond for a long period of time.

With ESX Server, it’s a simple matter to keep backup copies of critical server virtual machines in a suspended state, ready to take over for a failed primary server in seconds. ESX Server provides a suspend/resume feature that saves the memory state of a running virtual machine in a file so that it can be resumed quickly, with no boot delay.

Partition Super Servers — Logically

Today’s high-end enterprise servers typically offer hardware partitioning, making it possible to divide them up into multiple nodes, each running its own application. As valuable as this is, the feature typically is limited to partitions of four processors or more and doesn’t allow you to dynamically reconfigure partitions while the server is running.

ESX Server is an ideal complement to hardware partitioning because it adds a level of logical partitioning that enables you to dynamically allocate resources at the application level. In addition, ESX Server provides strong isolation from failure of any other applications or operating environments in the same hardware partition.

Host Multiple Web Sites on a Single Server Securely

Because of customers' concerns about resource availability and security, Web hosting companies typically must dedicate individual servers to each customer's Web site. This is especially wasteful when you consider that many of these servers are lightly loaded.

ESX Server lets you host multiple customers' Web sites on a single server by isolating each in its own virtual machine. As far as each Web site is concerned, it has a complete machine dedicated to it. Customers can rest assured that the resources they need will be there when they need them. And Web hosting companies enjoy the benefits of reduced hardware expenses and greatly simplified management.

One Computer. Multiple Worlds.

All VMware software is based on our patent-pending MultipleWorlds™ technology, a thin software layer that sits between the Intel architecture and the operating system, virtualizing the hardware and managing all hardware resources. MultipleWorlds technology combines mainframe-class virtual machine technology with memory-based virtual networking and advanced workload management capabilities. With MultipleWorlds technology, you can build and manage a dynamic, diverse and distributed enterprise computing infrastructure using Intel computers safely, reliably and cost effectively.

System Requirements

Server Hardware Requirements

Minimum

- Processor: Intel Pentium II 500MHz and above, AMD Athlon Model 2 and above, or AMD Duron
- 256MB RAM minimum, 512MB or more recommended
- An Ethernet card — Intel EEPro, 3Com or Tulip-based. (An Intel EEPro card is recommended for best performance.)
- A SCSI adapter, Fibre Channel adapter or internal RAID controller.

  The basic SCSI adapters supported are Adaptec, BusLogic and most NCR/Symbios SCSI adapters. The SCSI RAID adapters supported are Compaq SmartArray, Dell PercRaid (Adaptec RAID and AMI MegaRAID), IBM ServeRAID and Mylex SCSI RAID devices. The Fibre Channel adapters that are supported are Emulex and Qlogic adapters.
- A SCSI disk or RAID LUN with unpartitioned space. In a minimum configuration, this disk or RAID will be shared between the console operating system and the virtual machines.

Recommended for Enhanced Performance

- A second Ethernet adapter
- A second disk controller with one or more drives.

The lists above outline a basic configuration. In practice, you may use multiple physical disks, which may be SCSI disks, Fibre Channel disks or RAID LUNs. For best performance, all of the data used by the virtual machines should be on the physical disks allocated to virtual machines. Therefore, these physical disks should be large enough to hold disk images that will be used by all the virtual machines.

Similarly, you should provide enough RAM for all of the virtual machines plus the console operating system. For background on the console operating system, see Characteristics of the Console Operating System on page 82. For details on how to calculate the amount of RAM you will need, see How the System Uses Memory on page 114.

Note: To ensure the best possible I/O performance and workload management, VMware ESX Server provides its own drivers for supported devices. Be sure that the devices you plan to use in your server are supported. For additional detail on I/O device compatibility, download the VMware ESX Server I/O Adapter Compatibility Guide from the VMware Web site: *http://www.vmware.com/pdf/esx_io_devices.pdf*.

ESX Server virtual machines can share an Ethernet card with the console operating system (as described in Sharing Network Adapters and Virtual Networks on page 126). For best performance, however, you should configure the virtual machines to use an Ethernet card separate from the one used by the console operating system.

ESX Server virtual machines can share a SCSI disk with the console operating system, but for enhanced disk performance, you can configure the virtual machines to use a SCSI adapter and disk separate from those used by the console operating system. You should make sure enough free disk space is available to install the guest operating system and applications for each virtual machine on the disk that they will use.

Supported Guest Operating Systems

- Windows 2000 (any server version)
- Windows NT 4.0 — Service Pack 4 or higher
- Red Hat Linux 6.2 or 7.0

Note: The standard Linux kernels in Red Hat Linux 6.2 have a bug reported at *http://www.redhat.com/support/errata/RHBA-2000013-01.html* that can cause data corruption under heavy memory load. Therefore, the standard Red Hat 6.2 installation should not be used as a guest operating system to run server applications unless it is patched. One way to correct the problem is to recompile the guest Linux kernel with the configuration option CONFIG_X86_FX=n.

Remote Management Workstation Requirements

The remote workstation is a Windows NT 4.0, Windows 2000 or Linux system from which you launch a remote console and access the Web-based management interface using a browser.

Hardware Requirements

- Standard x86-based computer
- 266MHz or faster processor
- 64MB RAM minimum
- 10MB free disk space required for basic installation

Software — Windows Remote Workstation

In Windows, the remote console runs as a standalone application. The Web-based management interface uses a Web browser.

- Windows NT 4.0 Workstation or Server, Service Pack 6a
- Windows 2000 Professional, Server or Advanced Server
- The Web-based management interface requires one of these browsers:
  - Microsoft Internet Explorer 5.0 or higher
  - Netscape Navigator 4.5 or higher

Software — Linux Remote Workstation

Compatible with standard Linux distributions with glibc version 2 or higher and one of the following:

- For single-processor systems: kernel 2.0.32 or higher in the 2.0.x series, or kernel in the 2.2.x series
- For SMP systems: kernel in the 2.2.x series

  Note: If you are using Linux kernel 2.2.14-5.0 — in a standard Red Hat Linux 6.2 installation, for example — you may want to use the patch described in the section Supported Guest Operating Systems on page 13.
- The Web-based management interface requires Netscape Navigator 4.5 or higher

Technical Support Resources

The VMware Web Site

The latest technical support and troubleshooting notes are available on the VMware Web site at *http://www.vmware.com/support/*.

VMware Newsgroups

The VMware newsgroups are primarily forums for users to help each other. You are encouraged to read and post issues, work-arounds and fixes. While VMware personnel may read and post to the newsgroups, they are not a channel for official support. The VMware NNTP news server is at *news.vmware.com*.

The following groups are devoted to ESX Server issues:
vmware.esx-server.configuration
vmware.esx-server.guestos
vmware.esx-server.installation
vmware.esx-server.misc
vmware.esx-server.web-mgmt.misc

Reporting Problems

If you have problems while running VMware ESX Server, please report them to the VMware support team. Problems may occur either in the VMkernel or in the virtual machines that it hosts. A problem in the VMkernel will typically freeze the physical machine, while a problem in a virtual machine will generate a core file.

These guidelines describe the information we need from you to diagnose various types of problems.

- If a virtual machine exits abnormally or crashes, please save the log file (`vmware.log` in the same directory as your `.cfg` file) and any core files (`core` or `vmware-core`). Provide these to VMware along with the virtual machine's configuration (`.cfg`) file and any other information that might help us to reproduce the problem. In addition, include the contents of `/var/log/messages` from the console operating system, since the VMkernel logs informational and error messages in `/var/log/messages`. Be sure to include a description of your physical hardware and of the software (operating system and applications) that was running in the virtual machine. Include this information in your report to the VMware support team.
- Any problem in the VMkernel (for example, an ASSERT failure) causes the VMkernel to display an error screen for a period of time and then reboot the

machine. In this case, provide VMware with the steps you took to cause this failure (including any information listed in point 1 above, if applicable). Include this information in your report to the VMware support team, along with the contents of `/var/log/messages` from the console operating system.

If you specified a VMware core dump partition when you configured your machine, the VMkernel will generate a core dump upon a panic or an exception. This will create files named `vmkernel-core.<date>` and `vmkernel-log.<date>` in the `/root` directory. Include this information in your report to the VMware support team.

Be sure to register your serial number. You may then report your problems using the ESX Server incident report form on the VMware Web site at *http://www.vmware.com/forms/incident_loginesx*.

2

Installing and Configuring ESX Server

Installing the Software on the Server

This installation guide will step you through the process of installing and configuring the VMware ESX Server software on your server.

Later sections explain how to create and provision virtual machines, how to manage your virtual machines from a remote workstation, and how to work with the advanced features of VMware ESX Server.

Before You Begin

To install VMware ESX Server, you will need

- The VMware ESX Server software CD, which includes the VMware Console Operating System, VMware ESX Server software, and remote console software.
- A computer that meets the system requirements for ESX Server. See page 12 for details.

Installing VMware ESX Server

The VMware ESX Server installation includes the VMware Console Operating System, the `vmnixmod` module, the VMkernel, and VMkernel modules. The console operating system is based on a modified Red Hat Linux 6.2 installation and is called VMnix. It is used to configure, start and administer VMware virtual machines. The `vmnixmod` module is loaded into the VMnix kernel to facilitate loading and communicating with the VMkernel. The VMkernel manages system hardware and the virtual machines running on the server. Users communicate with the VMkernel via the console operating system.

The VMkernel manages all the operating systems on the machine, including both the console operating system and the operating systems running on each virtual machine. The VMkernel modules provide support for high-performance device I/O and allow run-time addition of functionality to the VMkernel (for example, network traffic filters).

Before you begin, be sure you have the network information you will need during installation. You will need to know

- The IP address for the server where you are installing ESX Server
- The host name for the server
- The netmask for the server’s subnet
- The IP address of the gateway
- The IP address of the name server

Installing the VMware ESX Server Software

1. Make sure the network cable is plugged into the main network adapter, so the installer will properly detect that the machine has a network card.
2. Insert the VMware ESX Server CD in the CD-ROM drive and power on the machine.
3. If necessary, enter the BIOS Setup screen and set the CD-ROM as the first boot device. The machine should display a screen saying, "Welcome to the VMware ESX Server Install."
4. If the installation screen does not come up properly, your CD-ROM drive may be having trouble booting from the CD. In this case, you may want to try booting from a VMware ESX Server boot floppy. Use the following steps to create a boot floppy.

   Windows System

   - Put the ESX Server CD in the CD-ROM drive.
   - Put a floppy disk in the floppy drive.
   - Bring up a DOS command window.
   - Use the `rawrite` program to copy the disk image to the floppy disk. If your CD-ROM drive is not `d:`, substitute the correct drive letter.

     ```
     d:\dosutils\rawrite -f d:\boot.img -d a
     ```

   Linux System

   - Put the ESX Server CD in the CD-ROM drive.
   - As root, mount the CD.

     ```
     mount /dev/cdrom /mnt/cdrom
     ```

   - Put a floppy disk in the floppy drive.
   - Copy the boot image from the CD to the floppy.

     ```
     dd if=/mnt/cdrom/boot.img of=/dev/fd0 bs=1474560 \
     count=1
     ```

     Note: The command should all be typed on one line. Do not type the backslash.

   Then insert the floppy disk in the floppy drive, reboot, and, if necessary, make the floppy drive the first boot device. You should leave the CD in the CD-ROM drive.

5. The first part of the setup process installs the Linux-based console operating system. When you reach the disk setup portion of the installation, be sure to follow these steps.

A. In Disk Setup, choose Disk Druid.

B. In Current Disk Partitions, delete any existing partitions.

C. Next, choose the disk where you will install the console operating system. It should be your first IDE disk (`hda`), if you have one; otherwise, use your first SCSI disk (`sda`).

D. You typically create three partitions for a Linux installation, using the Add option. The first partition should have a mount point of `/boot`, a size of 20MB, and a type of Linux native. The second partition should have no mount point, a size of about 128MB, and a type of Linux swap. The third partition should have a mount point of `/`, a size of about 1800MB, and a type of Linux native. The third partition will be your root file system, and most of the Linux and console operating system files will be installed there.

Respond OK when you have created these three partitions, and respond Yes to the Save Changes prompt.

Note: Do not create partitions on any other disks besides the main boot disk.

Respond OK in Choose Partitions to Format.

6. Enter the host name in Hostname Configuration. Include the full domain name if you are running with domains.

7. In Network Configuration, disable bootp/dhcp and enter the required network parameters. Setup will not ask for network parameters if you do not have a network card. Initially, only the first Ethernet card will be enabled. All other network adapters will be disabled.

   See Using DHCP for the Console Operating System on page 83 for instructions and cautions on setting up a DHCP-based console operating system.

8. At the Time Zone Selection screen, choose your time zone.

9. At the Root Password screen, specify your desired root password. In Add User and User Account Setup, you can add additional user accounts.

   You will need accounts for all users who need to log in to the Web-based management interface to create or run virtual machines. If you wish, you may add those users at this time. However, you may find it more convenient to add them later with the console operating system's `useradd` command or by copying the `/etc/passwd` file from another machine.

10. The installer will then format the disk and start installing the packages.

11. At the final Complete screen, respond OK. The system will reboot and you can begin configuring your server.

Using the Setup Wizard to Configure Your Server

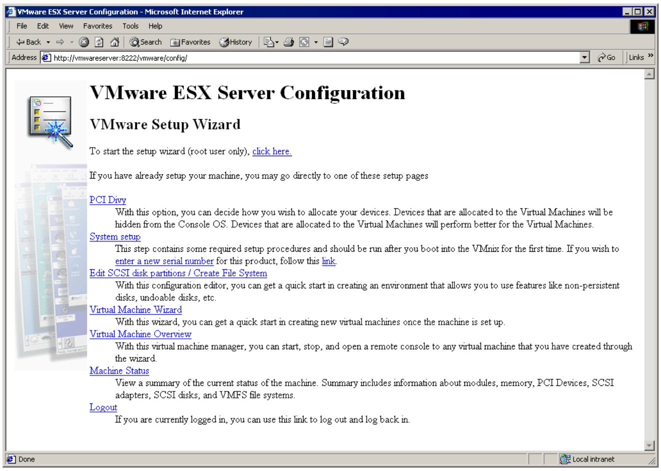
A Web-based Setup Wizard will guide you through the steps to configure your server. You may return to the wizard at any time to edit your configuration. You may run the Setup Wizard from any computer with network access to your server. Running X on your server's console operating system is not recommended. The steps that follow assume you are using a separate computer as your workstation.

Note: If you need secure communications between your management workstations and the server, you will need to install appropriate software to provide that security. For additional details, see the Network Security section in the technical note Authentication and Security Features in VMware ESX Server on page 99.

1. Launch a supported Web browser (Microsoft Internet Explorer 5.0 or later or Netscape Navigator 4.5 or later) and enter the URL for the VMware Setup Wizard

```
http://<hostname>:8222/vmware/config/
```

2. Log in as root.


VMware ESX Server Configuration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
History
Address http://vmwareserver:8222/vmware/config/
Go
Links
Done
Local intranet

VMware ESX Server Configuration

VMware Setup Wizard

To start the setup wizard (root user only), click here.

If you have already setup your machine, you may go directly to one of these setup pages

PCI Divy
With this option, you can decide how you wish to allocate your devices. Devices that are allocated to the Virtual Machines will be hidden from the Console OS. Devices that are allocated to the Virtual Machines will perform better for the Virtual Machines.

System setup
This step contains some required setup procedures and should be run after you boot into the VMnix for the first time. If you wish to enter a new serial number for this product, follow this link.

Edit SCSI disk partitions / Create File System
With this configuration editor, you can get a quick start in creating an environment that allows you to use features like non-persistent disks, undoable disks, etc.

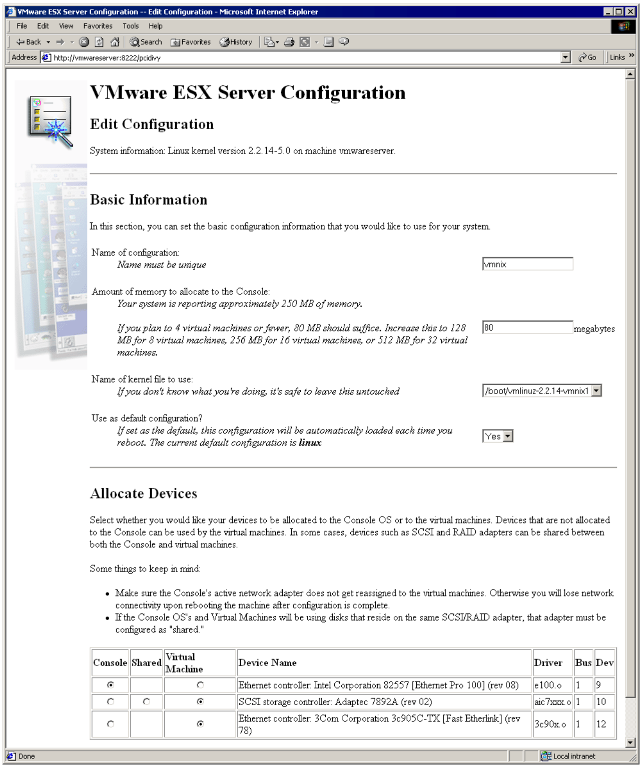
Virtual Machine Wizard
With this wizard, you can get a quick start in creating new virtual machines once the machine is set up.

Virtual Machine Overview
With this virtual machine manager, you can start, stop, and open a remote console to any virtual machine that you have created through the wizard.

Machine Status
View a summary of the current status of the machine. Summary includes information about modules, memory, PCI Devices, SCSI adapters, SCSI disks, and VMFS file systems.

Logout
If you are currently logged in, you can use this link to log out and log back in.

3. Start the wizard by clicking the `click here` link at the top of the page.


VMware ESX Server Configuration -- Edit Configuration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/pcidivy
Go Links
Done
Local intranet

VMware ESX Server Configuration

Edit Configuration

System information: Linux kernel version 2.2.14-5.0 on machine vmwareserver.

Basic Information

In this section, you can set the basic configuration information that you would like to use for your system.

Name of configuration:
Name must be unique — vmnix

Amount of memory to allocate to the Console:
Your system is reporting approximately 250 MB of memory.

If you plan to 4 virtual machines or fewer, 80 MB should suffice. Increase this to 128 MB for 8 virtual machines, 256 MB for 16 virtual machines, or 512 MB for 32 virtual machines. — 80 megabytes

Name of kernel file to use:
If you don't know what you're doing, it's safe to leave this untouched — /boot/vmlinuz-2.2.14-vmnix1

Use as default configuration?
If set as the default, this configuration will be automatically loaded each time you reboot. The current default configuration is ***linux*** — Yes

Allocate Devices

Select whether you would like your devices to be allocated to the Console OS or to the virtual machines. Devices that are not allocated to the Console can be used by the virtual machines. In some cases, devices such as SCSI and RAID adapters can be shared between both the Console and virtual machines.

Some things to keep in mind:

- Make sure the Console's active network adapter does not get reassigned to the virtual machines. Otherwise you will lose network connectivity upon rebooting the machine after configuration is complete.
- If the Console OS's and Virtual Machines will be using disks that reside on the same SCSI/RAID adapter, that adapter must be configured as "shared."

Console	Shared	Virtual Machine	Device Name	Driver	Bus	Dev
●		○	Ethernet controller: Intel Corporation 82557 [Ethernet Pro 100] (rev 08)	e100.o	1	9
○	○	●	SCSI storage controller: Adaptec 7892A (rev 02)	aic7xxx.o	1	10
○		●	Ethernet controller: 3Com Corporation 3c905C-TX [Fast Etherlink] (rev 78)	3c90x.o	1	12

4. Confirm that the defaults in the Basic Information section are appropriate for your server. The default amount of memory reserved for the console operating system — 80MB — is sufficient for managing up to three or four virtual machines.

5. Allocate storage and network adapters to be used by the console operating system and virtual machines on the server. Be sure that both the console operating system and the virtual machines have access to some device in each category.

   Storage: A SCSI or RAID adapter should be shared if you want to use that adapter or array for both the console operating system and virtual machines.

When you are allocating SCSI or RAID devices, the unit of device allocation is a PCI card device. You may connect multiple SCSI or RAID disks, CD-ROM drives, tape drives and other devices to the SCSI or RAID adapter.

You should give as many SCSI or RAID devices to the virtual machines as possible to ensure that the majority of your mass storage resources are used by your virtual machines. If you do not have any IDE disks, you may have to allocate at least one SCSI or RAID device to the console operating system, since the console operating system needs to have a disk from which it and the VMkernel can boot.

SCSI and RAID devices can be shared between the console operating system and virtual machines.

Some adapter cards have multiple functions, which means there are multiple adapters on each card. When you allocate a SCSI or RAID device to the console operating system or to the VMkernel, you are effectively allocating all the SCSI or RAID disks, CD-ROM drives and other attached devices along with the adapter. As a result, you have only coarse-grained control over how you allocate SCSI and RAID devices.

Consider this example. Suppose your machine has SCSI adapters `scsi0` and `scsi1` that are on the same SCSI adapter card. If you choose to share one of the adapters, you must share both. Similarly, if you choose to allocate one of the adapters for use by virtual machines, you must allocate both for use by virtual machines.

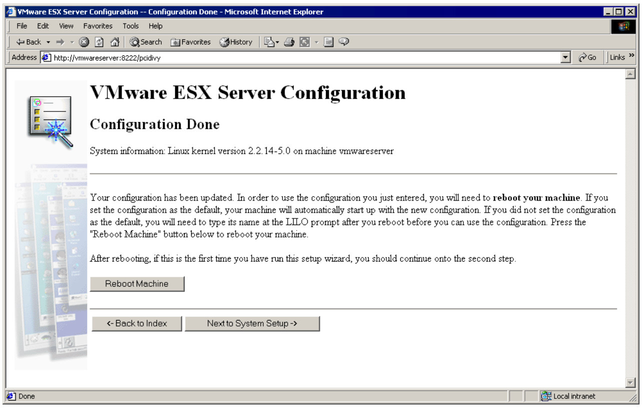
Network: It is generally best to assign the first Ethernet adapter on the list to the console operating system and set the other adapters to be used by virtual machines. If you assign the first adapter to be used by virtual machines, the console operating system may try to use an inappropriate driver for its network adapter. Ethernet adapters cannot be shared between the console operating system and virtual machines at this stage. To configure a shared adapter later, see Sharing Network Adapters and Virtual Networks on page 126.

As with SCSI and RAID controllers, the unit of device allocation is a PCI card. Some network adapter cards are multifunction PCI cards, which means there are multiple adapters on each card. Only one network adapter is displayed in the list of devices. When you allocate that device to the console operating system or to the VMkernel, you are effectively allocating all the adapters on that card.

It is generally good to give as many network adapters to the virtual machines as possible. Doing so will help ensure that the majority of your network resources will be devoted to the virtual machines. As the console operating system is intended primarily as a management interface, you should minimize resources allocated to the console operating system. You will need to allocate at least one Ethernet device to the console operating system in order to manage your ESX Server machine remotely.

In the likely event that you have fewer Ethernet devices than virtual machines, you can share VMkernel Ethernet adapters among the virtual machines with little performance penalty.

6. Click Save Configuration.



VMware ESX Server Configuration -- Configuration Done - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/pcidivy
Go Links
Reboot Machine
<- Back to Index
Next to System Setup ->
Done
Local intranet

VMware ESX Server Configuration

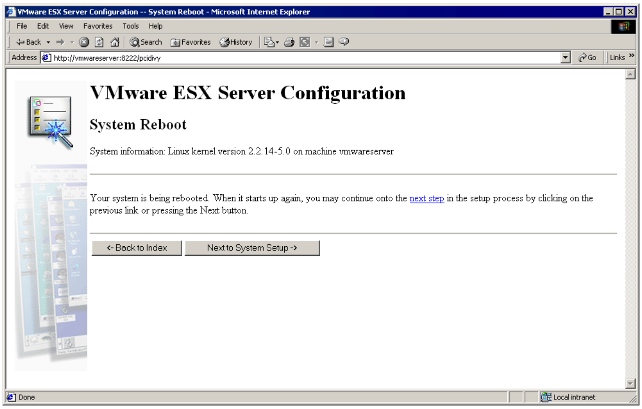
Configuration Done

System information: Linux kernel version 2.2.14-5.0 on machine vmwareserver

Your configuration has been updated. In order to use the configuration you just entered, you will need to **reboot your machine**. If you set the configuration as the default, your machine will automatically start up with the new configuration. If you did not set the configuration as the default, you will need to type its name at the LILO prompt after you reboot before you can use the configuration. Press the "Reboot Machine" button below to reboot your machine.

After rebooting, if this is the first time you have run this setup wizard, you should continue onto the second step.

7. Click Reboot Machine to restart your server using the configuration you just set up.


VMware ESX Server Configuration -- System Reboot - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/pcidivy
Go Links
Done
Local intranet

VMware ESX Server Configuration

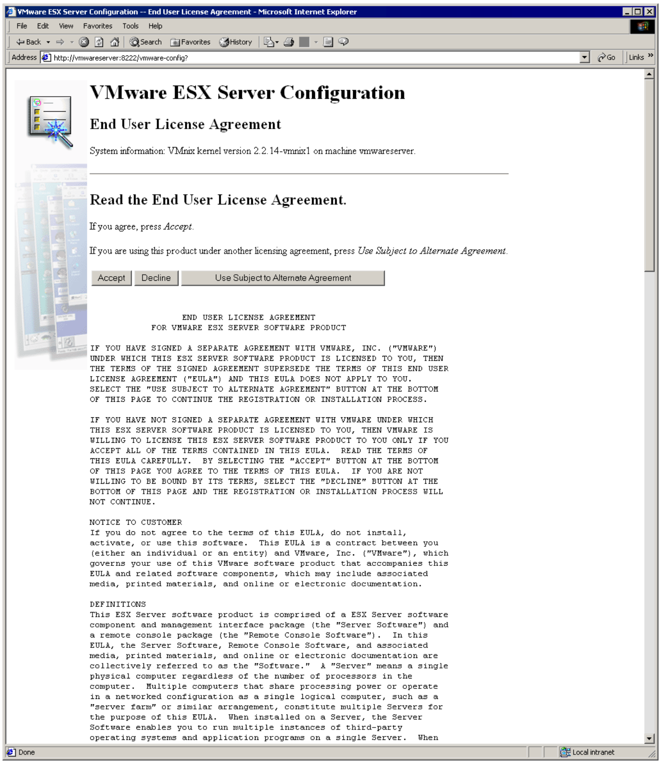
System Reboot

System information: Linux kernel version 2.2.14-5.0 on machine vmwareserver

Your system is being rebooted. When it starts up again, you may continue onto the next step in the setup process by clicking on the previous link or pressing the Next button.

<- Back to Index
Next to System Setup ->

8. After the server has rebooted, click Next to System Setup.


VMware ESX Server Configuration -- End User License Agreement - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/vmware-config?
Go Links
Accept
Decline
Use Subject to Alternate Agreement
Done
Local intranet

VMware ESX Server Configuration

End User License Agreement

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

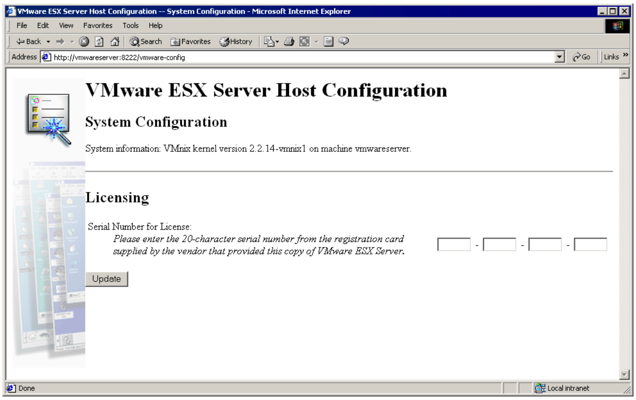
Read the End User License Agreement.

If you agree, press *Accept*.

If you are using this product under another licensing agreement, press *Use Subject to Alternate Agreement*.

```
                  END USER LICENSE AGREEMENT
            FOR VMWARE ESX SERVER SOFTWARE PRODUCT

IF YOU HAVE SIGNED A SEPARATE AGREEMENT WITH VMWARE, INC. ("VMWARE")
UNDER WHICH THIS ESX SERVER SOFTWARE PRODUCT IS LICENSED TO YOU, THEN
THE TERMS OF THE SIGNED AGREEMENT SUPERSEDE THE TERMS OF THIS END USER
LICENSE AGREEMENT ("EULA") AND THIS EULA DOES NOT APPLY TO YOU.
SELECT THE "USE SUBJECT TO ALTERNATE AGREEMENT" BUTTON AT THE BOTTOM
OF THIS PAGE TO CONTINUE THE REGISTRATION OR INSTALLATION PROCESS.

IF YOU HAVE NOT SIGNED A SEPARATE AGREEMENT WITH VMWARE UNDER WHICH
THIS ESX SERVER SOFTWARE PRODUCT IS LICENSED TO YOU, THEN VMWARE IS
WILLING TO LICENSE THIS ESX SERVER SOFTWARE PRODUCT TO YOU ONLY IF YOU
ACCEPT ALL OF THE TERMS CONTAINED IN THIS EULA.  READ THE TERMS OF
THIS EULA CAREFULLY.  BY SELECTING THE "ACCEPT" BUTTON AT THE BOTTOM
OF THIS PAGE YOU AGREE TO THE TERMS OF THIS EULA.  IF YOU ARE NOT
WILLING TO BE BOUND BY ITS TERMS, SELECT THE "DECLINE" BUTTON AT THE
BOTTOM OF THIS PAGE AND THE REGISTRATION OR INSTALLATION PROCESS WILL
NOT CONTINUE.

NOTICE TO CUSTOMER
If you do not agree to the terms of this EULA, do not install,
activate, or use this software.  This EULA is a contract between you
(either an individual or an entity) and VMware, Inc. ("VMware"), which
governs your use of this VMware software product that accompanies this
EULA and related software components, which may include associated
media, printed materials, and online or electronic documentation.

DEFINITIONS
This ESX Server software product is comprised of a ESX Server software
component and management interface package (the "Server Software") and
a remote console package (the "Remote Console Software").  In this
EULA, the Server Software, Remote Console Software, and associated
media, printed materials, and online or electronic documentation are
collectively referred to as the "Software."  A "Server" means a single
physical computer regardless of the number of processors in the
computer.  Multiple computers that share processing power or operate
in a networked configuration as a single logical computer, such as a
"server farm" or similar arrangement, constitute multiple Servers for
the purpose of this EULA.  When installed on a Server, the Server
Software enables you to run multiple instances of third-party
operating systems and application programs on a single Server.  When
```

9. Click Accept to accept the VMware license.


VMware ESX Server Host Configuration -- System Configuration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/vmware-config
Go Links
Done
Local intranet

VMware ESX Server Host Configuration

System Configuration

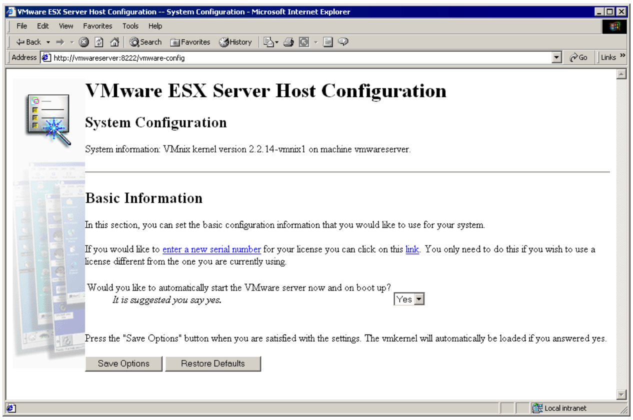
System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Licensing

Serial Number for License:
Please enter the 20-character serial number from the registration card supplied by the vendor that provided this copy of VMware ESX Server. ___ - ___ - ___ - ___

Update

10. Enter your serial number, then click Update.


VMware ESX Server Host Configuration -- System Configuration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/vmware-config
Go Links
Local intranet

VMware ESX Server Host Configuration

System Configuration

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Basic Information

In this section, you can set the basic configuration information that you would like to use for your system.

If you would like to enter a new serial number for your license you can click on this link. You only need to do this if you wish to use a license different from the one you are currently using.

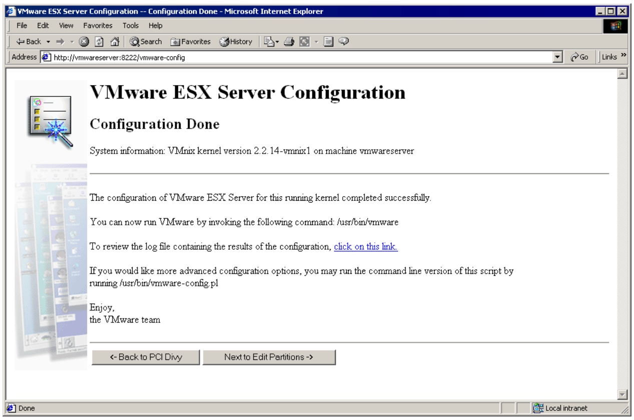
Would you like to automatically start the VMware server now and on boot up? Yes
It is suggested you say yes.

Press the "Save Options" button when you are satisfied with the settings. The vmkernel will automatically be loaded if you answered yes.

Save Options | Restore Defaults

11. If necessary, change the server start-up setting. In most cases, the defaults will be appropriate.

12. Click Save Options.


VMware ESX Server Configuration -- Configuration Done - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
History
Address http://vmwareserver:8222/vmware-config
Go
Links
Done
Local intranet

VMware ESX Server Configuration

Configuration Done

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver

The configuration of VMware ESX Server for this running kernel completed successfully.

You can now run VMware by invoking the following command: /usr/bin/vmware

To review the log file containing the results of the configuration, click on this link.

If you would like more advanced configuration options, you may run the command line version of this script by running /usr/bin/vmware-config.pl

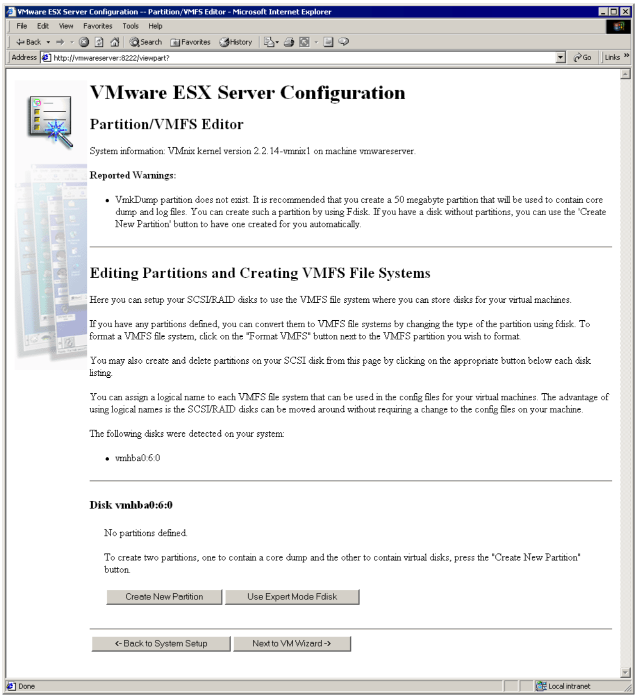
Enjoy,
the VMware team

Basic configuratiton of your server is complete.

Click Next to Edit Partitions.

Using an Entire SCSI Disk or RAID Array for Virtual Machines

If you have a SCSI disk or RAID array in addition to the disk or array that holds the console operating system, you will see the following screen. If you have only one SCSI disk or RAID array, skip to the next section (page 31). For background on how SCSI devices are identified, see Determining SCSI Target IDs on page 111.


VMware ESX Server Configuration -- Partition/VMFS Editor - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/viewpart?
Go Links
Create New Partition
Use Expert Mode Fdisk
<- Back to System Setup
Next to VM Wizard ->
Done
Local intranet

VMware ESX Server Configuration

Partition/VMFS Editor

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Reported Warnings:

- VmkDump partition does not exist. It is recommended that you create a 50 megabyte partition that will be used to contain core dump and log files. You can create such a partition by using Fdisk. If you have a disk without partitions, you can use the 'Create New Partition' button to have one created for you automatically.

Editing Partitions and Creating VMFS File Systems

Here you can setup your SCSI/RAID disks to use the VMFS file system where you can store disks for your virtual machines.

If you have any partitions defined, you can convert them to VMFS file systems by changing the type of the partition using fdisk. To format a VMFS file system, click on the "Format VMFS" button next to the VMFS partition you wish to format.

You may also create and delete partitions on your SCSI disk from this page by clicking on the appropriate button below each disk listing.

You can assign a logical name to each VMFS file system that can be used in the config files for your virtual machines. The advantage of using logical names is the SCSI/RAID disks can be moved around without requiring a change to the config files on your machine.

The following disks were detected on your system:

- vmhba0:6:0

Disk vmhba0:6:0

No partitions defined.

To create two partitions, one to contain a core dump and the other to contain virtual disks, press the "Create New Partition" button.

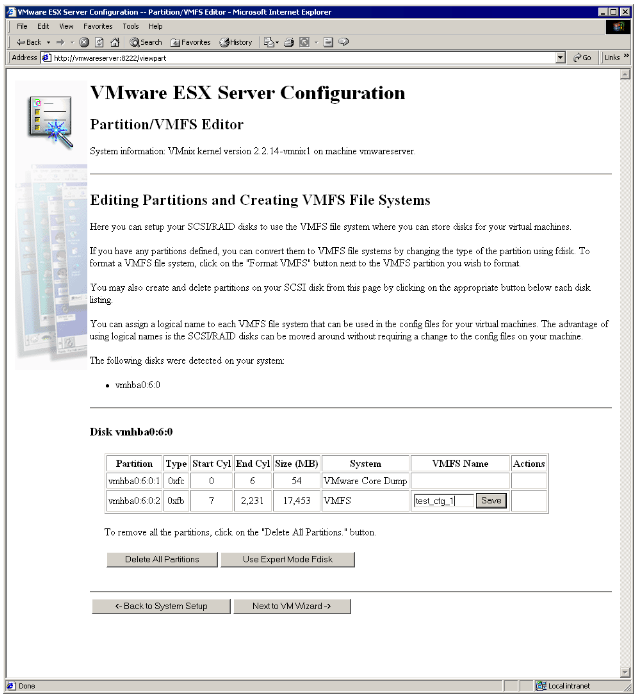
1. Set up partitions for your virtual machines.

   In this example, you want to make all of disk `vmhba0:6:0` available to store virtual machine files.

   Click Create New Partition to create a small core dump partition and a VMFS (VMware ESX Server file system) partition that uses the rest of the space available on the disk or array.

The VMFS partition provides high-performance access to the virtual machine's files — essentially the same performance you would get if the virtual machine were installed on a raw SCSI partition.

The core dump partition will store information generated if the VMkernel crashes. The core dump information is important in debugging any problems with the VMkernel.


VMware ESX Server Configuration -- Partition/VMFS Editor - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/viewpart
Go Links
Done
Local intranet

VMware ESX Server Configuration

Partition/VMFS Editor

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Editing Partitions and Creating VMFS File Systems

Here you can setup your SCSI/RAID disks to use the VMFS file system where you can store disks for your virtual machines.

If you have any partitions defined, you can convert them to VMFS file systems by changing the type of the partition using fdisk. To format a VMFS file system, click on the "Format VMFS" button next to the VMFS partition you wish to format.

You may also create and delete partitions on your SCSI disk from this page by clicking on the appropriate button below each disk listing.

You can assign a logical name to each VMFS file system that can be used in the config files for your virtual machines. The advantage of using logical names is the SCSI/RAID disks can be moved around without requiring a change to the config files on your machine.

The following disks were detected on your system:

- vmhba0:6:0

Disk vmhba0:6:0

Partition	Type	Start Cyl	End Cyl	Size (MB)	System	VMFS Name	Actions
vmhba0:6:0:1	0xfc	0	6	54	VMware Core Dump		
vmhba0:6:0:2	0xfb	7	2,231	17,453	VMFS	test_cfg_1 Save	

To remove all the partitions, click on the "Delete All Partitions." button.

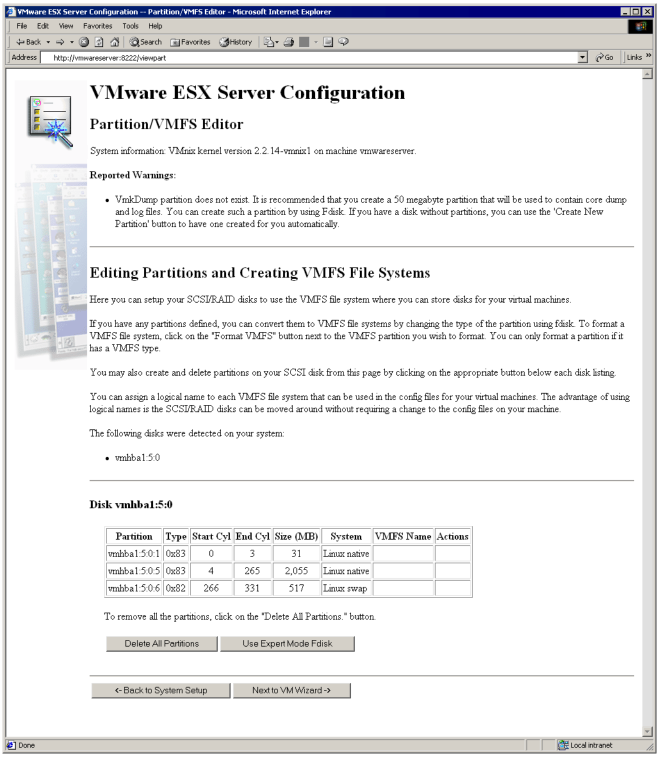
2. You see a screen that reports the sizes of the two partitions you have just created.

   You should assign a logical name to the VMFS partition. Choose a name that will make it easy to identify this particular partition even if you later decide to move the device to a different machine. Enter the logical name in the field under VMFS Name and click Save.

   Click Next to VM Wizard to begin creating a virtual machine

Sharing a SCSI Drive or RAID Array with the Console Operating System

In this example, the disk `vmhba1:5:0` already contains the partitions used by the console operating system. You should not make changes to these partitions. For background on how SCSI devices are identified, see Determining SCSI Target IDs on page 111.


VMware ESX Server Configuration -- Partition/VMFS Editor - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/viewpart
Go Links
Done
Local intranet

VMware ESX Server Configuration

Partition/VMFS Editor

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Reported Warnings:

- VmkDump partition does not exist. It is recommended that you create a 50 megabyte partition that will be used to contain core dump and log files. You can create such a partition by using Fdisk. If you have a disk without partitions, you can use the 'Create New Partition' button to have one created for you automatically.

Editing Partitions and Creating VMFS File Systems

Here you can setup your SCSI/RAID disks to use the VMFS file system where you can store disks for your virtual machines.

If you have any partitions defined, you can convert them to VMFS file systems by changing the type of the partition using fdisk. To format a VMFS file system, click on the "Format VMFS" button next to the VMFS partition you wish to format. You can only format a partition if it has a VMFS type.

You may also create and delete partitions on your SCSI disk from this page by clicking on the appropriate button below each disk listing.

You can assign a logical name to each VMFS file system that can be used in the config files for your virtual machines. The advantage of using logical names is the SCSI/RAID disks can be moved around without requiring a change to the config files on your machine.

The following disks were detected on your system:

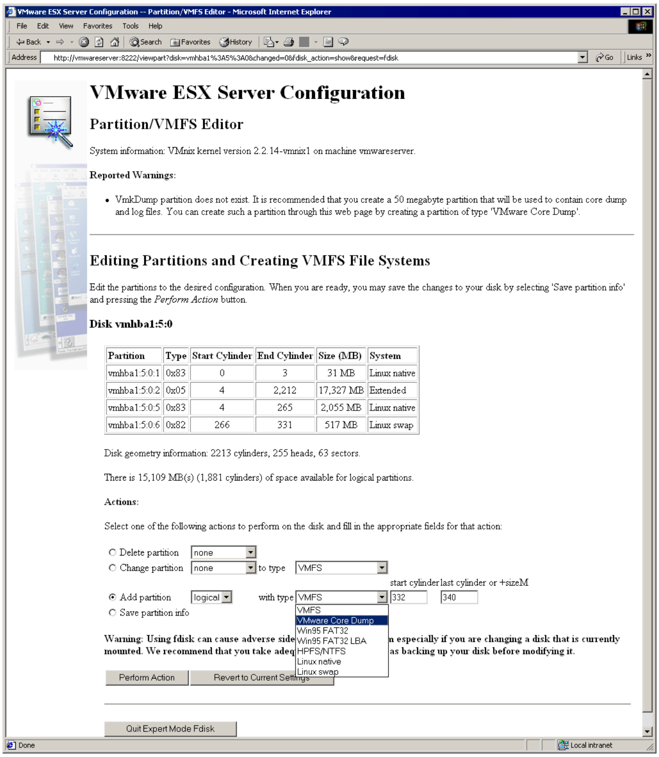
- vmhba1:5:0

Disk vmhba1:5:0

Partition	Type	Start Cyl	End Cyl	Size (MB)	System	VMFS Name	Actions
vmhba1:5:0:1	0x83	0	3	31	Linux native		
vmhba1:5:0:5	0x83	4	265	2,055	Linux native		
vmhba1:5:0:6	0x82	266	331	517	Linux swap		

To remove all the partitions, click on the "Delete All Partitions." button.

1. Click Use Expert Mode Fdisk.



VMware ESX Server Configuration -- Partition/VMFS Editor - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/viewpart?disk=vmhba1%3A5%3A0&changed=0&fdisk_action=show&request=fdisk
Go Links

VMware ESX Server Configuration

Partition/VMFS Editor

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Reported Warnings:

- VmkDump partition does not exist. It is recommended that you create a 50 megabyte partition that will be used to contain core dump and log files. You can create such a partition through this web page by creating a partition of type 'VMware Core Dump'.

Editing Partitions and Creating VMFS File Systems

Edit the partitions to the desired configuration. When you are ready, you may save the changes to your disk by selecting 'Save partition info' and pressing the *Perform Action* button.

Disk vmhba1:5:0

Partition	Type	Start Cylinder	End Cylinder	Size (MB)	System
vmhba1:5:0:1	0x83	0	3	31 MB	Linux native
vmhba1:5:0:2	0x05	4	2,212	17,327 MB	Extended
vmhba1:5:0:5	0x83	4	265	2,055 MB	Linux native
vmhba1:5:0:6	0x82	266	331	517 MB	Linux swap

Disk geometry information: 2213 cylinders, 255 heads, 63 sectors.

There is 15,109 MB(s) (1,881 cylinders) of space available for logical partitions.

Actions:

Select one of the following actions to perform on the disk and fill in the appropriate fields for that action:

○ Delete partition none
○ Change partition none to type VMFS
start cylinder last cylinder or +sizeM
● Add partition logical with type VMFS 332 340
○ Save partition info

Warning: Using fdisk can cause adverse side n especially if you are changing a disk that is currently mounted. We recommend that you take adeq as backing up your disk before modifying it.

2. First add a small core dump partition. The core dump partition will store information generated if the VMkernel crashes. The core dump information is important in debugging any problems with the VMkernel.

   Select Add partition, use the default of logical, and choose VMware Core Dump from the list of file types. In this scenario, an extended partition, which will contain this logical partition, should already exist. If it doesn't, you will need to add an extended partition before you add the logical partition.

   Do not change the number in the From Cylinder field.

   Change the number in the To Cylinder field to `+50M` to set aside about 50MB for the core dump partition. The partition will be at least 50MB, but is likely to be somewhat larger because partitions must begin and end on cylinder boundaries.

3. Click Perform Action.

Note: No changes are actually written to disk until you select Save Partition Info, then click Perform Action.

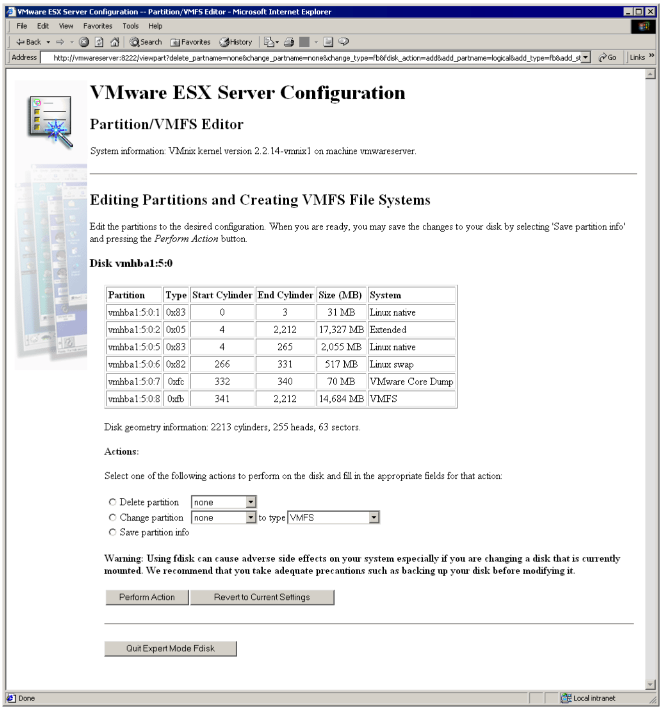
4. Use the rest of the disk or array as a VMFS partition, where you will store virtual machine disk files.

   The VMFS partition provides high-performance access to the virtual machine's files — essentially the same performance you would get if the virtual machine were installed on a raw SCSI partition.

   Select Add partition. You may use the default of logical or change the setting to primary, and choose VMFS from the list of file types. Keep in mind that only four primary partitions can exist on a drive. If you have an extended partition (to contain logical partitions), that counts as one of your four primary partitions.

   Do not change the number in the From Cylinder or To Cylinder fields.

5. Click Perform Action.


VMware ESX Server Configuration -- Partition/VMFS Editor - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/viewpart?delete_partname=none&change_partname=none&change_type=fb&fdisk_action=add&add_partname=logical&add_type=fb&add_st
Go
Links
Done
Local intranet

VMware ESX Server Configuration

Partition/VMFS Editor

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Editing Partitions and Creating VMFS File Systems

Edit the partitions to the desired configuration. When you are ready, you may save the changes to your disk by selecting 'Save partition info' and pressing the *Perform Action* button.

Disk vmhba1:5:0

Partition	Type	Start Cylinder	End Cylinder	Size (MB)	System
vmhba1:5:0:1	0x83	0	3	31 MB	Linux native
vmhba1:5:0:2	0x05	4	2,212	17,327 MB	Extended
vmhba1:5:0:5	0x83	4	265	2,055 MB	Linux native
vmhba1:5:0:6	0x82	266	331	517 MB	Linux swap
vmhba1:5:0:7	0xfc	332	340	70 MB	VMware Core Dump
vmhba1:5:0:8	0xfb	341	2,212	14,684 MB	VMFS

Disk geometry information: 2213 cylinders, 255 heads, 63 sectors.

Actions:

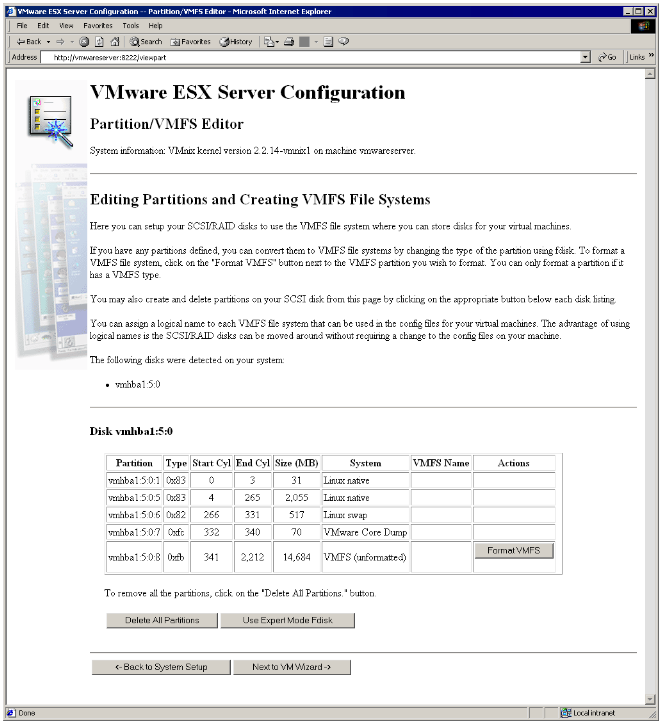
Select one of the following actions to perform on the disk and fill in the appropriate fields for that action:

○ Delete partition none
○ Change partition none to type VMFS
○ Save partition info

Warning: Using fdisk can cause adverse side effects on your system especially if you are changing a disk that is currently mounted. We recommend that you take adequate precautions such as backing up your disk before modifying it.

6. Select Save Partition Info.
7. Click Perform Action.

   Note: At this point, your changes are committed to the disk or array.



VMware ESX Server Configuration -- Partition/VMFS Editor - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/viewpart
Go Links

VMware ESX Server Configuration

Partition/VMFS Editor

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Editing Partitions and Creating VMFS File Systems

Here you can setup your SCSI/RAID disks to use the VMFS file system where you can store disks for your virtual machines.

If you have any partitions defined, you can convert them to VMFS file systems by changing the type of the partition using fdisk. To format a VMFS file system, click on the "Format VMFS" button next to the VMFS partition you wish to format. You can only format a partition if it has a VMFS type.

You may also create and delete partitions on your SCSI disk from this page by clicking on the appropriate button below each disk listing.

You can assign a logical name to each VMFS file system that can be used in the config files for your virtual machines. The advantage of using logical names is the SCSI/RAID disks can be moved around without requiring a change to the config files on your machine.

The following disks were detected on your system:

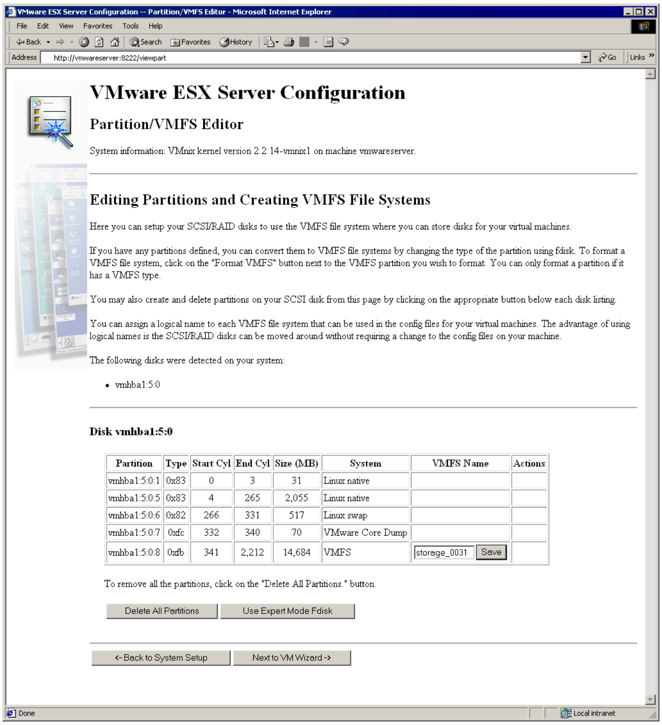
- vmhba1:5:0

Disk vmhba1:5:0

Partition	Type	Start Cyl	End Cyl	Size (MB)	System	VMFS Name	Actions
vmhba1:5:0:1	0x83	0	3	31	Linux native		
vmhba1:5:0:5	0x83	4	265	2,055	Linux native		
vmhba1:5:0:6	0x82	266	331	517	Linux swap		
vmhba1:5:0:7	0xfc	332	340	70	VMware Core Dump		
vmhba1:5:0:8	0xfb	341	2,212	14,684	VMFS (unformatted)		Format VMFS

To remove all the partitions, click on the "Delete All Partitions." button.

8. Locate the table row with information about the VMware partition you just created. Click the Format VMFS button in that row.


VMware ESX Server Configuration -- Partition/VMFS Editor - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
History
Address
http://vmwareserver:8222/viewpart
Go
Links
Done
Local intranet

VMware ESX Server Configuration

Partition/VMFS Editor

System information: VMnix kernel version 2.2.14-vmnix1 on machine vmwareserver.

Editing Partitions and Creating VMFS File Systems

Here you can setup your SCSI/RAID disks to use the VMFS file system where you can store disks for your virtual machines.

If you have any partitions defined, you can convert them to VMFS file systems by changing the type of the partition using fdisk. To format a VMFS file system, click on the "Format VMFS" button next to the VMFS partition you wish to format. You can only format a partition if it has a VMFS type.

You may also create and delete partitions on your SCSI disk from this page by clicking on the appropriate button below each disk listing.

You can assign a logical name to each VMFS file system that can be used in the config files for your virtual machines. The advantage of using logical names is the SCSI/RAID disks can be moved around without requiring a change to the config files on your machine.

The following disks were detected on your system:

- vmhba1:5:0

Disk vmhba1:5:0

Partition	Type	Start Cyl	End Cyl	Size (MB)	System	VMFS Name	Actions
vmhba1:5:0:1	0x83	0	3	31	Linux native		
vmhba1:5:0:5	0x83	4	265	2,055	Linux native		
vmhba1:5:0:6	0x82	266	331	517	Linux swap		
vmhba1:5:0:7	0xfc	332	340	70	VMware Core Dump		
vmhba1:5:0:8	0xfb	341	2,212	14,684	VMFS	storage_0031 Save	

To remove all the partitions, click on the "Delete All Partitions." button.

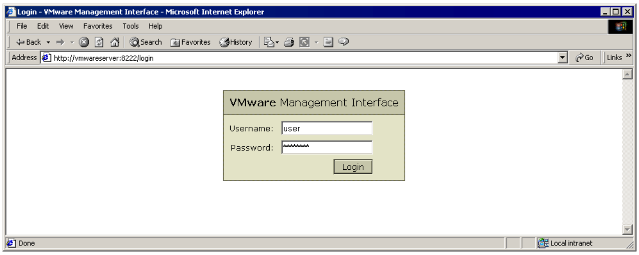
9. You should assign a logical name to the VMFS partition. Choose a name that will make it easy to identify this particular disk even if you later decide to move the device to a different machine. Enter the logical name in the field under VMFS Name and click Save.
10. Click Next to VM Wizard to begin creating a virtual machine.

Creating a New Virtual Machine

The Virtual Machine Wizard will guide you through the basic steps needed to create a virtual machine on your server. Any user who has an account on the server's console operating system may log in to the wizard and create a virtual machine. If you are logged in as root, you may wish to log out at this point, then log in again as a user who will be managing the new virtual machine.

To return to the wizard later, use this URL:

```
http://<hostname>:8222/vmcfg-esx
```



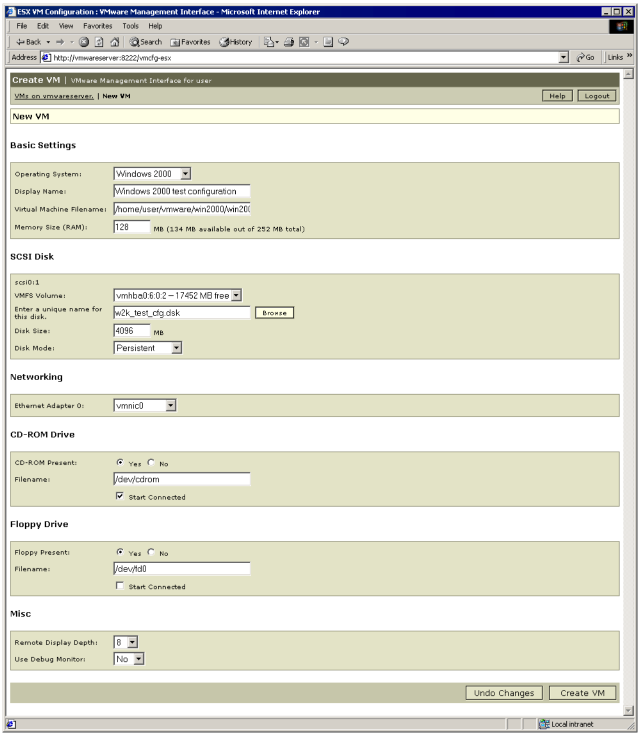
Address: http://vmwareserver:8222/login

VMware Management Interface

Username: user

Password: *********

1. Enter your user name and password, then click Login to begin using the wizard.


ESX VM Configuration : VMware Management Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
History
Address
http://vmwareserver:8222/vmcfg-esx
Go
Links
Help
Logout
Browse
Undo Changes
Create VM
Local intranet

Create VM | VMware Management Interface for user

VMs on vmwareserver. | New VM

New VM

Basic Settings

Operating System: Windows 2000

Display Name: Windows 2000 test configuration

Virtual Machine Filename: /home/user/vmware/win2000/win20(

Memory Size (RAM): 128 MB (134 MB available out of 252 MB total)

SCSI Disk

scsi0:1

VMFS Volume: vmhba0:6:0:2 – 17452 MB free

Enter a unique name for this disk. w2k_test_cfg.dsk

Disk Size: 4096 MB

Disk Mode: Persistent

Networking

Ethernet Adapter 0: vmnic0

CD-ROM Drive

CD-ROM Present: ● Yes ○ No

Filename: /dev/cdrom

☑ Start Connected

Floppy Drive

Floppy Present: ● Yes ○ No

Filename: /dev/fd0

□ Start Connected

Misc

Remote Display Depth: 8

Use Debug Monitor: No

2. Choose the guest operating system for your virtual machine. Corresponding default entries will appear for other configuration settings.
3. Make any changes you wish to the default settings.

   Under Basic Settings, the name you enter in the Display Name field is the name that will be listed in the Web-based management interface. Be sure to enter a name that will allow you to distinguish this virtual machine from others you have created or plan to create.

   In the Basic Settings section, be sure that the entry in the Virtual Machine Filename field is unique. The default path and file name are based on the guest operating system you have chosen. If other virtual machines have been created on this server, you will need to change the path to create a new, unique directory for the new virtual machine.

The default Memory Size setting depends on the guest operating system you have selected. You may need to change it to meet the demands of applications you plan to run in the virtual machine. You may change this setting later, using the Configure VM page of the Web-based management interface.

For background on allocating memory to virtual machines, see How the System Uses Memory on page 114.

In the SCSI Disk section, be sure that the virtual machine's file name is unique. The file name should end in `.dsk`.

Click the Browse button in the SCSI Disk section if you want to view file names already in use. If you use an existing file name, ESX Server will warn you that a virtual machine with that name already exists.

Select the disk mode for your virtual disk. ESX Server can use disks in four different modes: persistent, nonpersistent, undoable and append. **Persistent** disks behave exactly like conventional disk drives on a computer. All writes to a persistent disk are written out permanently to the disk as soon as the guest operating system writes the data. All changes to a **nonpersistent** mode disk are discarded when a virtual machine session is powered down. When you use **undoable** mode, you have the option later of keeping or discarding changes you have made during a working session. Until you decide, the changes are saved in a redo-log file. **Append** mode also stores changes in a redo log. It continually adds changes to the redo log until you remove the redo-log file or commit the changes using the `commit` command in `vmkfstools` (see page 106).

The setup process allows you to create one virtual disk for your virtual machine. You can add more virtual disks later, using the Configure VM page of the Web-based management interface.

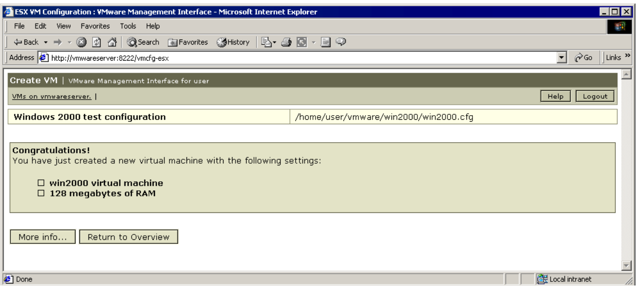
Similarly, after the virtual machine is created, you can use the Confgure VM page to assign additional network adapters to the virtual machine.

If you need help determining which network adapter is associated with a particular device name, you can use the console operating system's `findnic` command (see page 125).

If you want, you can change the color depth of your display using the the Remote Display Depth setting. A higher color depth setting slows down screen redraws and increases network load when you use a remote console to view a virtual machine across a network connection. However, with greater color depth, you get better color resolution and fidelity. The default setting is 8. Other options are 15, 16 or 24.

If you want to reset the entries to the defaults, click Undo Changes.

When you are satisfied with the settings, click Create VM.


ESX VM Configuration : VMware Management Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/vmcfg-esx
Go Links
Create VM | VMware Management Interface for user
VMs on vmwareserver. |
Help
Logout
More info...
Return to Overview
Done
Local intranet

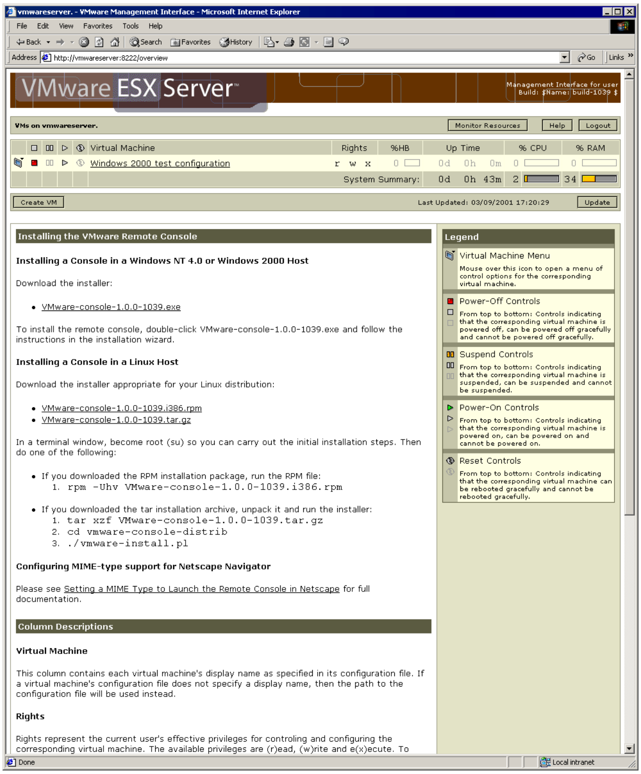
Windows 2000 test configuration /home/user/vmware/win2000/win2000.cfg

Congratulations!
You have just created a new virtual machine with the following settings:

- □ **win2000 virtual machine**
- □ **128 megabytes of RAM**

4. The confirmation page includes information on some basic configuration settings for your new virtual machine.

   To go to the main page of the Web-based management interface, click Return to Overview.


vmwareserver. - VMware Management Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/overview
Go Links
VMware ESX Server™
Management Interface for user
Build: $Name: build-1039 $
Monitor Resources
Help
Logout
Create VM
Update
Done
Local intranet

VMs on vmwareserver.

	□	▯▯	▷	⟳	Virtual Machine	Rights	%HB	Up Time	% CPU	% RAM
	🟥	▯▯	▷	⟳	Windows 2000 test configuration	r w x	0	0d 0h 0m	0	0
						System Summary:		0d 0h 43m	2	34

Last Updated: 03/09/2001 17:20:29

Installing the VMware Remote Console

Installing a Console in a Windows NT 4.0 or Windows 2000 Host

Download the installer:

- VMware-console-1.0.0-1039.exe

To install the remote console, double-click VMware-console-1.0.0-1039.exe and follow the instructions in the installation wizard.

Installing a Console in a Linux Host

Download the installer appropriate for your Linux distribution:

- VMware-console-1.0.0-1039.i386.rpm
- VMware-console-1.0.0-1039.tar.gz

In a terminal window, become root (su) so you can carry out the initial installation steps. Then do one of the following:

- If you downloaded the RPM installation package, run the RPM file:
  1. `rpm -Uhv VMware-console-1.0.0-1039.i386.rpm`
- If you downloaded the tar installation archive, unpack it and run the installer:
  1. `tar xzf VMware-console-1.0.0-1039.tar.gz`
  2. `cd vmware-console-distrib`
  3. `./vmware-install.pl`

Configuring MIME-type support for Netscape Navigator

Please see Setting a MIME Type to Launch the Remote Console in Netscape for full documentation.

Column Descriptions

Virtual Machine

This column contains each virtual machine's display name as specified in its configuration file. If a virtual machine's configuration file does not specify a display name, then the path to the configuration file will be used instead.

Rights

Rights represent the current user's effective privileges for controling and configuring the corresponding virtual machine. The available privileges are (r)ead, (w)rite and e(x)ecute. To

Legend

Virtual Machine Menu
Mouse over this icon to open a menu of control options for the corresponding virtual machine.

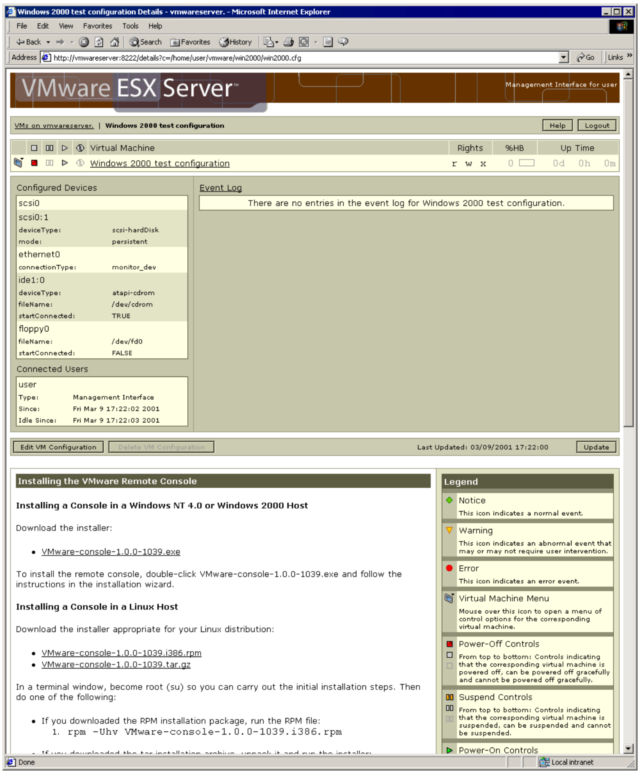
Power-Off Controls
From top to bottom: Controls indicating that the corresponding virtual machine is powered off, can be powered off gracefully and cannot be powered off gracefully.

Suspend Controls
From top to bottom: Controls indicating that the corresponding virtual machine is suspended, can be suspended and cannot be suspended.

Power-On Controls
From top to bottom: Controls indicating that the corresponding virtual machine is powered on, can be powered on and cannot be powered on.

Reset Controls
From top to bottom: Controls indicating that the corresponding virtual machine can be rebooted gracefully and cannot be rebooted gracefully.

5. To see additional details about a virtual machine, click the virtual machine’s name.


Windows 2000 test configuration Details - vmwareserver. - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/details?c=/home/user/vmware/win2000/win2000.cfg
Go Links
VMware ESX Server™
Management Interface for user
VMs on vmwareserver. | Windows 2000 test configuration
Help
Logout
Edit VM Configuration
Delete VM Configuration
Update
Done
Local intranet

	□	▯▯	▷	Ⓢ	Virtual Machine	Rights	%HB	Up Time
	■	▯▯	▷	Ⓢ	Windows 2000 test configuration	r w x	0	0d 0h 0m

Configured Devices

scsi0

scsi0:1

deviceType:	scsi-hardDisk
mode:	persistent

ethernet0

connectionType:	monitor_dev

ide1:0

deviceType:	atapi-cdrom
fileName:	/dev/cdrom
startConnected:	TRUE

floppy0

fileName:	/dev/fd0
startConnected:	FALSE

Connected Users

user

Type:	Management Interface
Since:	Fri Mar 9 17:22:02 2001
Idle Since:	Fri Mar 9 17:22:03 2001

Event Log

There are no entries in the event log for Windows 2000 test configuration.

Last Updated: 03/09/2001 17:22:00

Installing the VMware Remote Console

Installing a Console in a Windows NT 4.0 or Windows 2000 Host

Download the installer:

- VMware-console-1.0.0-1039.exe

To install the remote console, double-click VMware-console-1.0.0-1039.exe and follow the instructions in the installation wizard.

Installing a Console in a Linux Host

Download the installer appropriate for your Linux distribution:

- VMware-console-1.0.0-1039.i386.rpm
- VMware-console-1.0.0-1039.tar.gz

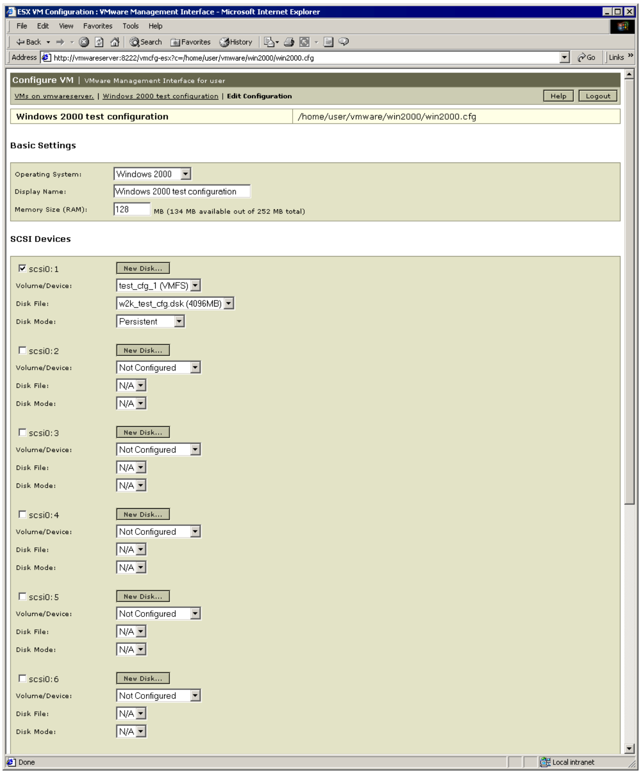
In a terminal window, become root (su) so you can carry out the initial installation steps. Then do one of the following:

- If you downloaded the RPM installation package, run the RPM file:
  1. `rpm -Uhv VMware-console-1.0.0-1039.i386.rpm`
- If you downloaded the tar installation archive, unpack it and run the installer:

Legend

◆	Notice This icon indicates a normal event.
▽	Warning This icon indicates an abnormal event that may or may not require user intervention.
●	Error This icon indicates an error event.
	Virtual Machine Menu Mouse over this icon to open a menu of control options for the corresponding virtual machine.
■ □ □	Power-Off Controls From top to bottom: Controls indicating that the corresponding virtual machine is powered off, can be powered off gracefully and cannot be powered off gracefully.
▯▯ ▯▯ ▯▯	Suspend Controls From top to bottom: Controls indicating that the corresponding virtual machine is suspended, can be suspended and cannot be suspended.
▷	Power-On Controls

6. Click Edit VM Configuration if you want to change settings for your virtual machine.


ESX VM Configuration : VMware Management Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go Links
Help
Logout
New Disk...
Done
Local intranet

http://vmwareserver:8222/vmcfg-esx?c=/home/user/vmware/win2000/win2000.cfg

Configure VM | VMware Management Interface for user

VMs on vmwareserver. | Windows 2000 test configuration | Edit Configuration

Windows 2000 test configuration /home/user/vmware/win2000/win2000.cfg

Basic Settings

Operating System: Windows 2000

Display Name: Windows 2000 test configuration

Memory Size (RAM): 128 MB (134 MB available out of 252 MB total)

SCSI Devices

☑ scsi0:1

Volume/Device: test_cfg_1 (VMFS)

Disk File: w2k_test_cfg.dsk (4096MB)

Disk Mode: Persistent

□ scsi0:2

Volume/Device: Not Configured

Disk File: N/A

Disk Mode: N/A

□ scsi0:3

Volume/Device: Not Configured

Disk File: N/A

Disk Mode: N/A

□ scsi0:4

Volume/Device: Not Configured

Disk File: N/A

Disk Mode: N/A

□ scsi0:5

Volume/Device: Not Configured

Disk File: N/A

Disk Mode: N/A

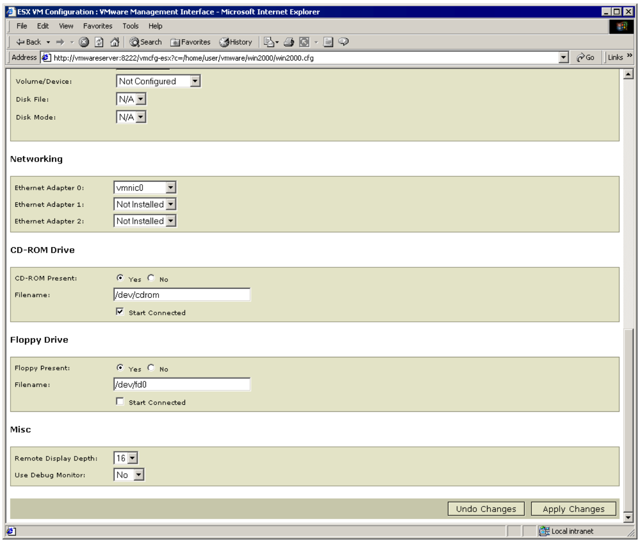
□ scsi0:6

Volume/Device: Not Configured

Disk File: N/A

Disk Mode: N/A

Top section of page for editing a virtual machine configuration


ESX VM Configuration : VMware Management Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go
Links
Local intranet

http://vmwareserver:8222/vmcfg-esx?c=/home/user/vmware/win2000/win2000.cfg

Volume/Device: Not Configured
Disk File: N/A
Disk Mode: N/A

Networking

Ethernet Adapter 0: vmnic0
Ethernet Adapter 1: Not Installed
Ethernet Adapter 2: Not Installed

CD-ROM Drive

CD-ROM Present: ● Yes ○ No
Filename: /dev/cdrom
☑ Start Connected

Floppy Drive

Floppy Present: ● Yes ○ No
Filename: /dev/fd0
□ Start Connected

Misc

Remote Display Depth: 16
Use Debug Monitor: No

Bottom section of page for editing a virtual machine configuration

When you are finished, click Apply Changes.

Your new virtual machine is like a new computer with a blank hard disk. You must install a guest operating system before you can use the virtual machine.

Installing a Guest Operating System and VMware Tools

You will probably configure your virtual machine with a blank (unformatted) SCSI virtual disk. You can install an operating system on this virtual disk just as you would on a new physical machine, using a standard installation CD-ROM, and formatting the virtual disk at the appropriate place in the installation process. See the technical notes on page 82 for details on installing specific guest operating systems. Or you may start with a virtual disk created with VMware Workstation 2.x or VMware GSX Server, then configure the guest operating system to work with VMware ESX Server.

Once your guest operating system is installed, be sure to follow the directions below for installing VMware Tools and the network driver.

Installing a Guest Operating System in a Virtual Machine

To install a guest operating system and other software, you should work on a separate workstation and use the VMware remote console. It is best *not* to run X on the server's console operating system.

For details on installing the remote console, see Installing the Remote Console Software on page 51. Follow the directions on that page for starting a remote console on your Windows or Linux workstation and connecting to a virtual machine.

Insert the installation CD-ROM for your guest operating system in the server's CD-ROM drive. Click the Power On button to begin setting up your guest operating system.

Note: When you are installing a guest operating system on a new virtual disk, you may see a message warning you that the disk is corrupted and asking if you want to place a partition table on the disk. This does not mean there is any problem with your physical hard disk. It simply means some data needs to be written to the file that holds your virtual hard disk. All you need to do is respond yes. You will also need to partition and format the virtual disk as you would with a new, blank hard drive.

Migrating VMware Workstation 2.x and VMware GSX Server Virtual Machines

You can modify virtual machines created with VMware Workstation 2.0 or higher or VMware GSX Server to run on VMware ESX Server.

The virtual machine you want to migrate must be set up on a SCSI disk — either a virtual disk or a raw disk. You will migrate it to run from a virtual SCSI disk under ESX Server.

Be sure you have enough space on the VMFS disk where you store virtual machines to hold the full size of the source virtual disk. In ESX Server the disk's full size is allocated at the time the virtual disk file is created. In VMware Workstation and GSX Server, the virtual disk file starts smaller and grows to the maximum size as data is added. Thus, a virtual disk defined as a 2GB disk may be contained in a 500MB file. When you migrate the virtual disk to ESX Server, it will occupy 2GB of disk space.

Note: When you install VMware Tools in the VMware ESX Server virtual machine, you set up a new network driver. This driver is not suitable for a virtual machine running under VMware Workstation 2.x or VMware GSX Server. If you think you may want to use this virtual machine under VMware Workstation or VMware GSX Server at a later time, you may find it most convenient to make a copy of the virtual machine before you migrate it. Otherwise, if you move it back to run under the original VMware product, you will need to reconfigure it to use the original network driver.

Follow these steps to migrate a virtual machine to VMware ESX Server:

1. Create a new configuration (`.cfg`) file in the directory where you are storing files associated with the new virtual machine — for example `/vm/vm1`. The simplest approach is to set up a new virtual machine as described in Creating a New Virtual Machine on page 37.

   Note the name and location of the virtual disk (`.dsk`) file created for the new virtual machine.

2. Working from the console operating system, use `vmkfstools` to remove the `.dsk` file of the new virtual machine. You must log in as root to use `vmkfstools`.

   ```
   vmkfstools -r vmhba<x>:<y>:<z>:<filename>.dsk
   ```

   For background on how SCSI devices are identified, see Determining SCSI Target IDs on page 111.

3. Be sure you have access to the files in the directory (in the case of a virtual disk) or partition (in the case of a raw disk) that holds the source virtual machine. You may be able to mount the source location, or you may prefer to copy the files to a temporary directory on the console operating system.

   If you are not sure where the source files are, open the virtual machine in the VMware product you used to create it, open the Configuration Editor (Settings >

Configuration Editor), expand the SCSI Drives tree, and click the name of the drive you want to migrate. The Name field provides the location information.

4. Use `vmkfstools` to import the source virtual machine to the target location. For `<targetfile>.dsk`, use the name of the `.dsk` file that you removed in step 2.

   ```
   vmkfstools -i <sourcepath>/<sourcefile>.dsk \
   vmhba<x>:<y>:<z>:<targetfile>.dsk
   ```

   Note: The backslash at the end of the first line above indicates that there should be no line break. Enter the whole command on one line. Do not type the backslash.

5. Boot your virtual machine using VMware ESX Server and follow the instructions below for installing VMware Tools and the network driver in the virtual machine.

Note: In most cases, the most convenient way to create the VMFS file system is through the Web-based configuration wizard. However, you can also create a VMFS file system on the partition or disk using the `vmkfstools` command. See the `vmkfstools(1)` man page for details.

Installing VMware Tools and Network Driver in the Guest Operating System

The steps below assume that you are working from a remote console connected to your virtual machine.

Prepare your virtual machine to install VMware Tools. Choose Settings > VMware Tools Install. This option prepares the floppy drive in the virtual machine to use a floppy image containing the VMware Tools packages. This image, which appears as a regular floppy disk in the virtual machine, was placed on your machine when you installed VMware ESX Server. Follow the directions below to install VMware Tools in your particular guest operating system.

Windows 2000 Guest

1. After choosing Settings > VMware Tools Install, open the Windows Control Panel (Start > Settings > Control Panel) and double-click Add/Remove Hardware.
2. In the Add/Remove Hardware Wizard, select Add/Troubleshoot a Device. Windows will search for Plug and Play devices.
3. From the long list of hardware devices, select Ethernet Controller and click Next. You should get a message that the drivers for this device are not installed. Click Finish to continue.

4. Click Next on the Upgrade Device Wizard screen. Select “Search for a suitable driver for my hardware device,” and instruct Windows to search the floppy drive. Windows should find `a:\vmnet\win2k\oemsetup.inf`. Click Next and Yes to complete the installation of the VMware network driver.

5. Run `VMwareTools.exe` from the floppy (Start > Run > `a:\VMwareTools.exe`) to complete the installation of VMware Tools.

Windows NT 4.0 Guest

1. After selecting Settings > VMware Tools Install, go to Start > Control Panel > Network > Adapters and click Add.

2. Click Have Disk and enter `a:\vmnet\winnt` in the Insert Disk dialog. Click OK when VMware Virtual Ethernet Adapter is displayed in the Select OEM Option dialog. The VMware network driver will be installed.

3. Click Close in the Adapters dialog to complete the installation. Windows will let you configure the Internet address for the card.

   If you are installing on a virtual machine that was created with VMware Workstation 2.x and used networking, you will need to use an address different from the one the original network configuration used (since that address is still assigned to the now non-existent virtual AMD card). Or you can change the address assigned to the AMD card at this point.

   Note: The VMware Virtual Ethernet Adapter driver will run correctly only if you have Service Pack 3 or later installed. If you do not have the proper service pack installed yet, you may get an error message such as: “System Process — Driver Entry Point Not Found; The `\SystemRoot\System32\drivers\vmxnet.sys` device driver could not locate the entry point NdisGetFirstBufferFromPacket in driver NDIS.SYS.” However, even if you get this message, the driver should work if you subsequently install the correct service pack.

4. Windows will ask you to reboot. Click No.

5. Run `VMwareTools.exe` from the floppy (Start > Run > `a:\VMwareTools.exe`) to complete the installation of VMware Tools.

6. Reboot the virtual machine when you finish.

Linux guest

1. In a Linux guest, become root, mount the VMware Tools floppy, copy the contents of the floppy to `/tmp`, then unmount the floppy.

```
su
cd /
mount -t vfat /dev/fd0 /mnt
cp /mnt/* /tmp
umount /dev/fd0
```

2. Untar the VMware Tools tar file in /tmp and install it.

   ```
   cd /tmp
   tar zxf vmware-linux-tools.tar.gz
   cd vmware-linux-tools
   ./install.pl
   ```

   Note: If you have a Red Hat Linux 7.0 guest operating system and have installed the kernel source files, you will get some compiler error messages during the VMware Tools installation, because of a bug in the Red Hat 7.0 kernel headers. However, the toolbox and drivers will still work correctly.

3. Test to be sure that the vmxnet driver is installed correctly.

   ```
   insmod vmxnet
   ```

4. If the driver is installed correctly, you will see some informative output but no error messages. In addition, you should now have an entry such as `alias eth0 vmxnet` in the file `/etc/conf.modules` (or `/etc/modules.conf` in Red Hat Linux 7.0).

5. If you wish, start X and your graphical environment and launch the VMware Tools background application.

   ```
   vmware-toolbox &
   ```

Preparing to Use the Remote Management Software

You can manage VMware ESX Server from a remote workstation using the VMware remote console and the Web-based management interface.

Remote console software is available for Windows and Linux workstations. The remote console lets you attach directly to a virtual machine. You can start and stop programs, change the configuration of the guest operating system, and do other tasks as if you were working at a physical computer.

The Web-based management interface can be used from any workstation with a compatible browser — Internet Explorer 5.0 or higher or Netscape 4.5 or higher. It gives you a bird's-eye view of all the registered virtual machines on a server and allows you to stop, start, suspend, resume and reset a virtual machine.

Note: If you need secure communications between your management workstations and the server, you will need to install appropriate software to provide that security. For additional details, see the Network Security section in the technical note Authentication and Security Features in VMware ESX Server on page 99.

Registering Your Virtual Machines

If you create your virtual machines using the Virtual Machine Configuration Wizard, they are automatically registered in the file `/etc/vmware/vm-list` on the server's console operating system. The remote management software checks this file for pointers to the virtual machines you want to manage.

If you want to manage virtual machines that you set up in some other way, without using the wizard, you must first register them. To do so, use this command:

```
vmware-control -s register /<configpath>/<configfile>.cfg
```

To remove a virtual machine from the list, use this command:

```
vmware-control -s unregister \
/<configpath>/<configfile>.cfg
```

Note: Type the whole command on one line. Do not type the backslash.

Installing the Remote Console Software

Use the package that corresponds to the operating system running on your management workstation, and follow the installation steps below.

Installer files are available on the distribution CD-ROM. You may also download the appropriate installer from the Overview page of the Web-based management interface.

Windows NT 4.0 or Windows 2000

1. Find the installer file — `VMware-console-1.v.v-xxxx.exe` — on the distribution CD or in the directory where you downloaded it.
2. Double-click `VMware-console-1.v.v-xxxx.exe` to start the installation wizard.
3. Follow the on-screen instructions.

Linux — RPM Installer

1. Find the installer file — `VMware-console-1.v.v-xxxx.i386.rpm` — on the distribution CD or in the directory where you downloaded it and change to that directory.
2. Run the RPM installer.

   ```
   rpm -Uhv VMware-console-1.v.v-xxxx.i386.rpm
   ```

Linux — Tar Installer

1. Find the installer file — `VMware-console-1.v.v-xxxx.tar.gz` — on the distribution CD or in the directory where you downloaded it and copy it to the `/tmp` directory or another directory of your choice.
2. Unpack the tar archive.

   ```
   tar zxf VMware-console-1.v.v-xxxx.tar.gz
   ```

3. Change to the directory where the archive was unpacked.

   ```
   cd vmware-console-distrib
   ```

4. Run the installer.

   ```
   ./vmware-install.pl
   ```

For information on running virtual machines from the remote console, see Using the Remote Console on page 60.

3

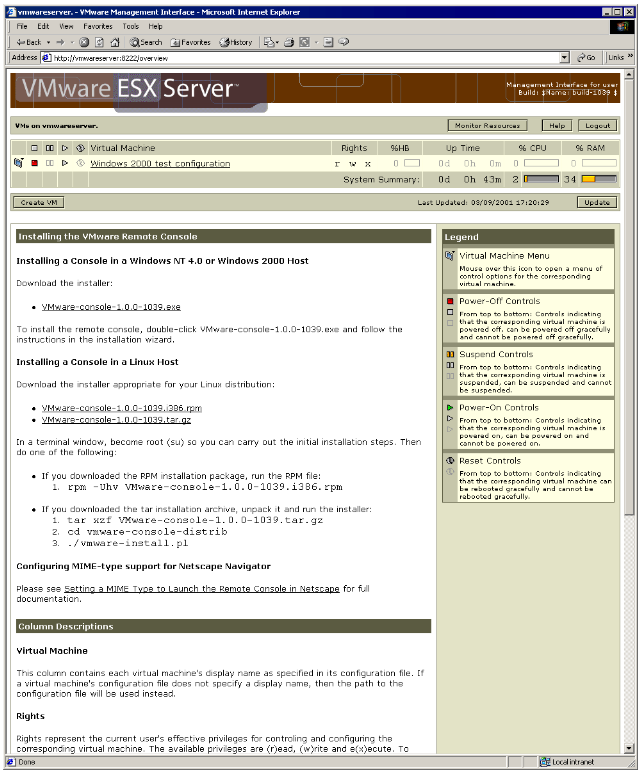
Running ESX Server

Using the Web-Based Management Interface

To use the Web-based interface, you should be running Internet Explorer 4.0 or higher or Netscape Navigator 4.5 or higher. If you are using Netscape Navigator, check the advanced preferences (Edit > Preference > Advanced) to be sure JavaScript and style sheets are both enabled. You will need to know the host name or IP address of the server you want to monitor.

The URL to connect to the server is

```
http://<hostname>:8222/overview/
```



Address: http://vmwareserver:8222/overview

Management Interface for user
Build: $Name: build-1039 $

VMs on vmwareserver.

	□	▯▯	▷	⟳	Virtual Machine	Rights	%HB	Up Time	% CPU	% RAM
	■	▯▯	▷	⟳	Windows 2000 test configuration	r w x	0	0d 0h 0m	0	0
					System Summary:			0d 0h 43m	2	34

Last Updated: 03/09/2001 17:20:29

Installing the VMware Remote Console

Installing a Console in a Windows NT 4.0 or Windows 2000 Host

Download the installer:

- VMware-console-1.0.0-1039.exe

To install the remote console, double-click VMware-console-1.0.0-1039.exe and follow the instructions in the installation wizard.

Installing a Console in a Linux Host

Download the installer appropriate for your Linux distribution:

- VMware-console-1.0.0-1039.i386.rpm
- VMware-console-1.0.0-1039.tar.gz

In a terminal window, become root (su) so you can carry out the initial installation steps. Then do one of the following:

- If you downloaded the RPM installation package, run the RPM file:
  1. `rpm -Uhv VMware-console-1.0.0-1039.i386.rpm`
- If you downloaded the tar installation archive, unpack it and run the installer:
  1. `tar xzf VMware-console-1.0.0-1039.tar.gz`
  2. `cd vmware-console-distrib`
  3. `./vmware-install.pl`

Configuring MIME-type support for Netscape Navigator

Please see Setting a MIME Type to Launch the Remote Console in Netscape for full documentation.

Column Descriptions

Virtual Machine

This column contains each virtual machine's display name as specified in its configuration file. If a virtual machine's configuration file does not specify a display name, then the path to the configuration file will be used instead.

Rights

Rights represent the current user's effective privileges for controling and configuring the corresponding virtual machine. The available privileges are (r)ead, (w)rite and e(x)ecute. To

Legend

Virtual Machine Menu
Mouse over this icon to open a menu of control options for the corresponding virtual machine.

Power-Off Controls
From top to bottom: Controls indicating that the corresponding virtual machine is powered off, can be powered off gracefully and cannot be powered off gracefully.

Suspend Controls
From top to bottom: Controls indicating that the corresponding virtual machine is suspended, can be suspended and cannot be suspended.

Power-On Controls
From top to bottom: Controls indicating that the corresponding virtual machine is powered on, can be powered on and cannot be powered on.

Reset Controls
From top to bottom: Controls indicating that the corresponding virtual machine can be rebooted gracefully and cannot be rebooted gracefully.

The information and controls in the Web-based management interface are arranged in columns containing symbols, some of which are similar to those on a CD player, and text. These symbols and icons appear on the Overview, Details and Event Log pages.

Item	Description
🖥	Hold the mouse over the icon to display a menu of options for the virtual machine, or click to launch a remote console. Netscape users must define a MIME type for the console first; Internet Explorer is automatically configured when the remote console is installed. The menu includes the following commands. Depending on your permissions and the state of the virtual machine, some options may not be available. **Launch Remote Console** – launches the remote console. This is the same as clicking 🖥. **View Details** – opens the Details page for this virtual machine. This is the same as clicking the display name link in the Virtual Machine column. **View Event Log** – opens the Event Log page for this virtual machine. This is the same as clicking the Event Log link on the Overview page. **Power-Off** – gracefully powers off the guest operating system and the virtual machine. This is the same as clicking 🟥. **Suspend** – suspends a running virtual machine or resumes a suspended virtual machine. This is the same as clicking ⏸. **Power-On** – powers on a stopped virtual machine or resumes a suspended virtual machine. This is the same as clicking ▶. **Reset** – gracefully resets the guest operating system and the virtual machine. This is the same as clicking ↻. **Force Power-Off** – shuts down the virtual machine immediately. This is the same as turning off the power to a physical computer. **Force Reset** – resets the virtual machine immediately. This is the same as pressing the reset button on a physical computer.
🟥	Click to gracefully power off the virtual machine. ESX Server closes any open applications and shuts down the guest operating system before powering off the virtual machine. When this icon is red, the virtual machine has been powered off.
⏸	Click to suspend a running virtual machine or resume a suspended virtual machine. When this icon is orange, the virtual machine has been suspended.
▶	Click to power on a stopped virtual machine or to resume a suspended virtual machine. When this icon is green, the virtual machine is running.

Item	Description
↻	Click to gracefully reset a running virtual machine. ESX Server closes any open applications and shuts down the guest operating system before resetting the virtual machine.
Virtual Machine	The path to the configuration file for the virtual machine; if a display name for the virtual machine is specified in the configuration file, then that name appears here instead. Click the link for more details about the virtual machine.
Rights	Rights represent the permissions you have for each configuration file on the host machine. The available permissions are **r**ead, **w**rite and e**x**ecute.
% HB	% HB is the average percentage of heartbeats received by a virtual machine during the minute prior to the last page upddate. Heavily loaded guest operating systems may not send 100% of the expected heartbeats, even though the system is otherwise operating normally; in general, only when the heartbeat percentage drops to zero should the virtual machine or guest operating system be considered unhealthy. Note that if VMware Tools is not installed or is not running, the guest operating system will not send any heartbeats to its virtual machine and this meter will be disabled.
Up Time	The length of time the virtual machine has been running in days, hours, minutes and seconds.
% CPU	The average percentage of host operating system processor capacity the virtual machine used during the final minute before the page was last updated. **Note:** This column appears on the Overview page only.
% RAM	The average percentage of host operating system memory the virtual machine used during the final minute before the page was last updated. **Note:** This column appears on the Overview page only.
System Summary	The total up time for the host system, as well as processor consumption and memory usage for **all** processes running on your host system.

In addition, the following buttons appear on most or all of the pages in the management interface.

Update – This button refreshes or reloads the current page. To avoid conflicts with other users, click this button before you perform an operation like shutting down, suspending, resuming or starting a virtual machine. The Update button does not appear on the New VM page.

Logout – This button logs you out of the management interface. Click Logout to return to the Login page.

Help – This button connects you to the main page for ESX Server online documentation.

Edit VM Configuration – This button appears on a virtual machine's Details page. It takes you to the Configure VM page, where you can change many of a virtual machine's configuration settings. This button is active only when the virtual machine is halted.

When a virtual machine is running, the Overview page displays its ID number in parentheses after the machine's name.

Setting the MIME Type in Netscape Navigator

If you are using Netscape Navigator and want to launch a remote console from the Web-based management interface, as described above, you must first set a MIME type for the remote console program.

Windows

In Netscape Navigator on Windows, follow these steps to set the MIME type.

1. Use the browser to connect to the server you want to manage.
2. Click the terminal icon for the virtual machine you want to view in a remote console.
3. A dialog will ask what you want to do with the file.

   Click Pick App
4. Another dialog will let you enter the path to the application or browse to it.

   Fill in the path or browse to the remote console program.

   The default path is `C:\Program Files\VMware\Programs\Console\vmxRemote.exe`.
5. Your browser is now set to launch the remote console when you click the terminal icon in the future.

Linux

In Netscape Navigator on Linux, follow these steps to set the MIME type.

1. Select Edit > Preferences...
2. Expand Navigator.
3. Highlight Applications.
4. Click New.

   An input dialog is displayed.

5. Fill in the Description field with VMware remote console.
6. Fill in MIME Type with `application/x-vmware-console`.
7. Leave Suffixes blank.
8. Click the Application diamond.
9. Fill in Application with the path to the remote console program or click Choose to navigate to the program on your computer. The default path is `/usr/bin/vmware-console -o %s`.
10. Click OK to close the input dialog.
11. Click OK to close the preferences dialog.

Using Disk Modes

You can use the Configure VM page of the Web-based management interface to change the disk mode for the disks used by your virtual machine.

1. Connect to the server that hosts the virtual machine as a user who has rights to administer the virtual machine. The virtual machine should be powered off.
2. Click the link under the name of the virtual machine you want to modify.
3. Click Edit VM Configuration.
4. Find the listing for the drive you want to change.
5. Choose the appropriate option for Persistent, Nonpersistent, Undoable or Append disk mode from the drop-down list, then click Apply Changes.

ESX Server can use disks in four different modes: persistent, nonpersistent, undoable and append.

Persistent disks behave exactly likeconventional disk drives on a computer. All writes to a persistent disk are written out permanently to the disk as soon as the guest operating system writes the data.

All changes to a **nonpersistent** mode disk are discarded after that ESX Server session is powered down.

When you use **undoable** mode, you have the option later of keeping or discarding changes you have made during a working session. Until you decide, the changes are saved in a redo-log file.

ESX Server supports an additional **append** mode for virtual disks stored as VMFS files. Like undoable mode, append mode maintains a redo log. However, in this mode, no dialog appears when the virtual machine is powered off to ask whether you want to commit changes. All changes are continually appended to the redo log. At any point, the changes can be undone by removing the redo log. You should shut down the guest operating system and power off the virtual machine before deleting that virtual machine's redo log.

Using the Remote Console

The remote console gives you a direct window into an individual virtual machine running under VMware ESX Server. Remote console software is available for Windows NT, Windows 2000 and Linux management workstations. For instructions on installing the software, see Installing the Remote Console Software on page 51.

Note: You must use the version of the remote console supplied with ESX Server in order to connect to an ESX Server virtual machine. At this time, it is not possible to connect to an ESX Server virtual machine using the remote console supplied with VMware GSX Server, and the two versions cannot be installed on the same computer.

If you need to manage ESX Server and GSX Server virtual machines from the same computer, there is a workaround. You can run VMware Workstation on your management workstation and run one version of the remote console in a VMware Workstation virtual machine. Install the more often used remote console software on your physical computer and the less often used remote console in the virtual machine.

Starting the Remote Console on Windows

1. Start the remote console program.

   Start > Programs > VMware > VMware Remote Console

2. A dialog asks for the information needed to connect you to the virtual machine. Fill in the blanks with

   - The host name (or IP address)
   - The port (keep the default — 902 — unless you are using a mapped port on a secure connection)
   - The path to the virtual machine's configuration file on the server
   - Your user name
   - Your password

Starting the Remote Console on Linux

1. Start the remote console program.

   ```
   vmware-console
   ```

2. A dialog asks for the information needed to connect you to the virtual machine. Fill in the blanks with

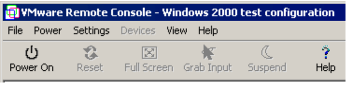
   - The host name (or IP address)

- The port (keep the default — 902 — unless you are using a mapped port on a secure connection)
- The path to the virtual machine's configuration file on the server
- Your user name
- Your password

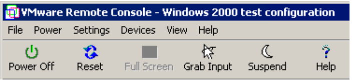
Running a Virtual Machine Using the Remote Console

When you view your virtual machine through a remote console, it behaves much like a separate computer that runs in a window on your computer's desktop.

Instead of using physical buttons to turn this computer on and off, you use buttons at the top of the VMware console window.

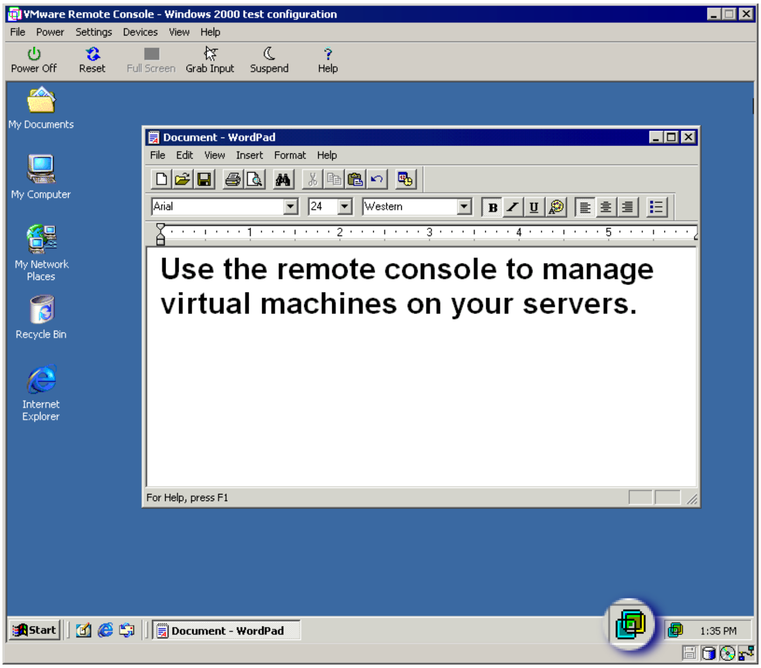


This virtual machine is powered off



This virtual machine is powered on

The power button is labeled Power On or Power Off, depending on whether your virtual computer is running or not.



Use the remote console to manage virtual machines on your servers.

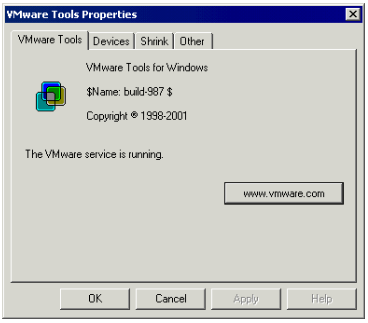
When VMware Tools for Windows is running, the VMware Tools icon appears in the system tray

VMware Tools Settings

The following description of the settings for VMware Tools is based on a Windows 2000 guest operating system. Similar configuration options are available in VMware Tools for other guest operating systems.

1. To open the VMware Tools control panel, double-click the VMware Tools icon in the virtual machine’s system tray. The VMware Tools Properties dialog appears.

2. On the VMware Tools tab, you see status information about the virtual machine. Click the link button to visit the VMware home page.



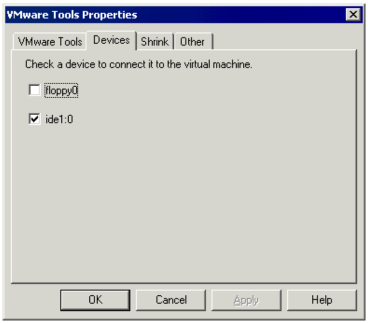
VMware Tools for Windows

$Name: build-987 $

Copyright ® 1998-2001

The VMware service is running.

3. To enable or disable removable devices, click the Devices tab. (You can also set these options from the Devices menu of the ESX Server remote console window.) The devices you can enable or disable include the server machine's floppy disk drive and the CD-ROM drive.



Check a device to connect it to the virtual machine.

□ floppy0

☑ ide1:0

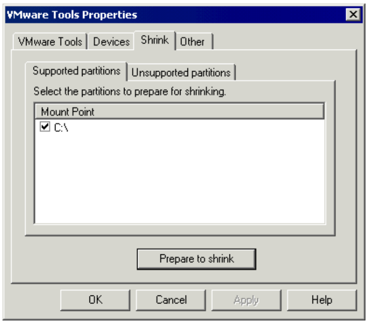
4. The Shrink tab lets you prepare to export a virtual disk to VMware GSX Server using the smallest possible disk files. This step is optional.

   Virtual disks on ESX Server take up the full amount of disk space indicated by the virtual disk's size. In other words, the `.dsk` file for a 4GB virtual disk will occupy 4GB of disk space.

GSX Server works differently. Under GSX Server, virtual disk files start small — only as big as needed to hold the data stored on the virtual disk — and grow as needed up to the designated maximum size.

If you plan to export a virtual disk for use under GSX Server, click the Shrink tab, be sure there is a check beside the name of the disk you plan to export, then click Prepare to shrink.

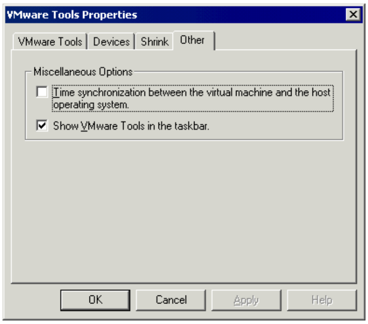
Note: When you export the virtual disk (using the `vmkfstools` program), a single virtual disk may be exported to multiple `.dsk` files.



Select the partitions to prepare for shrinking.

☑ C:\

5. On the Other tab, you can specify whether you want to synchronize the time between the virtual machine and the console operating system. You can also specify whether you want to display the VMware Tools icon in the system tray.



☐ Time synchronization between the virtual machine and the host operating system.

☑ Show VMware Tools in the taskbar.

Installing New Software Inside the Virtual Machine

Installing new software in an ESX Server virtual machine is just like installing it on a regular computer. You will need to have access to the ESX Server computer to insert installation CD-ROM discs or floppy disks into the server's drives. The following steps are based on using a Windows 2000 or Windows NT guest operating system. If you are using a Linux guest operating system, some details will vary.

1. Be sure you have started the virtual machine and, if necessary, logged on. Check the Devices menu to be sure the virtual machine has access to the CD-ROM and floppy drives.
2. Insert the installation CD-ROM or floppy disk into the proper drive. If you are installing from a CD-ROM, the installation program may start automatically.
3. If the installation program does not start automatically, click the Windows Start button, go to Settings > Control Panel, then double-click Add/Remove Programs and click Add New Programs. Follow the instructions on screen and in the user manual for your new software.

Cutting, Copying and Pasting

Be sure you have installed and started VMware Tools in your virtual machine.

In a Windows guest operating system, you will see a VMware Tools icon in the system tray when VMware Tools is running.

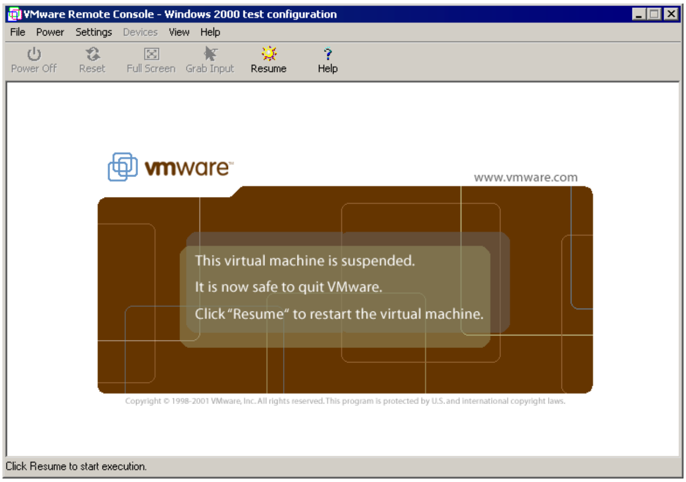
When VMware Tools is running, you can copy and paste text between applications in the virtual machine and the host computer or between two virtual machines.

Using Suspend and Resume

You can save the current state of your virtual machine. Then the resume feature lets you quickly pick up work right where you stopped — with all running applications in the same state they were at the time you suspended the virtual machine.

There are two ways to suspend a virtual machine:

With a remote console connected to that virtual machine, click Suspend on the button bar.



This virtual machine is suspended.

It is now safe to quit VMware.

Click “Resume” to restart the virtual machine.

Copyright © 1998-2001 VMware, Inc. All rights reserved. This program is protected by U.S. and international copyright laws.

Click Resume to start execution.

With the Web-based management interface connected to the virtual machine’s server, click the pause button (⏸) on the row for that virtual machine.

There are two ways to restore a virtual machine that you have suspended:

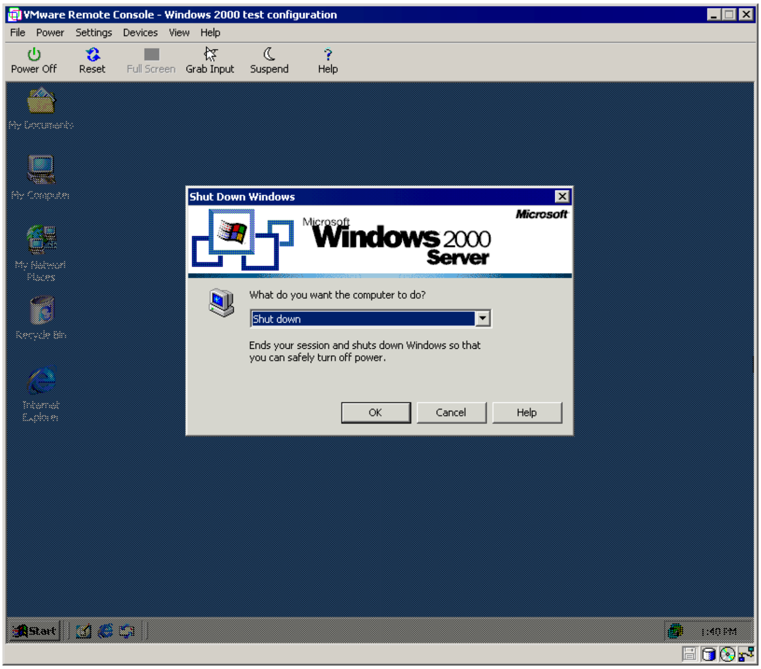
With a remote console connected to that virtual machine, click Resume on the button bar.

With the Web-based management interface connected to the virtual machine’s server, click the pause button (⏸) on the row for that virtual machine.

Shutting Down a Virtual Machine

The following steps are based on using a Windows 2000 or Windows NT guest operating system. If you are using a Linux guest operating system, follow the usual steps to shut down the guest operating system inside your virtual machine.

1. Select Shut Down from the Start menu of the guest operating system (inside the virtual machine).



What do you want the computer to do?

Shut down

Ends your session and shuts down Windows so that you can safely turn off power.

2. Select Shut Down, then click OK.

4

Reference: Guest Operating Systems

Installing Specific Guest Operating Systems

Guest operating system installation instructions assume you are using a remote console on a management workstation with a network connection to the server that hosts your virtual machine. You will need access to the server to insert CD-ROM discs or floppy disks that are needed to install the guest operating system.

For an overview of the guest operating system installation process, see Installing a Guest Operating System and VMware Tools on page 45.

Windows 2000 Installation Guidelines

Windows 2000 server versions can be installed in a virtual machine using the corresponding Windows 2000 distribution CD. Before installing the operating system, be sure that you have already created a new virtual machine and configured it using the ESX Server Virtual Machine Wizard.

Note: Some Microsoft Windows 2000 disks included with new computers are customized for those computers and include device drivers and other utilities specific to the hardware system. Even if you can install this Windows 2000 operating system on your actual computer, you may not be able to install it in a VMware ESX Server virtual machine. You may need to purchase a new copy of Windows to install in a virtual machine.

Windows 2000 Installation Steps

1. Before starting the installation, use the Web-based management interface to verify the virtual machine's devices are set up as you expect. For example, if you would like networking software to be installed during the Windows 2000 installation, be sure the virtual machine's Ethernet adapter is configured and enabled.
2. Insert the Windows 2000 CD in the CD-ROM drive.
3. Power on the virtual machine to start installing Windows 2000.
4. If you enabled the virtual machine's Ethernet adapter, a VMware PCI Ethernet Adapter will be detected and set up automatically.

VMware Tools

Be sure to install VMware Tools in your guest operating system. After you install VMware Tools, you need to change your Windows 2000 screen area to be greater than

640x480 pixels; otherwise, Windows 2000 uses the standard VGA driver, and your performance will suffer.

Windows NT Installation Guidelines

Windows NT 4.0 can be installed in a virtual machine using the standard Windows NT CD. Before installing the operating system, be sure that you have already created a new virtual machine and configured it using the ESX Server Virtual Machine Wizard.

Windows NT Installation Steps

1. Use the Web-based management interface to verify the virtual machine's devices are set up as you expect before starting the installation. For example, if you would like networking software to be installed during the Windows NT installation, be sure the virtual machine's Ethernet adapter is configured and enabled.
2. Insert the Windows NT CD in the CD-ROM drive.
3. Power on the virtual machine to start installing Windows NT.
4. If you have enabled the virtual machine's Ethernet Adapter, a VMware PCI Ethernet Adapter will be detected and set up automatically. The default settings should work fine and do not need to be changed.
5. Finish the Windows NT installation.

VMware Tools

Be sure to install VMware Tools in your guest operating system.

Red Hat Linux 7.0 Installation Guidelines

The easiest method of installing Red Hat Linux 7.0 in a virtual machine is to use the standard Red Hat distribution CD. The notes below describe an installation using the standard distribution CD; however, installing Red Hat Linux 7.0 via the boot floppy/ network method is supported as well. Before installing the operating system, be sure that you have already created a new virtual machine and configured it using the ESX Server Virtual Machine Wizard.

Note: Due to VGA performance issues installing Red Hat 7.0 with the graphics mode installer, we highly recommend you install the operating system with the text mode installer. At the Red Hat 7.0 CD boot prompt, you are offered the following choices:

```
To install or upgrade a system ... in graphical mode ...
To install or upgrade a system ... in text mode, type: text <ENTER>.
To enable expert mode, ...
Use the function keys listed below ...
```

Choose the text mode installer by typing `text` followed by `<ENTER>`.

Note: During the Red Hat Linux 7.0 text mode installation, a standard XFree86 version 4 server (without support for VMware SVGA or standard VGA) will be installed. Do not run that X server. Instead, to get an accelerated SVGA X server running inside the virtual machine, you should install the VMware Tools package immediately after installing Red Hat Linux 7.0.

Red Hat Linux 7.0 Installation Steps

1. Use the ESX Server Web-based management interface to verify the virtual machine's devices are set up as you expect before starting the installation. For example, if you would like networking software to be installed during the Red Hat Linux 7.0 install process, be sure the virtual machine's Ethernet adapter is enabled and configured.
2. Insert the Red Hat Linux 7.0 CD in the CD-ROM drive and click the Power On button. The virtual machine should start booting from the CD and the install process will begin.
3. Follow the installation steps as you would for a real PC. Be sure to make the choices outlined in the following steps.
4. In Video Card Selection choose Generic VGA compatible, then click OK.
5. Near the end of the installation, after files have been copied, you reach the Monitor Setup screen. Choose Generic Standard VGA, 640x480 @ 60 Hz, then click OK.
6. At the Video Memory screen, choose 256Kb, then click OK.
7. At the Clockchip Configuration screen, choose No Clockchip Setting (recommended), which is the default, then click OK.
8. At the Probe for Clocks screen, click Skip.
9. At the Select Video Modes screen, don't choose anything. Just click OK.
10. At the Starting X screen, click Skip.

    Note: This is the most important step. Clicking OK will run the XFree86 version 4 server, which will fail, and the installer will abort.
11. This completes basic installation of the Red Hat Linux 7.0 guest operating system.

    Note: We have occasionally observed an error message at the end of the Red Hat 7.0 installation process — one that sounds serious but does not, in fact, indicate a problem.

After your Red Hat 7.0 installation is completed and you click OK in the final dialog to reboot the machine, you might see this message:

```
Install exited abnormally -- recived signal 11
```

as the machine is being shut down. However, the Red Hat 7.0 installation has completed successfully, and the operating system will boot with no problems when you restart the virtual machine.

VMware Tools

Be sure to install VMware Tools in your guest operating system.

Note: With a Red Hat Linux 7.0 guest, you should install VMware Tools from the Linux console. Do not start X until you have installed VMware Tools.

Installing a 16-color X Server

If you want to run the standard 16-color VGA X server, skip the installation of VMware Tools and instead take the following steps.

Note: If you use the standard 16-color VGA X server, you will not have the performance advantages of the accelerated SVGA X server included in VMware Tools.

1. After you finish the basic installation of the Red Hat Linux 7.0 guest operating system and the virtual machine reboots, log in as root.
2. Set up the X server:

   ```
   ln -sf ../../usr/X11R6/bin/XF86_VGA16 /etc/X11/X
   ```

   This sets the current X server to `XF86_VGA16` (the XFree86 3.3.6 16-color VGA X server).

Red Hat Linux 6.2 Installation Guidelines

The easiest method of installing Red Hat Linux 6.2 in a virtual machine is to use the standard Red Hat distribution CD. The notes below describe an installation using the standard distribution CD; however, installing Red Hat Linux 6.2 via the boot floppy/ network method is supported as well. Before installing the operating system, be sure that you have already created a new virtual machine and configured it using the ESX Server Virtual Machine Wizard.

Note: Due to VGA performance issues installing Red Hat 6.2 with the graphics mode installer, we highly recommend you install the operating system with the text mode installer. At the Red Hat 6.2 CD boot prompt, you are offered the following choices:

```
To install or upgrade a system ... in graphical mode ...
To install or upgrade a system ... in text mode, type: text <ENTER>.
To enable expert mode, ...
Use the function keys listed below ...
```

Choose the text mode installer by typing `text` followed by `<ENTER>`.

Note: during the Red Hat Linux 6.2 installation, a standard VGA16 X server (without support for the VMware ESX Server X server) will be installed. To get an accelerated SVGA X server running inside the virtual machine, you should install the VMware Tools package immediately after installing Red Hat Linux 6.2.

Red Hat Linux 6.2 Installation Steps

1. Use the Web-based management interfaceto verify the virtual machine's devices are set up as you expect before starting the installation. For example, if you would like networking software to be installed during the Red Hat Linux 6.2 install process, be sure the virtual machine's Ethernet adapter is enabled and configured.
2. Insert the Red Hat Linux 6.2 CD in the CD-ROM drive and click the Power On button. The virtual machine should start booting from the CD and the install process will begin.
3. Follow the installation steps as you would for a real PC.

   Note: if the virtual machine's Ethernet adapter has been enabled, the installation program will auto-detect and load the AMD PC/Net 32 driver (no command line parameter will be necessary to load the driver).
4. During the Linux installation, select the standard VGA16 X server. Select the "Generic VGA compatible/Generic VGA" card from the list in the Choose a Card screen. Select the Generic Monitor entry from the list in the Monitor Setup screen. Select the Probe button from the Screen Configuration dialog and select OK from the Starting X dialog. When Linux is installed, the generic X server will be replaced with the accelerated X server included in the VMware Tools package.
5. Finish installing Red Hat Linux 6.2 as you would on a real PC.

   At this point Red Hat 6.2 will boot and present a login screen.

VMware Tools

Be sure to install VMware Tools in your guest operating system.

The VMware Guest Operating System Service

When you install VMware Tools in a virtual machine, the VMware guest operating system service is one of the primary components installed. The guest service can do the following:

- Execute commands in the virtual machine when it is requested to halt or reboot the guest operating system.
- Gracefully power off and reset a virtual machine.
- Send a heartbeat to ESX Server so that it knows the guest operating system is running.
- Synchronize the time of the guest operating system with the time in the console operating system.
- Pass a string from the console operating system to the guest operating system.

The guest service starts automatically when you boot the guest operating system.

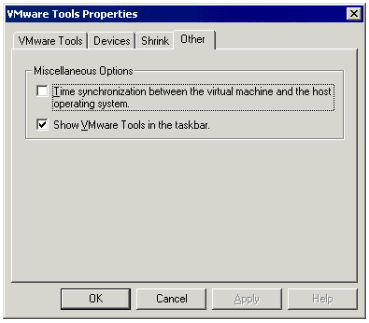
In a Windows guest, the guest service program file is called `VMwareService.exe`. For help information, right-click the VMware Tools icon in the system tray and choose Help.

In a Linux guest, the guest service is called `vmware-guestd`. To display help about the guest service, including a list of all options, use the following command:

```
/etc/vmware/vmware-guestd --help
```

Synchronizing the Time Between the Guest and Console Operating Systems

The guest service can synchronize the date and time between the guest and console operating systems once every second. In the VMware Tools control panel, on the Other tab (Options in a Linux guest), select Time synchronization between the virtual machine and the host operating system.



Miscellaneous Options

☐ Time synchronization between the virtual machine and the host operating system.

☑ Show VMware Tools in the taskbar.

In addition, the guest service can synchronize the date and time between the guest and console operating systems in response to various system events — for example, when you resume from disk. You can disable this in the configuration file by setting

```
time.synchronize.resume.disk = FALSE
```

Gracefully Powering Off or Resetting a Virtual Machine

You can have ESX Server ask the guest service to gracefully power off or reset a virtual machine. After the guest service receives a request to power off or reset, it sends an acknowledgment back to ESX Server.

You can send these requests from the Web-based management interface or the console operating system's command line.

Whether it is possible to gracefully power off or reset a virtual machine depends on the state of the virtual machine.

Gracefully Powering Off or Resetting a Virtual Machine from the Web-Based Management Interface

You can click 🟥 to gracefully power off or ↻ to gracefully reset a virtual machine from the Web-based management interface. These operations are also available from the menu that appears when you hold your mouse over the terminal icon (🖥). After you select one of these operations, you should click to the Event Log page for this virtual machine to respond to any messages that require a response.

If you receive an event log message saying, “You will need to power off or reset the virtual machine at this point,” you must connect to the virtual machine with a remote console and click the power off or reset button to complete the operation.

The power off and reset commands are not available while these operations are in progress.

You can also force power off or force reset from the menu. These commands bypass the guest service and perform the virtual equivalent of shutting off the power to the machine or pressing the reset button.

Gracefully Powering Off or Resetting a Virtual Machine from the Command Line

You can gracefully reset and power off a virtual machine from the console operating system command line using the `vmware-control` program.

The following commands return you to the command prompt immediately, before they finish executing, although the power off orreset may take some time to complete:

```
vmware-control /<path_to_config_file>/<configfile>.cfg \
request_stop
```

```
vmware-control /<path_to_config_file>/<configfile>.cfg \
request_reset
```

Note: Enter the `vmware-control` command you want to use on a single line. Do not type the backslash.

Executing Commands When ESX Server Requests the Guest Service to Halt or Reboot a Virtual Machine

In a Linux guest, you can have the guest service execute specific commands when ESX Server asks it to halt or reboot the virtual machine's guest operating system. If you use nonstandard utilities, or want to do additional things before shutting down or rebooting the guest operating system, you can override the default commands the guest service executes by modifying the `/etc/vmware/dualconf.vm` startup script in the guest to start the guest service with the following command line options:

```
/etc/vmware/vmware-guestd --halt-command <command>
```

where `<command>` is the command to execute when ESX Server asks the guest service to halt the guest operating system

```
/etc/vmware/vmware-guestd --reboot-command <command>
```

where `<command>` is the command to execute when ESX Server asks the guest service to reboot the guest operating system

Passing a String from the Console Operating System to the Guest Operating System

With ESX Server and knowledge of a scripting language like Perl or NetShell (in a Windows 2000 guest operating system), you can pass a string from your virtual machine's configuration file to the guest operating system when you use the configuration file to launch a virtual machine. This string is known as `machine.id`. The content of the string you pass to the guest operating system is up to you.

You should use this feature only if you have a good understanding of a scripting language and know how to modify system startup scripts.

Example

If you use multiple configuration files that point to the same virtual disk, each configuration file can contain its own unique `machine.id` line.

`<config_file_1>.cfg` contains:

```
scsi0:1.present = TRUE
scsi0:1.name = "my_common_virtual_hard_drive.dsk"
scsi0:1.mode = "persistent"
machine.id = "the_id_for_my_first_vm"
```

`<config_file_2>.cfg` contains:

```
scsi0:1.present = TRUE
scsi0:1.name = "my_common_virtual_hard_drive.dsk"
scsi0:1.mode = "persistent"
machine.id = "the_id_for_my_second_vm"
```

Using `machine.id`, you may pass such strings as the Windows system ID (SID), a machine name or an IP address. In the guest operating system startup script, you may then have the guest service retrieve this string, which can then be used by your script to set your virtual machine's system ID, machine name or IP address.

In the following example, we use a Linux guest to illustrate how you can use the guest service to retrieve a string containing what will become the virtual machine's machine name and IP address. We use RedHat62VM as the machine name and 148.30.16.24 as the IP address.

1. Define the machine.id string. Add the following line to your virtual machine's configuration file:

   ```
   machine.id = "RedHat62VM 148.30.16.24"
   ```

   Then launch a virtual machine using this configuration file.

2. Retrieve the `machine.id` string in the virtual machine. In your system startup script, before the network startup section, add the following command:

   ```
   /etc/vmware/vmware-guestd --cmd 'machine.id.get'
   ```

   Note: in a Windows guest, the command to retrieve the string is

   ```
   VMwareService --cmd machine.id.get
   ```

You need to further customize this startup script so it uses the string the guest service retrieved during startup to set the virtual machine's network name to RedHat62VM and its IP address to 148.30.16.24. This should be located in the script before the network services are started. If you're using a Windows 2000 guest operating system, for example, you can call the NetShell utility (`netsh`) and pass it the contents of the string, which it can then use appropriately (that is, it can set a new IP address for the virtual machine, if that is what was passed in the string originally).

From your console operating system, you can prevent the console operating system from passing a string to the guest operating system via the guest service. To do this, set the following line in your virtual machine's configuration file.

```
isolation.tools.machine.id.get.disable = TRUE
```

5

Reference: Console Operating System and VMkernel

Characteristics of the VMware Console Operating System

The purpose of the VMware Console Operating System is to start up and administer your virtual machines. It is a customized version of Linux based on the Red Hat 6.2 distribution. It has been modified to allow itself to be managed by the VMkernel.

The console operating system has been customized to disable unneeded services. In particular, most network services have been disabled, except for telnet and auth. You may want to disable telnet for added security and use ssh instead for remote access to the console operating system.

The following services in `/etc/rc.d/init.d` are left enabled:

network
configures network interfaces when the system is booted

syslog
logging of system messages

atd
runs jobs queued by `at`

crond
runs cron jobs

inet
`telnet` and `auth` enabled, also `vmware-authd` in `/etc/inetd.conf`

lpd
printer daemon

gpm
console mouse support

xfs
X font server

The console operating system is scheduled by the VMkernel just as any other virtual machine is. You should not attempt to run heavy workloads on the console operating system, because it will take processor cycles away from your virtual machines.

You should also avoid running X on the console operating system. However, if you do choose to run X and the GNOME desktop, remember to disable any GNOME applications that automount the CD-ROM, such as magicdev. Otherwise the CD-ROM will be unavailable to the guest operating systems. You can either use the command `killall magicdev` or edit `.gnome/magicdev` and add the following line:

```
do_automount=false, do_cd_play=false
```

Using DHCP for the Console Operating System

The recommended setup is to use static IP addresses for the console operating system. It is also possible to set up the console operating system to use DHCP, as long as your DNS server is capable of mapping the console operating system's host name to the dynamically-generated IP address. If your DNS server cannot map the host's name to its DHCP-generated IP address, which may be the case, you will have to determine the console operating system's numeric IP address yourself and use that numeric address when accessing the management interface's Web pages. Keep in mind that the numeric IP address may change as DHCP leases run out or when the system is rebooted. For this reason, we do not recommend using DHCP for the console operating system unless your DNS server can handle the host name translation.

Loading and Unloading VMkernel

The VMkernel Loader

The program `vmkloader` loads or unloads the VMkernel. With no flags, it loads the VMkernel specified by `<vmkernel-binary>`. If the VMkernel is already loaded, the load will fail.

If the unload option (`-u`) is specified, the `<vmkernel-binary>` argument is ignored and the VMkernel is unloaded as long as no virtual machines are currently running on the VMkernel. If there are virtual machines running, then the unload fails. If the force option is specified (`-f`), `vmkloader` unloads the VMkernel even if there is a virtual machine running.

If you have a SCSI adapter or RAID controller shared between the console operating system and the virtual machines, you cannot unload the VMkernel.

Options

`-f`
Unload the VMkernel even if a virtual machine is currently running on it.

`-n <num>`
Force the VMkernel and all virtual machines to run on only `<num>` processors, even if the physical machine has more than `<num>` processors.

`-s`
Force the VMkernel and all virtual machines to run on only a single processor, even if the physical machine is a multiprocessor computer.

`-u`
Unload the VMkernel

Examples

```
vmkloader /usr/lib/vmware/vmkernel
```

loads the VMkernel binary `/usr/lib/vmware/vmkernel`.

```
vmkloader -u
```

unloads the VMkernel if there are no virtual machines running.

```
vmkloader -uf
```

unloads the VMkernel even if there are virtual machines running.

Configuring Your Server to Use VMkernel Device Modules

Loading VMkernel Device Modules

The install process should detect the devices that are assigned to the VMkernel and automatically load appropriate modules into the VMkernel to make use of these devices.

However, there may be situations in which you wish to load VMkernel device modules explicitly. Modules supported in this release are located in `/usr/lib/vmware/vmkmod`. The command `vmkload_mod(1)` loads VMkernel modules.

VMkernel module loader

The program `vmkload_mod` is used to load device driver and network shaper modules into the VMkernel. `vmkload_mod` can also be used to unload a module, list the loaded modules and list the available parameters for each module.

The format for the command is

```
vmkload_mod <options> <module-binary> <module-tag> \
<parameters>
```

Note: The command should be typed on one line. Do not type the backslash.

`<module-binary>` is the name of the module binary that is being loaded. `<module-tag>` is the name that the VMkernel associates with the loaded module. The tag can be any string of letters and numbers. If the module is a device driver, the VMkernel will name the module with the `<module-tag>` plus a number starting from zero. If there are multiple device instances created by loading the module or multiple device driver modules loaded with the same tag, each device will get a unique number based on the order in which device instances are created.

The `<module-binary>` and `<module-tag>` parts of the command line are required when a module is loaded and are ignored when the `--unload`, `--list`, and `--showparam` options are used. The `<parameters>` part of the command line is optional and is used only when a module is being loaded.

Options

```
-l
--list
```

List out the current modules loaded. If the `-l` option is given, other arguments on the command line are ignored.

```
-u <module-binary>
--unload <module-binary>
```

Unload the module named `<module-binary>`.

```
-v
--verbose
```

Be verbose during the module loading.

```
-d <scsi-device-name>
--device <scsi-device-name>
```

The module being loaded is for a SCSI adapter that is currently being used by the console operating system. After the module is loaded the SCSI adapter will be controlled by the VMkernel but the console operating system will continue to be able to access all SCSI devices. The format of `<scsi-device-name>` is `<PCI-Bus>:<PCI-slot>`.

```
-e
--exportsym
```

Export all global exported symbols from this module. This will allow other modules to use exported functions and variables from the loaded module. This option should not be used for normal device driver and shaper modules since there may be symbol conflicts.

```
-s
--showparam
```

List all available module parameters that can be specified in the `<parameter>` section of the command line.

Parameters

Modules can specify parameters that can be set on the command line. A list of these parameters is shown via the `--showparam` option. In order to set one of these parameters, you must specify a name-value pair at the end of the command line. The syntax is of the form `<name>=<value>`. Any number of parameters can be specified.

Examples

```
vmkload_mod ~/modules/e100.o vmnic debug=5
```

loads the module `~/modules/e100.o` into the VMkernel. The tag for this module is `vmnic`. Each EEPro card that was assigned to the VMkernel will be given the name `vmnic<#>`, where `<#>` starts at 0. For example, if there were two EEPro cards assigned to the VMkernel, they would have VMkernel names of vmnic0 and vmnic1. The module parameter debug will be set to the value 5.

```
vmkload_mod --device 0:12 ~/modules/aic7xxx.o vmhba
```

loads the module ~/modules/aic7xxx.o into the VMkernel. The tag for this module is vmhba. The Adaptec SCSI adapter is currently being used by the console operating system. The SCSI adapter is located on PCI bus 0, slot 12.

```
vmkload_mod --exportsym ~/modules/vmklinux linuxdrivers
```

loads the module ~/modules/vmklinux into the VMkernel. All exported symbols from this module will be available to other modules that are subsequently loaded. The vmklinux module is the module that allows Linux device drivers to run in the VMkernel so it is one of the few modules for which the --exportsym option makes sense.

Here are several examples of command lines that load various modules:

Preparing to load modules

```
vmkload_mod -e /usr/lib/vmware/vmkmod/vmklinux linux
```

This command must be given before you load other device modules. It loads in common code that allows the VMkernel to make use of modules derived from Linux device drivers to manage its high-performance devices. The -e option is required so that the vmklinux module exports its symbols, making them available for use by other modules.

Loading modules

```
vmkload_mod /usr/lib/vmware/vmkmod/e100.o vmnic
vmkload_mod /usr/lib/vmware/vmkmod/aic7xxx.o vmhba
```

The first of these commands loads a module to control the EEPro Ethernet device(s) reserved for the VMkernel. The second loads a module to control the Adaptec SCSI device(s). The last argument supplied (vmnic and vmhba in the above examples) determines the base name that VMware uses to refer to the device(s) in the VMware virtual machine configuration file.

For example, suppose your machine has two EEPro Ethernet cards and three Adaptec SCSI cards, and you assigned one Ethernet card and two SCSI cards to the VMkernel during the installation process. After you issue the two commands above, the EEPro Ethernet card assigned to the VMkernel is given the name vmnic0 and the two SCSI cards assigned to the VMkernel are given the names vmhba0 and vmhba1.

Note: the Adaptec VMkernel module need only be loaded once, even though two Adaptec SCSI cards are assigned to the VMkernel.

The VMkernel can also share SCSI adapters with the console operating system, rather than exclusively controlling them. The installation process will allow you to specify SCSI adapters that are shared and load the device module appropriately. However, if you wish to control the sharing explicitly, assign the SCSI device to the console operating system during the installation process. Then load the VMkernel SCSI module using the following syntax:

```
    vmkload_mod -d bus:slot \
    /usr/lib/vmware/vmkmod/aic7xxx.o vmhba
```

Note: This command should be entered on a single line. Do not type the backslash.

To obtain the bus and slot (also known as device or cardnum) information, examine `/proc/pci`, output from the `scanpci` command, or both.

Note: The device must be correctly assigned to the console operating system. Devices assigned exclusively to the VMkernel during the installation process will no longer appear in `/proc/pci`.

After you load a VMkernel device module, an entry will appear in `/proc/vmware/net` or `/proc/vmware/scsi`. For example, when `e100.o` is loaded as described above, the entry `/proc/vmware/net/vmnic0` will appear, indicating there is one EEPro card controlled by the VMkernel and available as `vmnic0` to the virtual machines. See Configuring Virtual Machines on page 92 for information on how to configure virtual machines to use VMkernel devices.

Other Information about VMkernel Modules

The only non-device VMkernel module available in this release of VMware ESX Server is the `nfshaper` module, which provides support for network filtering, as described in the technical note Network Resource Management. Load `nfshaper` using the following syntax.

```
vmkload_mod /usr/lib/vmware/vmkmod/nfshaper.o nfshaper
```

To list the loaded VMkernel modules, run `vmkload_mod -l`. For more information on loading VMkernel modules, see Loading and Unloading VMkernel and VMkernel Modules on page 84.

VMkernel modules must be reloaded each time the VMkernel is loaded. If you have configured your system to load the VMkernel automatically on each reboot, you can have the modules loaded automatically as well by adding entries to the file `/etc/vmware/vmkmodule.conf`. The `vmkmodule.conf` file is read only if it contains a comment line containing the keyword `MANUAL-CONFIG`. Otherwise, the configuration is obtained automatically from the database of the management interface.

Each line that is not blank and does not begin with # should contain the name of a module file, the tag that will be associated with the module in the VMkernel and possibly a sharing specification (the argument specified with the -d flag above). The module file should just be the base file name, without the /usr/lib/vmware/... path. A sample vmkmodule.conf file is:

```
# MANUAL-CONFIG
vmklinux.o linux
nfshaper.o nfshaper
e100.o vmnic
aic7xxx.o vmhba 0:1
```

6

Reference: Configuring and Running Virtual Machines

Configuring Virtual Machines

The simplest way to set up a new virtual machine is to use the Virtual Machine Configuration Wizard, as described in the section on installing the ESX Server software on page 18.

Key configuration settings for an existing virtual machine can be changed from the Web-based management interface. The virtual machine must be powered off when you change the configuration.

1. Log in to the server from the Web-based management interface (`http://<hostname>:8222/`) as a user who has rights to change the configuration file.
2. From the server's overview page (`http://<hostname>:8222/overview`), click the link under the name of the virtual machine you want to reconfigure.
3. On the details page for that virtual machine, click Edit VM Configuration.
4. Make any changes you wish to the configuration, then click Apply Changes.

To modify other settings in the configuration, log in to the VMware Console Operating System. Then manually edit the configuration file as described in this section. For purposes of illustration, we will assume that you are working with the file `newvm.cfg` in a directory named `/vms/vm1`.

There may also be situations when you want to create virtual machines that are more complex than you can create using the Web-based interface. In these cases, you will need to start with the configuration file template — `/usr/share/doc/vmware/sample.cfg` — copy it to a new file and manually edit the copy as described in this section.

Using VMkernel Devices

The VMkernel devices — whether shared or not — must be referenced and activated in the VMware virtual machine's configuration (`.cfg`) file, as described in this section. You must also load a special VMware network driver into the guest operating system, as described in the section on how to install VMware Tools on page 47.

Ethernet

The Ethernet section of the configuration will be in this format:

```
ethernet0.present = TRUE
ethernet0.connectionType = monitor_dev
ethernet0.virtualDev = vmxnet
```

```
    ethernet0.devName = vmnic0
    ethernet0.exclusive = TRUE
```

In this configuration, `ethernet0.connectionType = monitor_dev` and `ethernet0.virtualDev = vmxnet` specify that the virtual machine's Ethernet will use the VMkernel high-performance network device. `ethernet0.devName = vnmic0` specifies that the virtual network device will correspond to the first network device activated by the `vmkload_mod .../vmkernel .../XXX.o vmnic` command. The `ethernet0.exclusive = TRUE` line makes the networking more efficient if only one virtual machine will be using the network card. You should remove this line if more than one virtual machine will use the card.

VMFS Virtual SCSI Disks

VMware ESX Server supports a simple file system known as VMFS (VMWare ESX Server File System) on physical SCSI disks and partitions to make it easy to allocate space for a disk image. VMFS allows many disk images to stored on one large physical SCSI disk or partition. The Web-based management interface will automatically create VMFS file systems and VMFS files as you configure your system and create virtual machines. However, VMFS files can also be created and managed via the `vmkfstools(1)` command. There is almost no performance penalty for accessing disk images allocated using this file system, since accesses are still via raw SCSI I/O. An example configuration that use a disk image allocated in a VMFS is:

```
    scsi0.present = TRUE
    scsi0.virtualDev = vmxbuslogic

    scsi0:2.present = TRUE
    scsi0:2.name = vmhba1:3:0:2:data.dsk
```

In this configuration, `scsi0.present = TRUE` specifies that the virtual machine has a SCSI adapter called `scsi0`. `scsi0.virtualDev = vmxbuslogic` specifies that the virtual machine's first SCSI adapter will access data from the VMkernel SCSI device. `scsi0:2.name = vmhba1:3:0:2:data.dsk` specifies the location of the disk image that will be used for SCSI target 2 on the first virtual SCSI adapter.

The location of the disk image is specified in a notation with the form `<adapterName>:<target>:<lun>:<partition>:<fileName>.` An adapter name such as `vmhba1` specifies the second physical SCSI adapter activated by the `vmkload_mod .../XXX.o vmhba` command. The second component of the location specifies the ID of the target on the named adapter. The third component specifies the LUN (logical unit number) and is typically zero. The fourth component specifies the partition. The last component specifies the name of the disk image in the VMFS file system on the specified partition.

So `scsi0:2.name=vmhba1:3:0:2:data.dsk` indicates that the disk image is in the file `data.dsk` on partition 2 of the disk at target 3 and LUN 0 on the second SCSI adapter activated by the `vmkload_mod .../XXX.o vmhba` command.

A specification may have a partition specified as 0, in which case it refers to a VMFS that covers a complete, unpartitioned disk (target). However, if your SCSI adapter is shared with the console operating system, rather than assigned exclusively to the VMkernel, you cannot access a VMFS that covers the entire disk. Thus, we recommend that you always create at least one partition on each disk and create the VMFS within that partition.

For information on copying an existing virtual disk from the console operating system to a VMFS file, see the section Migrating VMware Workstation and VMware GSX Server Disks on page 45.

Note: If you have not determined which SCSI target ID corresponds to the disk you wish to use in the virtual machine, see Determining SCSI Target IDs on page 111.

Access Modes

By default, disk images on a raw SCSI disk are accessed in persistent mode. That is, all changes are written directly to the disk image and cannot be undone. This mode provides the most efficient access to the data. ESX Server also supports nonpersistent, undoable and append modes. You can change the disk mode setting on the Edit VM Configuration page of the Web-based management interface. The virtual machine must be powered down before you change the disk mode. You can also make the changes directly in the configuration file by including lines in the following format:

```
scsi0:2.mode = nonpersistent
```

or

```
scsi0:2.mode = undoable
```

If the mode of a disk image is nonpersistent, any changes to the disk will be lost when the associated virtual machine shuts down. If the mode of the disk image is undoable, the changes will be maintained in a separate file, known as the redo log, on the SCSI disk. Each time the virtual machine is powered down, a dialog will ask whether changes made to the disk during the current session should be discard, committed to the base disk image, or appended (kept in the redo log).

VMware ESX Server supports an additional append mode for disk images stored as VMFS files. Like undoable mode, append mode maintains a redo log. However, in this mode, no dialog appears when the virtual machine is powered off to ask whether you want to commit changes. All changes are continually appended to the redo log. At any point, you can undo all the changes by removing the redo log. Its name is derived from the original name of the file that contains the disk by adding `.REDO`. Changes can be committed permanently to the base disk image via the commit option of the `vmkfstools` command.

Virtual SCSI Disks on the Console Operating System

VMware ESX Server also supports virtual SCSI disks that are stored on the file system of the console operating system. Virtual SCSI disks created under VMware Workstation 2.x are supported, although a new network driver will need to be loaded onto the disk for use by the guest operating system. Disks created under VMware GSX Server are also supported. For details, see Migrating VMware Workstation 2.x and VMware GSX Server Disks on page 45.

To create a new, blank virtual SCSI disk for your virtual machine, copy the file `/usr/lib/vmware/virt-scsi.dsk` from the ESX Server installation CD-ROM to the working directory for your virtual machine.

```
    cp virt-scsi.dsk /vms/vm1/virt-scsi.dsk
```

Then add lines to your virtual machine's configuration file to describe the new disk. Those lines have the following format:

```
scsi0.present = TRUE
scsi0.virtualDev = buslogic

scsi0:1.present = TRUE
scsi0:1.fileName = virt-scsi.dsk
scsi0:1.mode = nonpersistent
```

Note: Using virtual disks will not take advantage of ESX Server's new high-performance SCSI disk architecture and therefore the performance of the virtual machine may suffer.

Naming VMFS File Systems

If you create a VMFS file system on a SCSI disk or partition, you can give a name to that file system and use that name when specifying VMFS files on that file system. For instance, suppose you have a VMFS file system on the SCSI partition `vmhba0:3:1` and have created a VMFS file `nt4.dsk`. You can name that file system either using the Web-based configuration wizard or via a `vmkfstools` command such as

```
vmkfstools -S mydisk vmhba0:3:1
```

You can then refer to the `nt4.dsk` file as `mydisk:nt4.dsk` (instead of `vmhba0:3:1:0:nt4.dsk`) in a virtual machine configuration file and in other `vmkfstools` commands. Naming VMFS file systems is especially useful if you may be adding SCSI adapters or disks to your system, in which case the actual disk and target numbers specifying a particular VMFS may change, but the name will stay the same.

Recommended Configuration Options

This section details options that can influence the performance of your virtual machines. These settings are not required to run VMware ESX Server correctly.

SleepWhenIdle

The configuration file option `monitor.SleepWhenIdle` determines whether the VMkernel deschedules an idle virtual machine. By default, this option is enabled, a setting that ensures much better performance when running multiple virtual machines.

When you are running running only a single virtual machine (such as for benchmarking VMware ESX Server), add the following line to the virtual machine's configuration (`.cfg`) file if you want to achieve the best possible performance in the virtual machine (at the expense of responsiveness in the console operating system):

```
monitor.SleepWhenIdle = 0
```

Suspending and Resuming Virtual Machines

Suspending a virtual machine, then later resuming its operation, can speed provisioning tasks — for example, deployment of standby servers. VMware ESX Server supports two configurations for resuming a suspended virtual machine.

- You can suspend a running virtual machine at any time, then resume operation, suspend at a later time, then resume with the machine in the second state, and so on.
- You can suspend a virtual machine at any desired point in its operation, then lock in the suspended state at that chosen point. Any time you restart the virtual machine, it will resume in the same state — the state it was in when you first suspended it.

Note: You should not change a configuration file after you suspend a virtual machine, since the virtual machine will not resume properly if the configuration file is inconsistent with the suspended virtual machine. Also, you should not move any physical disks or change the name of any VMFS file systems that the virtual machine uses, since the virtual machine will not then be able to access its virtual disks when it resumes.

You can also set the configuration of each virtual machine so the file that stores information on the suspended state is saved in a location of your choice.

Suspend Directory

When a virtual machine is suspended, its state is written to a file with a `.std` extension. By default, the `.std` file is written to the same directory as the configuration file. Similarly, when a virtual machine is being resumed, ESX Server looks for the `.std` file in the same directory as the configuration file. You can change the directory where the `.std` file is stored by adding a line to the virtual machine's configuration file, using this format:

```
suspend.directory = /vmfs/vmhba0:0:0:8
```

Note that the configuration file points to a directory referencing a VMFS disk. (For more information on VMFS disks, see Mounting VMFS File Systems on the Console Operating System on page 108.) Using this format, you can store the `.std` file on a VMFS file system on a fast SCSI disk. You can reduce suspend and resume times by storing the `.std` file on a fast SCSI disk, especially if the file system containing the configuration file is on an IDE disk or is remote.

Repeatable Resumes

The configuration file option `resume.repeatable` determines the system's behavior when you suspend a virtual machine, then resume operation. When a virtual machine is suspended, a file with a `.std` extension is written out. This file contains the entire state of the virtual machine. When the virtual machine is resumed, its state is restored from the `.std` file. If you have not modified the virtual machine's configuration, the `.std` file is then removed.

This behavior ensures that a `.std` file is used only once to resume a virtual machine — which is the safest behavior. Note that a virtual machine you have suspended and resumed may be suspended again, creating a new `.std` file.

If, on the other hand, you want to be able to resume a virtual machine in the same state repeatedly — for example, in a QA testing environment — include the following line in that virtual machine's configuration file.

```
resume.repeatable = TRUE
```

If this line is present, the `.std` file is not deleted after it is used to resume the virtual machine. When you power off the resumed virtual machine, you can again resume its operation in the same state using the same `.std` file.

This option makes it easy to start a virtual machine again and again in the exact same state. If you no longer want to resume the virtual machine using the existing `.std` file, you must remove the `.std` file manually. Once it is deleted, you may suspend the virtual machine in a new state to create a new `.std` file.

Authentication and Security Features

There are three key aspects to security with VMware ESX Server.

- VMware ESX Server authenticates all remote users who connect to a server using the Web-based management interface or the remote console.
- Network traffic to and from the server may be secured using SSH or other security software.
- Two TCP/IP ports are used for access. The Web-based management interface uses port 8222. The remote console uses port 902. Depending on your remote access requirements, you may need to configure your firewall to allow access on one or both of these ports.

Authenticating Users

VMware ESX Server uses Pluggable Authentication Modules (PAM) for user authentication in the remote console and the Web-based management interface. The default installation of ESX Server uses `/etc/passwd` authentication, just as Linux does, but it can easily be configured to use LDAP, NIS, Kerberos, or another distributed authentication mechanism.

The PAM configuration is in `/etc/pam.d/vmware-authd`.

Every time a connection is made to the server running VMware ESX Server, the `inetd` process runs an instance of the VMware authentication daemon (`vmware-authd`). The `vmware-authd` process requests a user name and password, then hands them off to PAM, which performs the authentication.

Once a user is authenticated, `vmware-authd` accepts a path name to a virtual machine configuration file. Access to the configuration file is restricted in the following ways:

- The user must have **read** access to the configuration file to see and control the virtual machine in the Web-based management interface and to view the Details and Event Log pages.
- The user must have **read** access to the configuration file to use the local console on the console operating system or to connect to the virtual machine with the VMware Perl API.
- The user must have **read** and **execute** access to the configuration file to connect to and control (start, stop, reset or suspend) a virtual machine in a remote console, with the VMware Perl API or with the management interface.

- The user must have **read** and **write** access to the configuration file to change the configuration using the Configure VM page in the Web-based management interface.

Note: If you have users with **list** access, but not **read** access, they may encounter errors in the Web-based management interface.

If a `vmware` process is not running for this configuration file, `vmware-authd` examines `/etc/vmware/vm-list`, the file where you register your virtual machines. If the configuration file is listed in `vm-list`, `vmware-authd` (not necessarily the user who is currently authenticated) becomes the owner of the configuration file and starts VMware ESX Server with this configuration file as an argument (for example, `vmware /<path_to_config>/<configfile>.cfg`).

Registered virtual machines (those listed in `/etc/vmware/vm-list`) also appear in the Web-based management interface. The virtual machines you see on the Overview page must be listed in `vm-list`, and you must have read access to their configuration files.

The `vmware-authd` process exits as soon as a connection to a `vmware` process is established. Each `vmware` process shuts down automatically after the last user disconnects.

Default Permissions

When you create a virtual machine with VMware ESX Server, its configuration file is registered with the following default permissions, based on the user accessing it:

- Read, execute and write — for the user who created the configuration file (the owner)
- Read and execute — for the group
- Read — for users other than the owner or a member of the owner's group

Security on the Network

The user name and password sent to the server when you use the remote console or the management interface are not encrypted in this version of ESX Server.

If you are managing the server and its virtual machines entirely on a secure network, you may not need to take any special network security measures. However, if you are using the product on an externally visible system or in a less secure environment, you should tunnel the network connection between the remote management workstations and the server using SSH2 or another form of strong encryption.

SSH Communications Security provides a commercial version of SSH2, with software available for a variety of server and client operating systems. The server version that supports Red Hat Linux with kernel 2.2.x on an Intel platform can be used with the ESX Server console operating system.

OpenSSH is a free version of the SSH tools. The "portable" version of OpenSSH 2.2.1 or higher is appropriate for use with the ESX Server console operating system.

Once you have the software running on the ESX Server console operating system and on the remote workstation, you can use an SSH2 command to start a tunnel connection from the workstation to the server. If you are using OpenSSH, for example, you would establish a tunneled connection like this.

1. As root on the remote workstation, type the following at the command line:

   ```
   ssh -f -n -N -x -2 -L 902:localhost:902 -L \
   8222:localhost:8222 <user>@<hostname>
   ```

   Note: Type the command on one line. Do not type the backslash.

2. Then do one of the following:

   - Connect to the Web-based management interface via the SSH tunnel by pointing your browser to the following URL:
     `http://localhost:8222/login`
   - Using the remote console, specify localhost as the host, and use port 902 as you normally would.

Alternately, you can create a configuration file with the SSH options shown in the command line above. Save this file in `~root/.ssh/config`.

For greater security, you can lock down the server. Key steps are listed below. For more information, see the comp.os.linux.security FAQ.

1. Completely disable any unnecessary services, such as NFS or telnet.
2. Lock out all remote connections — especially any connections not from the local host on ports 902 and 8222 — except those on port 22, the location of the SSH service. You need to modify the configurations of `inetd` and `apache` respectively.
3. If you want to prevent remote users from accessing a shell or running arbitrary commands via SSH, modify the `sshd` configuration file on the server to restrict connections to those with known DSA keys, then edit the `/etc/ssh/authorized_keys2` file by adding the following options along with the authorized keys.

```
command="/bin/sleep 8h",no-port-forwarding,\
no-X11-forwarding,no-pty <hostkey>
```

Note: Type the entry on one line. Do not type the backslash.

These options specify that port forwarding times out after 8 hours, ports are not forwarded from the server to the client and a pty terminal is not allocated.

4. Further disable shell access by invalidating users' shell accounts in `/etc/passwd`.

7

Reference: Disks

File System Management on SCSI Disks and RAID

The VMFS file system is a simple, high-performance file system on physical SCSI disks and partitions, used for storing large files such as the disk images for ESX Server virtual machines and, optionally, the memory images of suspended virtual machines. A server's VMFS file systems are mounted automatically by the console operating system and appear in the `/vmfs` directory.

Files in these mounted VMFS file systems can be viewed and manipulated with ordinary file commands such as `ls` and `cp`. As noted later in this section, there are limitations when you use some standard disk utilities with files in a VMFS file system — limitations caused by the fact that the utilities often assume a file is no larger than 2GB. The `vmfkstools` program provides additional functions that are particularly useful when you need to create files of a particular size and when you need to import files from and export files to the console operating system's file system. In addition, `vmfkstools` is designed to work with large files, overcoming the 2GB limit of some standard file utilities.

Using vmfkstools

To create and manipulate files on SCSI disks managed by VMware ESX Server, use `vmkfstools`. It supports the creation of a VMware ESX Server file system (VMFS) on a SCSI disk or partition and the management of files stored in the VMFS. It is useful for storing multiple virtual disk images on a single SCSI disk or partition of a SCSI disk.

The format for the command is

```
    vmkfstools <options> <device>[:<file>]
```

The `vmkfstools` command is issued with a device specification and one or more options.

`<device>` specifies a SCSI device (a SCSI disk or a partition on a SCSI disk) being manipulated and `<options>` specifies the operation to be performed.

`<device>` is specified in a form such as:

`vmhba1:2:0:3`

Here, `vmhba1` specifies the second SCSI adapter activated by the `vmkload_mod .../XXX.o vmhba` command. The second number specifies the target on the adapter, the third number specifies the LUN (logical unit number) and the fourth number specifies the partition. If the partition number is zero, the whole disk is implied; otherwise, the indicated partition is specified.

`<file>` is the name of a file stored in the file system on the specified device.

Options

The long and short forms of options, shown together in this list, are equivalent.

```
-C   --createfs
-b   --blocksize #[mMkK]
-n   --numfiles #
```

Create a file system on the specified SCSI device. The file block size can be specified via the -b option. The default file block size is 1MB. The maximum number of files in the file system can be specified with the `-n` option. The default maximum is 64 files.

```
-N   --consolename
```

Print out the name of a Linux device that represents the specified SCSI device on the console operating system. The resulting device name can be used in accessing the SCSI device via commands such as `fdisk` on the console operating system. The association between the Linux device name and the specified SCSI device lasts only until ESX Server is unloaded or the machine is rebooted.

```
-c   --createfile #[mMkK]
```

Create a file with the specified size on the file system of the specified SCSI device. The size is specified in bytes by default, but it can be specified in kilobytes or megabytes by adding a suffix of `k` or `m` respectively.

```
-e   --exportfile <dstFile>
```

Export the contents of the specified file on the specified SCSI device to a virtual disk on the file system of the console operating system. The virtual disk may then be transferred to another machine and imported to a SCSI device on the remote machine. Hence the combination of `exportfile` and `importfile` may be used for copying images of raw SCSI disks to remote machines. The virtual disk will likely take less space than the full size of the raw disk, since the virtual disk does not include zeroed sectors of the disk.

`-d   --copyfile <srcFile>` or
`-i   --importfile <srcFile>`

Import the contents of a VMware virtual, plain or raw disk on the host to the specified file on the specified SCSI device. This command is often used to import the contents of a VMware Workstation virtual disk onto a SCSI device. It may also be used to import a virtual disk that was created by exporting the contents of a disk from another SCSI device. The complete contents of the source disk are copied, even if it is mostly free space, so the destination device must have space for the entire size of the virtual disk.

`-E    --exportraw <dstFile>`
Export the contents of the specified file on the specified SCSI device in raw form to a file on the file system of the console operating system. This command differs from `exportfile` in that it copies the source file exactly, rather than creating a virtual disk. Since the console operating system has a maximum file size of 2GB, this command is not useful for large disk images; use `exportfile` instead. However, `exportraw` is useful for distributing incremental updates to a disk image. If the disk image is used in undoable or append mode, then a redo log file is created. The name of that file is derived by appending `.REDO` to the name of the associated disk image file. The redo log contains the incremental changes to the disk image. The contents of the redo log can be copied to the file system of the console operating system using the exportraw command. The redo log can then be transported to a remote site and copied to the SCSI disk that contains a copy of the original disk image with the `importraw` command. The contents of the redo log can then be merged into the copy of the disk image using the commit command.

`-I    --importraw <srcFile>`
Import the exact contents of the specified file on the console operating system to the specified file on the specified SCSI device. This command differs from `importfile` in that it copies the source file exactly. As explained in the discussion of `exportraw` above, the combination of `exportraw` and `importraw` is useful for distributing incremental updates to a disk image.

`-l    --list`
List the files on the file system on the specified device, including their sizes.

`-r    --removefile`
Remove the specified file from the file system on the specified device.

`-m    --commit`
Commit the redo log of the specified file, making the associated changes permanent. The redo log is created when a file is used in undoable mode or append mode via a virtual machine. The name of the redo log is derived by appending `.REDO` to the name of the file that contains the base disk image. The changes to the disk that are stored in the redo log can either be committed using the commit option or eliminated by removing the redo file using the remove option.

`-S    --setfsname <fsName>`
Set the name of the VMFS file system on the specified SCSI device to `<fsName>`. This name can then be used to specify a VMFS file in subsequent vmkfstools commands or

in a virtual machine configuration file. The name will also appear in a listing produced by `vmkfstools -l`.

Examples

```
vmkfstools -C -b 2m -n 32 vmhba1:3:0:1
```

Create a new file system on the first partition of target 3, LUN 0 of SCSI adapter 1. The file block size is 2MB, and the maximum number of files is 32.

```
vmkfstools -S mydisk vmhba1:3:0:1
```

Give the name of `mydisk` to the new file system.

```
vmkfstools -c 2000m mydisk:rh6.2.dsk
```

Create a 2GB VMFS file with the name of `rh6.2.dsk` on the VMFS file system named `mydisk`. This file represents an empty disk and may be accessed by a virtual machine.

```
vmkfstools -r vmhba0:2:0:1:file2
```

Remove the file named `file2` in the file system on target 2, partition 1 of SCSI adapter 0.

```
vmkfstools -i ~/vms/nt4.dsk vmhba0:2:0:0:nt4.dsk
```

Copy the contents of a virtual disk (that contains Windows NT 4.0) from the host file system to a file named `nt4.dsk` on target 2 of SCSI adapter 0. A virtual machine can be configured to use this virtual disk by adding lines to its configuration file in the following format:

```
    scsi0.virtualDev = vmxbuslogic
    scsi0:0.present = TRUE
    scsi0:0.name = scsi0:2:0:0:nt4.dsk
```

```
vmkfstools -l vmhba0:2:0:0
```

List the contents of the file system on target 2 of SCSI adapter 0.

Naming VMFS File Systems

If you create a VMFS file system on a SCSI disk or partition, you can give a name to that file system and use that name when specifying VMFS files on that file system. For instance, suppose you have a VMFS file system on the SCSI partition `vmhba0:3:0:1` and have created a VMFS file `nt4.dsk`. You can name that file system via a `vmkfstools` commands such as:

```
    vmkfstools -S mydisk vmhba0:3:0:1
```

You can then refer to the `nt4.dsk` file as `mydisk:nt4.dsk` (instead of `vmhba0:3:0:1:nt4.dsk`) in a virtual machine configuration file and in other `vmkfstools` commands. Naming VMFS file systems is especially useful if you may be adding SCSI adapters or disks to your system, in which case the actual disk and target numbers specifying a particular VMFS may change, but the name will stay the same.

Mounting VMFS File Systems on the Console Operating System

VMFS file systems are automatically mounted in the `/vmfs` directory on the console operating system when the VMkernel is loaded during boot-up. The `mount-vmfs` script may be used manually to mount new VMFS file systems. The reverse operation (unmounting all VMFS partitions) can be performed by executing the `umount-vmfs` script.

Although mounted VMFS file systems may appear similar to any other file system such as ext2, VMFS is only intended to store large files such as disk images. Unfortunately, the console operating system (which is based on a Linux 2.2 kernel) does not currently support files greater than 2GB. NFS and `scp` are known to run into this limitation, while FTP and `cp` are not affected by it. Thus, you should use FTP and `cp` for copying files to and from a VMFS file system.

For more information, see Utility to Mount VMFS File Systems on page 108.

Utility to Mount VMFS File Systems

`mount-vmfs` is a program that mounts all VMFS (VMware ESX Server File System) file systems. It is useful for automatically mounting all partitions with valid VMFS file systems on the console operating system.

`mount-vmfs` does not take any arguments. It checks every SCSI device available to virtual machines for valid file systems. If a valid file system is found, `mount-vmfs` will mount it at `/vmfs/vmhba<a>:<t>:<l>:<p>`, where `<a>` specifies the SCSI adapter number, `<t>` specifies the SCSI target, `<l>` specifies the LUN (logical unit number) and `<p>` specifies the disk partition. If the disk has no partitions and the disk has a valid file system, `<p>` will be zero.

If a partition has an associated file system name (`vmkfstools -S`), then `mount-vmfs` will also create a symbolic link from `/vmfs/<fsname>` to the corresponding mount point (`/vmfs/vmhba<a>:<t>:<l>:<p>`).

The reverse operation — unmounting all VMFS partitions — can be performed by executing `umount-vmfs` script.

`mount-vmfs` calls the regular mount command to actually mount the disks. The file system type is vmfs and the device name is obtained by calling `vmkfstools -N vmhba<a>:<t>:<l>:<p>`. For example, `mount-vmfs` would call `mount -t vmfs 'vmkfstools -N vmhba0:1:0:2' /vmfs/vmhba0:1:0:2` in order to mount partition 2 of the disk with target 1 on the adapter `vmhba0`.

Although VMFS file systems may appear similar to any other file system such as ext2, VMFS is only meant to store large files such as disk images. It does not support directory hierarchies. New file systems can be created using `vmkfstools -C`.

The reported file length of all VMFS files (disk images) is 512 bytes longer than the disk image. The additional 512 bytes contain certain file attributes such as the size of the disk image represented by the file. VMFS files that are not disk images do not incur this 512-byte overhead.

Limitations

Disk images tend to be large. Unfortunately, the console operating system does not support files greater than 4GB, and there is only limited functionality for files between 2GB and 4GB. The file size field of the `stat` system call has only 32 bits, therefore `stat` will return incorrect information for files equal to or bigger than 4GB. For such files, VMFS returns 4GB-1 as the file size in the `stat` system call. NFS and `scp` are known to run into this limitation, while FTP and `cp` are not affected by it. We provide a modified `ls` binary that uses a special interface into VMFS to report the correct file size.

Currently, VMFS does not support flexible file permissions. All files are owned and writable by root and readable by other users. Also, VMFS file names are currently limited to 128 bytes.

For further information, see File System Management on SCSI Disks and RAID on page 104.

Determining SCSI Target IDs

In order to assign SCSI drives to a virtual machine, you need to know which controller the drive is on and what the SCSI target ID of the controller is. This section can help you determine these values without opening your computer and physically looking at the SCSI target ID settings on the drives.

On a standard Linux system, or for a VMware Console Operating System that has SCSI controllers assigned to the console operating system rather than the VMkernel, information on attached SCSI devices, including SCSI target IDs is available in the boot log (usually `/var/log/messages`), or from examining `/proc/scsi/scsi`.

Information about the SCSI controllers assigned to the VMkernel and about the devices attached to these controllers is available in the `/proc/vmware/scsi` directory once the VMkernel and the VMkernel device module(s) for the SCSI controller(s) have been loaded.

Each entry in the `/proc/vmware/scsi` directory corresponds to a SCSI controller assigned to the VMkernel. For example, if you issued a `vmkload_mod` command with the base name `vmhba` and a single SCSI controller was found, you would see this:

```
# ls -l /proc/vmware/scsi
total 0
dr-xr-xr-x   2 root    root       0 Jun 22 12:44 vmhba0
```

Each SCSI controller's subdirectory contains entries for the SCSI devices on that controller, numbered by SCSI target ID and LUN (logical unit number). Run `cat` on each target ID:LUN pair to get information about the device with that target ID and LUN. For example:

```
# cat /proc/vmware/scsi/vmhba0/1:0
Vendor: SEAGATE   Model: ST39103LW         Rev: 0002
Type:   Direct-Access                      ANSI SCSI
revision: 02
Size:   8683 Mbytes
Queue Depth: 28

Partition Info:
Block size: 512
Num Blocks: 17783240

     num:    Start      Size     Type
     4:          1  17526914       fb
```

```
    Partition 0:
        VM                  11
        Commands            2
        Kbytes read         0
        Kbytes written      0
        Commands aborted    0
        Bus resets          0
    Partition 4:
        Commands            336
        Kbytes read         857
        Kbytes written      488
        Commands aborted    0
        Bus resets          0
```

This information should help you determine the SCSI target ID to use in the virtual machine configuration file, as detailed in Configuring Virtual Machines on VMware ESX Server on page 92.

8

Reference: Memory

How the System Uses Memory

If you are planning to deploy virtual machines on physical servers, you need to know how much memory to install in a server to support the virtual machines that will be running there.

This note describes how to account for the memory sizes of VMware ESX Server components and the memory required for virtual machines. As you will see, you must allow a certain amount of memory for overhead. However, it is also possible to “overcommit” the physical memory in your system — taking advantage of the fact that for much of the time it is running, a virtual machine is likely to use only part of the memory allocated to it.

Overhead

Memory allocated to the console operating system is not available for other uses. The recommended size for the console operating system in a default configuration is 80MB. This is an appropriate size for up to four virtual machines.

Memory dedicated to the VMkernel is not available for other uses. In the current release, the VMkernel consumes approximately 8MB.

Each virtual machine requires additional memory for virtualization, the frame buffer and various other overhead uses. In the current release, this overhead is 32MB per virtual machine.

Example: A Single Virtual Machine

Suppose a system has 512MB of physical RAM. Subtract the fixed overhead for both the console operating system (80MB) and the VMkernel (8MB). This leaves 424MB for running virtual machines. Subtract the overhead of 32MB per virtual machine, and the maximum size for a new virtual machine would be 392MB.

Dynamic Memory Allocation and Overcommitment

VMware ESX Server provides dynamic control over the amount of physical memory allocated to each virtual machine. Memory may be overcommitted, if you wish, so that the total size configured for all running virtual machines exceeds the total amount of available physical memory.

To enable overcommitment and dynamic control over virtual machine sizes, ESX Server provides support for expanding or contracting the amount of memory allocated to running virtual machines. A VMware-supplied `vmmemctl` driver module must be loaded into the guest operating system running in each virtual machine to support dynamic memory allocation. Drivers are currently provided for Windows NT, Windows 2000 and Linux guests. They are automatically installed as part of the VMware Tools installation procedure.

Three basic parameters control the allocation of memory to each virtual machine:

- Its minimum size — min
- Its maximum size — max
- Its shares allocation

A virtual machine's use of physical memory is always bounded by its configured minimum and maximum sizes, regardless of whether or not `vmmemctl` is installed and running. The system automatically allocates memory for each virtual machine based on two factors: the number of shares it has been given and an estimate of its recent working set size.

Even when memory is overcommitted, each virtual machine is guaranteed to receive an amount of physical memory at least as large as its specified minimum size. The maximum size for a virtual machine must also be specified in its configuration file. The maximum size is the amount of memory configured for use by the guest operating system running in the virtual machine. By default, virtual machines operate at their maximum allocation, unless memory is overcommitted. The system limits the maximum size of a virtual machine based on its minimum size and the systemwide `MemMaxOvercommit` parameter. With the default maximum overcommitment level of 100 percent, the maximum size may be no greater than twice the minimum size.

For additional details, please see VMware ESX Server Memory Resource Management on page 137.

Example: Multiple Virtual Machines

Suppose a system has 512MB of physical RAM. Subtract the fixed overhead for both the console operating system (80MB) and the VMkernel (8MB). This leaves 424MB for running virtual machines. Account for the 32MB overhead per virtual machine, and the maximum size for a single new virtual machine would be 392MB.

Suppose that a 256MB virtual machine named A is started. A's maximum size is set to 256MB. Unless otherwise configured, its minimum size defaults to half of its maximum size, or 128MB.

If virtual machine A has not yet started its `vmmemctl` driver — because it is still booting or because VMware Tools has not been installed — the maximum memory available for starting additional virtual machines is the original 424MB minus the memory used by A (256MB + 32MB overhead = 288MB). This leaves 136MB available for running additional virtual machines. Accounting for the 32MB overhead for a virtual machine, the maximum size for a new virtual machine would be 104MB.

However, once virtual machine A has booted and its `vmmemctl` driver has started, the system is able to dynamically reclaim 128MB from A. This changes the total memory available for running additional virtual machines to 136MB + 128MB = 264MB, or 232MB after adjusting for the overhead per virtual machine.

Starting a new virtual machine B that is larger than 104MB would be overcommitting physical memory. Assuming that B also starts its `vmmemctl` driver, the system will automatically allocate memory between the two virtual machines dynamically, based on their memory share allocations and an estimate of their recent working set sizes.

Finally, suppose that a 202MB virtual machine B is started. The total memory available for running additional virtual machines then becomes the previous total of 264MB minus the memory used by B (202MB + 32MB overhead = 234MB), leaving 30MB available for running additional virtual machines. Because this is less than the 32MB overhead per virtual machine, no additional virtual machines can be started.

However, once B has booted and its `vmmemctl` driver has started, the system is able to dynamically reclaim 101MB from B. This changes the total memory available for running additional virtual machines to 32MB + 101MB = 133MB, or 101MB after adjusting for the overhead per virtual machine.

The current release also supports an experimental `MemRelaxAdmit` option that can be used to reduce the amount of reserved memory required to start a new virtual machine. See this technical note for more details. Additional information on `MemRelaxAdmit` is available on page 141.

Querying System Information

The computations described in the preceding section are performed automatically and may be viewed by reading the procfs node `/proc/vmware/mem` on the console operating system.

```
cat /proc/vmware/mem
```

In particular, this report lists the maximum size for a new virtual machine as Maximum new VM size. The current release does not report this information via the Web-based management interface.

To see more detailed information, including the current allocations for all running virtual machines, read the procfs node `/proc/vmware/sched/mem` on the console operating system.

```
cat /proc/vmware/sched/mem
```

In addition, the Web-based management interface displays a useful subset of this information on the Monitor Resources page.

Dynamic Memory Management

VMware ESX Server uses the vmmemctl module to support dynamic memory resource management. This note provides background on how vmmemctl works and describes limitations of the feature in the current release.

Overview

VMware ESX Server provides dynamic control over the amount of physical memory allocated to each virtual machine. Memory may be overcommitted, at the discretion of the administrator, so that the total size configured for all running virtual machines exceeds the total amount of available physical memory.

To enable memory overcommitment and dynamic control over virtual machine sizes, VMware ESX Server provides support for expanding or contracting the amount of memory allocated to each running virtual machine. A VMware-supplied vmmemctl driver module must be loaded into the guest operating system running in each virtual machine to support dynamic memory allocation. Drivers are currently provided for Windows NT, Windows 2000 and Linux guests. They are automatically installed as part of the VMware Tools installation procedure and loaded automatically when VMware Tools starts.

The vmmemctl driver cooperates with the server to reclaim those pages of memory that are considered least valuable by the guest operating system. This proprietary technique has several advantages. It provides predictable performance that closely matches the behavior of a native system under similar memory constraints. Any paging or swapping that may be required is performed directly by the guest operating system to its own virtual disk storage, using its own native memory management algorithms.

Limitations and Issues

If the vmmemctl driver is not installed or not running in a virtual machine, VMware ESX Server will be unable to control that virtual machine's size dynamically. Such a virtual machine will always consume its maximum configured memory size, regardless of its configured minimum size or memory shares parameters. There are two situations where this might occur.

Guest Operating System Boot or Reboot

When the guest operating system is booting, its vmmemctl driver has not yet been loaded. Since booting is not memory-intensive, one might reasonably expect that the guest would not exceed its configured minimum memory size during the boot process. However, some operating systems, such as Windows 2000, touch all of memory while booting. Once the boot process has completed, the vmmemctl driver can start reclaiming memory immediately, if necessary.

For this reason, the current release refuses to start a virtual machine if there is insufficient memory available initially to allow it to reserve its configured maximum size (which may be up to three times as large as its minimum size). This effectively reduces the maximum level of memory overcommitment, although it is not very restrictive for systems running several virtual machines.

See also the ESX Server MemZeroCompress (page 141) and MemRelaxAdmit (page 141) configuration options, which may be enabled to avoid these limitations.

No vmmemctl Driver

If the VMware Tools installation is never performed, then the vmmemctl driver will not be installed in the guest operating system. Similarly, a malicious user with root or Administrator access to a guest operating system could delete or otherwise disable an installed vmmemctl driver, although once the driver is started, it cannot be unloaded without rebooting the guest operating system.

Note that, in any case, a virtual machine's use of physical memory is always bounded by its configured minimum and maximum sizes, regardless of whether vmmemctl is installed and running. However, it is possible for a virtual machine that is not running vmmemctl to use more than its "fair share" of memory in an overcommitted system, since the server is unable to reduce that virtual machine's memory consumption below its configured maximum size.

The current release logs a warning for each virtual machine that has not started running vmmemctl after a specified time. This timeout interval can be changed dynamically via the MemDriverTimeout configuration option.

Future releases may allow ESX Server administrators to specify what action to take if no vmmemctl driver is running, in addition to logging a warning. Possible actions include suspending the virtual machine to disk or modifying its configuration file to automatically reduce its maximum memory size the next time it is powered on.

9

Reference: Networking

Setting the MAC Address Manually for a Virtual Machine

VMware ESX Server automatically generates MAC addresses for the virtual network adapters in each virtual machine. In most cases, these MAC addresses will be appropriate. However, there may be times when you need to set a virtual network adapter's MAC address manually — for example:

- You have more than 256 virtual network adapters on a single physical server.
- Virtual network adapters on different physical servers share the same subnet and are assigned the same MAC address, causing a conflict.
- You want to ensure that a virtual network adapter will always have the same MAC address.

This document explains how VMware ESX Server generates MAC addresses and how you can set the MAC address for a virtual network adapter manually.

How VMware ESX Server Generates MAC Addresses

Each virtual network adapter in a virtual machine gets its own unique MAC address. ESX Server attempts to ensure that the network adapters for each virtual machine that are on the same subnet have unique MAC addresses. The algorithm used by ESX Server puts a limit on how many virtual machines can be running and suspended at once on a given machine. It also does not handle all cases when virtual machines on distinct machines share a subnet.

A MAC address is a six-byte number. Each network adapter manufacturer gets a unique three-byte prefix called an OUI — organizationally unique identifier — that it can use to generate unique MAC addresses. VMware has OUIs — one for automatically generated MAC addresses and one for manually set addresses. The VMware OUI for generated MAC addresses is 0x00:0x50:0x56. Thus the first three bytes of the MAC address that is automatically generated for each virtual network adapter will have this value. ESX Server then uses a MAC address generation algorithm to produce the other three bytes. The algorithm guarantees unique MAC addresses within a machine and attempts to provide unique MAC addresses between ESX Server machines.

The algorithm that ESX Server uses is the following:

When the algorithm generates the last 24 bits of the MAC address, the first 16 bits are set to the same values as the last 16 bits of the console operating system's primary IP address.

The final eight bits of the MAC address are set to a hash value based on the name of the virtual machine's configuration file.

ESX Server keeps track of all MAC addresses that have been assigned to network adapters of running and suspended virtual machines on a given physical machine. ESX Server ensures that the virtual network adapters of all of these virtual machines will have unique MAC addresses.

The MAC address of a powered-off virtual machine is not remembered. Thus it is possible that when a virtual machine is powered on again it can get a different MAC address.

For example, if a machine had IP address 192.34.14.81 (or in hex, 0xc0.0x22.0x0e.0x51) and the configuration file hashed to the value 0x95 the MAC address would have the following value:

0x00:0x50:0x56:0x0e:0x51:0x95

Since there are only eight bits that can vary for each MAC address on an ESX Server machine, this puts a limit of 256 unique MAC addresses per ESX Server machine. This in turn limits the total number of virtual network adapters in all powered-on and suspended virtual machines to 256. This limitation can be eliminated by using the method described in the section Setting MAC Addresses Manually (below).

Note: The use of parts of the console operating system's IP address as part of the MAC address is an attempt to generate MAC addresses that are unique across different ESX Server machines. However, there is no guarantee that different ESX machines with physical network adapters that share a subnet will generate mutually exclusive MAC addresses.

Setting MAC Addresses Manually

In order to work around both the limit of 256 virtual network adapters per physical machine and possible MAC address conflicts between virtual machines, the MAC addresses can be assigned manually by system administrators. VMware uses a different OUI for manually generated addresses: 0x00:0x50:0x56. The addresses can be set by adding the following line to a virtual machine's configuration file:

```
ethernet0.address = 00:50:56:XX:YY:ZZ
```

where `XX` is a valid hex number between 00h and 3Fh and `YY` and `ZZ` are valid hex numbers between 00h and FFh. The value for `XX` must not be greater than 0x3F in order to avoid conflict with MAC addresses that are generated by the VMware Workstation and VMware GSX Server products. Thus the maximum value for a manually generated MAC address is

```
    ethernet0.address = 00:50:56:3F:FF:FF
```

VMware ESX Server virtual machines do not support arbitrary MAC addresses, hence the above format must be used. So long as you choose `XX:YY:ZZ` so it is unique among your hard-coded addresses, conflicts between the automatically assigned MAC addresses and the manually assigned ones should never occur.

The VMkernel Network Card Locator

When network interface cards are assigned to the VMkernel, sometimes it is difficult to map from the name of the VMkernel device to the physical network adapter on the machine.

For example, if there are four Intel EEPro cards in a machine and all are dedicated to the VMkernel, these four cards will end up being called `vmnic0`, `vmnic1`, `vmnic2` and `vmnic3`. The name of a card is based on its order in the PCI bus/slot hierarchy on the machine — the lower the bus and slot, the lower the number at the end of the name.

If you know the bus and slot order of the adapters, you can figure out which adapter has which name. However, if you don’t, you can use the `findnic` program to help you make the proper association of network adapter to name.

The format of the command is

```
    findnic <options> <nic-name> <local-ip> <remote-ip>
```

The `findnic` program takes a VMkernel network device name, an IP address to give the device on the local machine and an IP address that `findnic` should try to ping. When you issue the command, `findnic` will ping the remote IP address.

This will allow you to determine which adapter is which by looking at the LEDs on the cards to see which one is flashing or by seeing if the ping itself is successful.

Options

`-f`
Do a flood ping.

`-i <seconds>`
Interval in seconds between pings.

Examples

```
findnic vmnic0 10.2.0.5 10.2.0.4
```

Binds VMkernel device vmnic0 to IP address 10.2.0.5 and then tries to ping the remote machine with the IP address 10.2.0.4.

```
findnic -f vmnic1 10.2.0.5 10.2.0.4
```

Binds VMkernel device vmnic1 to IP address 10.2.0.5 and then tries to flood ping the remote machine with the IP address 10.2.0.4.

Sharing Network Adapters and Virtual Networks

In many ESX Server configurations, there will be a clear distinction between networking resources used by the virtual machines and those used by the console operating system. This may be important for security reasons, for example — isolating the management network from the network used by applications in the virtual machines.

However, there may be times when you want to share resources, including physical network adapters and virtual networks.

This technical note provides instructions on sharing in both directions — making the virtual machines' resources available to the console operating system and allowing virtual machines to share the network adapter used by the console operating system.

This sharing is made possible by the `vmxnet_console` driver, which is installed with the console operating system.

We recommend that only advanced users make these configuration changes. The steps below will be easier for someone who is familiar with administering a Linux system.

Note: If you accidentally bring down the local loopback interface while you are reconfiguring network devices, the Web-based management interface will not function properly. To bring it back up, use the command `ifconfig lo up`.

Allowing the Console Operating System to Use the Virtual Machines' Devices

All network adapters used by virtual machines (that is, assigned to the VMkernel) and virtual networks can be made accessible to the console operating system. Virtual networks — identified as `vmnet_<n>` on the Edit Configuration page of the Web-based management interface — provide high-speed connections among virtual machines on the same physical server.

To give the console operating system access to VMkernel network adapters and virtual networks, you must install the `vmxnet_console` module. When you install it, you provide a list of VMkernel network adapters and virtual networks that the `vmxnet_console` module should attach to. For example, if the VMkernel had an adapter named `vmnic1` and a virtual network named `vnet_0`, and you wanted to provide access to them from the console operating system, you would use the following command to install the `vmxnet_console` module.

```
    insmod vmxnet_console devName=vmnic1,vmnet_0
```

The devName parameter is a comma-separated list of names of VMkernel network adapters and virtual networks.

When you install the module, it will add the appropriate number of eth<n> devices on the console operating system in the order that you list the VMkernel network adapter and virtual network names after the devName parameter. In the example above, if the console operating system already had a network adapter named eth0, when you load vmxnet_console with vmnic1 and vmnet_0, vmnic1 will be seen as eth1 on the console operating system and vmnet_0 will be seen as eth2.

Once the eth<n> devices are created on the console operating system, you can bring the interfaces up in the normal manner. For example, if you want the console operating system to use IP address 10.2.0.4 for the network accessed via the vmnic1 adapter, use the following command:

```
    ifconfig eth1 up 10.2.0.4
```

If you want an easy way to see which eth<n> devices are added by the insmod command, you can add the tagName parameter to the insmod command, as shown in this example:

```
    insmod vmxnet_console devName=vmnic1,vmnet0 tagName=<tag>
```

In this case the vmxnet_console module will add the names of each of the eth<n> devices that it created to /var/log/messages. Each message will begin with the string <tag>. To figure out the names of the devices that were added, use this command:

```
    grep <tag> /var/log/messages
```

Starting Shared VMkernel Network Adapters and Virtual Networks when the Console Operating System Boots

There are two ways you can configure the console operating system to start VMkernel network adapters when the console operating system boots. The simpler case involves sharing a network adapter other than eth0. Sharing eth0 is more complicated and is described later.

Continuing with the example from the previous section, you can append the following lines to /etc/rc.d/rc.local:

```
    insmod vmxnet_console devName=vmnic1,vmnet0
    ifconfig eth1 up 10.2.0.4
    ifconfig eth2 up 63.93.12.47
```

Another method is to set up the files `/etc/sysconfig/network-scripts/ifcfg-eth1` and `/etc/sysconfig/network-scripts/ifcfg-eth2` with the appropriate network information. And be sure the `ONBOOT=` line is `ONBOOT=yes`. The `ifcfg-eth1` file for this example would be

```
DEVICE=eth1
BOOTPROTO=static
BROADCAST=10.255.255.255
IPADDR=10.2.0.4
NETMASK=255.0.0.0
NETWORK=10.0.0.0
ONBOOT=yes
```

In this case, the lines you add to `/etc/rc.d/rc.local` would be:

```
insmod vmxnet_console devName=vmnic1,vmnet0
ifup eth1
ifup eth2
```

Sharing the Console Operating System's Network Adapter with Virtual Machines

If you intend to share the adapter that is `eth0` on the console operating system, be careful as you implement the following steps. In order to configure ESX Server initally, you need to have a network connection. Once the initial configuration is set, you will make several changes. At one point in the process, there will be no network connection to the console operating system, and you will need to work directly at the server.

When you first install and configure ESX Server, the VMkernel is not loaded, so the console operating system needs to control the network adapter that is `eth0`. When you configure ESX Server, assign the adapter that is `eth0` to the console operating system.

Once you have completely configured ESX Server properly and rebooted, the VMkernel will be loaded. At that point, you need to take the following steps:

1. Edit `/etc/conf.modules` and comment out the line that refers to `alias eth0`.

   If the original line is
   ```
   alias eth0 e100
   ```
   edit it to be
   ```
   # alias eth0 e100
   ```

   This will disable `eth0` on the console operating system when it boots.

2. Use the Web-based management interface to reconfigure the server. Log in as root and go to `http://<hostname>:8222/pcidivy`, then click the Edit link for the configuration you want to change. Find the table row that lists the Ethernet controller assigned to the console and click the radio button in the Virtual Machine column to reassign it.

   Click Save Configuration, then reboot the machine when prompted.

3. When the machine reboots, no network adapter will be assigned to the console operating system, so you must do this step at the server.

   Add the appropriate lines to `/etc/rc.d/rc.local`. For example, if `eth0` is the only network adapter that you intend to share between the VMkernel and the console operating system, and if it will be named `vmnic0` in the VMkernel, you would add the lines

   ```
   insmod vmxnet_console devName=vmnic0
   ifup eth0
   ```

   If you are unsure what name the VMkernel has assigned to the network adapter that formerly was `eth0` in the console operating system, you can determine its name using the `findnic` program (see page 125).

4. The next time you reboot the system, the network adapter will be shared by the console operating system and the virtual machines.

   To begin sharing the network adapter without rebooting the system, you can manually issue the same commands you added to `/etc/rc.d/rc.local`.

   ```
   insmod vmxnet_console devName=vmnic0
   ifup eth0
   ```

10

Reference: Resource Management

CPU Resource Management

VMware ESX Server provides dynamic control over both the execution rate and the processor assignment of each scheduled virtual machine. The scheduler performs automatic load balancing on multiprocessor systems.

You can manage the CPU resources on a server from the Web-based management interface or from the console operating system's command line.

Proportional-share processor scheduling enables intuitive control over execution rates. Each scheduled virtual machine is allocated a number of shares that entitle it to a fraction of processor resources. For example, a virtual machine that is allocated twice as many shares as another is entitled to consume twice as many CPU cycles. In general, a runnable virtual machine with *S* shares on a processor with an overall total of *T* shares is guaranteed to receive at least a fraction *S/T* of the processor CPU time.

For example, if you are running three virtual machines, each will start with a default allocation of 1,000 shares. If you want to give one virtual machine half the CPU time and give each of the other two virtual machines one-quarter of the CPU time, you can assign 2,000 shares to the first virtual machine and leave the other two at their default allocations. Since these share allocations are relative, the same effect may be achieved by giving 500 shares to the first virtual machine and 250 to each of the other two virtual machines.

An administrator can control relative CPU rates by specifying the number of shares allocated to each virtual machine. The system automatically keeps track of the total number of shares *T*. Increasing the number of shares allocated to a virtual machine will dilute the effective value of all shares by increasing *T*. Absolute guarantees for minimum CPU rates can be specified by following the simple convention of limiting the total number of shares allocated across all virtual machines. For example, if the total number of shares is limited to 10,000 or less, each share represents a guaranteed minimum of at least 0.01 percent of processor CPU cycles.

The console operating receives 1,000 shares by default. In most cases, this should be an appropriate allocation, since the console operating system should not be used for CPU-intensive tasks. If you do find it necessary to adjust the console operating system's allocation of CPU shares, you can use the procfs interface, as described in this section. Or you can achieve a similar result indirectly, using the Web-based management interface, by adjusting the shares of the virtual machines running on the server so the console operating system's 1,000 shares represent a greater or smaller proportion of the total.

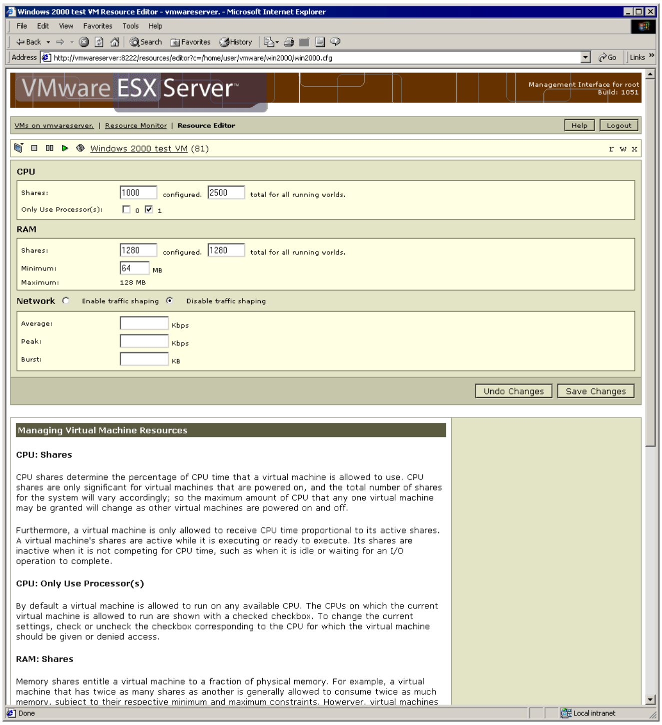
Shares are not hard partitions or reservations, so underutilized allocations are not wasted. Instead, inactive shares are effectively removed from consideration, allowing active virtual machines to benefit when extra resources are available.

Multiprocessor Systems

In multiprocessor systems, an administrator can also restrict the assignment of virtual machines to a subset of the available processors by specifying an affinity set for each virtual machine. The system will automatically assign each virtual machine to a processor in the specified affinity set in order to balance the number of active shares across processors. If the affinity set contains only a single processor, then the virtual machine will be placed there.

Any one virtual machine will be assigned to only one processor. And the guest operating system will see a virtual machine with a single processor.

The current release allows CPU shares and affinity sets to be specified and modified dynamically at any time using a simple procfs interface. Initial values for a virtual machine may also be specified in its configuration file.


Windows 2000 test VM Resource Editor - vmwareserver. - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://vmwareserver:8222/resources/editor?c=/home/user/vmware/win2000/win2000.cfg
Go Links
VMware ESX Server™
Management Interface for root
Build: 1051
VMs on vmwareserver. | Resource Monitor | Resource Editor
Help Logout
Windows 2000 test VM (81)
r w x
Undo Changes
Save Changes
Done
Local intranet

CPU

Shares: 1000 configured. 2500 total for all running worlds.

Only Use Processor(s): □ 0 ☑ 1

RAM

Shares: 1280 configured. 1280 total for all running worlds.

Minimum: 64 MB

Maximum: 128 MB

Network ○ Enable traffic shaping ● Disable traffic shaping

Average: Kbps

Peak: Kbps

Burst: KB

Managing Virtual Machine Resources

CPU: Shares

CPU shares determine the percentage of CPU time that a virtual machine is allowed to use. CPU shares are only significant for virtual machines that are powered on, and the total number of shares for the system will vary accordingly; so the maximum amount of CPU that any one virtual machine may be granted will change as other virtual machines are powered on and off.

Furthermore, a virtual machine is only allowed to receive CPU time proportional to its active shares. A virtual machine's shares are active while it is executing or ready to execute. Its shares are inactive when it is not competing for CPU time, such as when it is idle or waiting for an I/O operation to complete.

CPU: Only Use Processor(s)

By default a virtual machine is allowed to run on any available CPU. The CPUs on which the current virtual machine is allowed to run are shown with a checked checkbox. To change the current settings, check or uncheck the checkbox corresponding to the CPU for which the virtual machine should be given or denied access.

RAM: Shares

Memory shares entitle a virtual machine to a fraction of physical memory. For example, a virtual machine that has twice as many shares as another is generally allowed to consume twice as much memory, subject to their respective minimum and maximum constraints. Howerver, virtual machines

Settings may also be changed from the Resource Editor page of the Web-based management interface. On the server's Overview page, click Manage Resources. The Resource Monitor page appears. Click the link under the name of the virtual machine for which you want to change settings. Enter the desired settings, then click Save Changes.

You must log in as root in order to change resource management settings using either the Web-based interface or procfs.

How It Works

`sched.cpu.shares = <nshares>`

This configuration file option specifies the initial share allocation for a virtual machine to `<nshares>` shares. The valid range of values for `<nshares>` is 1 to 100000, enabling a large range of allocation ratios. The default allocation is 1,000 shares.

`sched.cpu.affinity = <set>`

This configuration file option specifies the initial processor affinity set for a virtual machine. If <set> is all or default, then the affinity set contains all available processors. The specified set may altenatively be a comma-separated list of CPU numbers such as 0,2,3.

/proc/vmware/vm/<id>/cpu/shares
Reading from this file reports the number of shares allocated to the virtual machine identified by <id>.

Writing a number <nshares> to this file changes the number of shares allocated to the virtual machine identified by <id> to <nshares>. The valid range of values for <nshares> is 1 to 100000.

/proc/vmware/vm/<id>/cpu/affinity
Reading from this file reports the number of each CPU in the current affinity set for the virtual machine identified by <id>.

Writing a comma-separated list of CPU numbers to this file, such as 0,2,3, changes the affinity set for the virtual machine identified by <id>. Writing all or default to this file changes the affinity set to contain all available processors.

/proc/vmware/vm/<id>/cpu/status
Reading from this file reports current status information for the virtual machine identified by <id>, including the specified shares and affinity parameters, as well as the virtual machine name, state (running, ready, waiting), current CPU assignment and cumulative CPU usage in seconds.

/proc/vmware/sched/cpu.<n>
Reading from this file reports the status information for all active virtual machines currently assigned to cpu number <n>, as well as some aggregate totals.

/proc/vmware/sched/cpu
Reading from this file reports the status information for all virtual machines in the entire system.

/proc/vmware/config/CpuBalancePeriod
This ESX Server option specifies the periodic time interval, in seconds, for automatic multiprocessor load balancing based on active shares. Defaults to 1 second.

Examples

Suppose that we are interested in the CPU allocation for the virtual machine with ID 103. To query the number of shares allocated to virtual machine 103, simply read the file:

```
% cat /proc/vmware/vm/103/cpu/shares
1000
```

This indicates that virtual machine 103 is currently allocated 1,000 shares. To change the number of shares allocated to virtual machine 103, simply write to the file. Note that you need root privileges in order to change share allocations:

```
# echo 2000 > /proc/vmware/vm/103/cpu/shares
```

The change can be confirmed by reading the file again:

```
% cat /proc/vmware/vm/103/cpu/shares
2000
```

To query the affinity set for virtual machine 103, simply read the file:

```
% cat /proc/vmware/vm/103/cpu/affinity
0,1
```

This indicates that virtual machine 103 is allowed to run on CPUs 0 and 1. To restrict virtual machine 103 to run only on CPU 1, simply write to the file. Note that you need root privileges in order to change affinity sets:

```
# echo 1 > /proc/vmware/vm/103/cpu/affinity
```

The change can be confirmed by reading the file again.

Cautions

CPU share allocations do not necessarily guarantee the rate of progress within a virtual machine. For example, suppose virtual machine 103 is allocated 2,000 shares, while virtual machine 104 is allocated 1,000 shares. If both virtual machines are CPU-bound — for example, both are running the same compute-intensive benchmark — then virtual machine 103 should indeed run twice as fast as virtual machine 104. However, if virtual machine 103 instead runs an I/O-bound workload that causes it to stop as it waits for other resources, it will not run twice as fast as virtual machine 103, even though it is allowed to use twice as much CPU time.

Memory Resource Management

VMware ESX Server provides dynamic control over the amount of physical memory allocated to each virtual machine. Memory may be overcommitted, if you wish, so that the total size configured for all running virtual machines exceeds the total amount of available physical memory. Three basic parameters control the allocation of memory to each virtual machine: its minimum size, its maximum size and its shares allocation.

You can manage the memory resources on a server from the Web-based management interface or from the console operating system's command line.

Static Partitioning

To statically partition physical memory across virtual machines, simply specify the maximum size of each virtual machine exactly. The maximum size is the amount of memory configured for use by the guest operating system running in the virtual machine, and must be specified as `memsize` in its configuration file. For such manual allocations, no other parameters need to be set.

Flexible Partitioning

VMware ESX Server also supports flexible partitioning of memory across virtual machines. This is useful when the total number of virtual machines or the memory needed by each virtual machine for optimum performance varies over time. It also allows memory to be overcommitted, enabling more virtual machines to run than would be possible with a static partitioning.

To enable overcommitment and dynamic control over virtual machine sizes, support is provided for expanding or contracting the amount of memory allocated to running virtual machines. A VMware-supplied `vmmemctl` module must be loaded into the guest operating system running in each virtual machine that supports dynamic memory allocation. It is installed as part of the VMware Tools package and loaded automatically when VMware Tools starts.

The `vmmemctl` driver cooperates with the server to reclaim those pages that are considered least valuable by the guest operating system. This proprietary technique provides predictable performance that closely matches the behavior of a native system under similar memory constraints.

When flexible partitioning is used, it is important to specify the minimum size of each virtual machine carefully. Even when memory is overcommitted, each virtual machine is guaranteed to receive an amount of physical memory at least as large as its specified minimum size.

The system refuses to start a virtual machine if there is insufficient memory available to reserve its minimum size. System administrators should typically configure the minimum virtual machine memory size to a level that will allow the virtual machine to run without excessive swapping or thrashing. The guest operating system running in the virtual machine must also be configured with sufficient swap space to store its dynamically reclaimed memory.

The maximum size for a virtual machine must also be specified in its configuration file; it is the amount of memory configured for use by the guest operating system running in the virtual machine. By default, virtual machines operate at their maximum allocation unless memory is overcommitted. In general, it is reasonable to specify the maximum size for a virtual machine to be considerably larger than its minimum size, allowing it to exploit additional system memory that may be available.

The system limits the maximum size of a virtual machine based on its minimum size and the systemwide `MemMaxOvercommit` parameter. With the default maximum overcommitment level of 100 percent, the maximum size may be no greater than twice the minimum size. Unless the optional minimum size parameter is explicitly specified, it will be set automatically to a fraction of the required maximum size, based on the maximum overcommitment level. The current release limits the overall level of memory overcommitment to a factor of three.

When memory is overcommitted, each virtual machine will be allocated an amount of memory somewhere between its minimum and maximum sizes. The amount of memory granted to a virtual machine above its minimum size is referred to as its flex allocation, representing a flexible allocation that may vary with the current memory load. The system automatically determines flex allocations for each virtual machine based on two factors: the number of shares it has been given and an estimate of its recent working set size.

Shares entitle a virtual machine to a fraction of physical memory. For example, a virtual machine that has twice as many shares as another is generally entitled to consume twice as much memory, subject to their respective minimum and maximum constraints. However, virtual machines that are not actively using their currently-allocated memory will automatically have their effective number of shares reduced. This is achieved by charging a virtual machine more for an idle page than for one that it is actively using. This prevents an idle virtual machine from hoarding memory unless it has a very large number of shares.

The current release allows memory allocations to be specified and modified dynamically at any time using the Web-based management interface or a simple procfs interface on the console operating system. Initial values for a virtual machine may also be specified in its configuration file. Reasonable defaults are automatically used when parameters are not specified explicitly.

Using a Web browser, you may change settings from the Resource Editor page of the Web-based management interface. On the server's Overview page, click Manage Resources. The Resource Monitor page appears. Click the link under the name of the virtual machine for which you want to change settings. Enter the desired settings, then click Save Changes.

You must log in as root in order to change resource management settings using either the Web-based interface or procfs.

Allocating Memory

The console operating system commands to check or modify the memory allocation for a virtual machine use the formats shown below.

`memsize = <size>`
This configuration file option specifies the maximum virtual machine size to be `<size>`MB.

`sched.mem.minsize = <size>`
This configuration file option specifies the guaranteed minimum virtual machine size to be `<size>`MB. The maximum valid value for `<size>` is 100 percent of the specified maximum virtual machine size. The minimum valid value for `<size>` depends on the systemwide `MemMaxOvercommit` parameter. By default, the minimum valid value for `<size>` is 50 percent of the specified maximum virtual machine size.

`sched.mem.shares = <nshares>`
This configuration file option specifies the initial memory share allocation for a virtual machine to be `<nshares>` shares. The valid range of values for `<nshares>` is 0 to 100000, enabling a large range of allocation ratios. The default allocation is 10 times the maximum virtual machine size in megabytes.

`/proc/vmware/vm/<id>/mem/min`
Reading from this file reports the minimum memory size in megabytesfor the virtual machine identified by `<id>`.

Writing a number <size> to this file changes the minimum memory size for the virtual machine identified by `<id>` to `<size>`MB.

`/proc/vmware/vm/<id>/mem/shares`
Reading from this file reports the number of memory shares allocated to the virtual machine identified by `<id>`.

Writing a number `<nshares>` to this file changes the number of memory shares allocated to the virtual machine identified by `<id>` to `<nshares>`. The valid range of values for `<nshares>` is 0 to 100000. Note that a value of zero shares will result in no flex memory allocation, causing the virtual machine memory size to be exactly equal to its specified minimum size, even if excess memory is available.

`/proc/vmware/vm/<id>/mem/status`
Reading from this file reports current status information for the virtual machine identified by `<id>`, including the specified shares, minimum size and maximum size parameters, as well as the virtual machine name, current status (static or dynamic), whether the virtual machine is currently waiting for memory to be reserved, current memory usage, current target size, memory overhead for virtualization and percentage of allocated memory actively in use. All memory sizes are reported in kilobytes.

`/proc/vmware/sched/mem`
Reading from this file reports the memory status information for all nonsystem virtual machines in the entire system, as well as several aggregate totals.

Writing the string `realloc` to this file causes an immediate memory reallocation. Memory is normally reallocated periodically every `<MemBalancePeriod>` seconds.

`/proc/vmware/mem`
Reading from this file reports the total amount of memory that is available to be allocated, computed as the total amount of actual physical memory plus the total amount of flex memory that can be reclaimed from running virtual machines.

`/proc/vmware/config/MemMaxOvercommit`
This ESX Server option specifies the maximum level of memory overcommitment, expressed as a percentage. For example, a value of 100 allows the system to run virtual machines with an aggregate maximum size 100 percent larger than physical memory. This means that the total configured maximum sizes for all virtual machines can be as large as twice the size of physical memory. The valid range for this option is 0 (use physical memory only) to 200 (use up to three times the size of physical memory). The setting defaults to 100.

`/proc/vmware/config/MemMinFree`
This ESX Server option specifies the amount of memory, in megabytes, that the system should attempt to keep free at all times in order to handle small allocation requests immediately. The setting defaults to 2MB.

`/proc/vmware/config/MemLazyAlloc`
This ESX Server option specifies whether or not the system must eagerly reclaim all

memory reserved for a virtual machine before allowing it to start running. Lazy allocation is the default, allowing a virtual machine to start running while the system reclaims memory as needed, reducing delays. Valid values for this option are 0 (disabled) and 1 (enabled). This setting defaults to 1 (enabled).

`/proc/vmware/config/MemZeroCompress`
This ESX Server option specifies whether or not the system may reclaim from a virtual machine empty (zero-filled) pages that would otherwise block operations while waiting for sufficient memory to continue execution. This is useful for ensuring that a virtual machine will have enough memory to reboot when memory is heavily overcommitted.

When a virtual machine's guest operating system is booting, its `vmmemctl` driver has not yet been loaded, and although booting is not normally memory-intensive, some operating systems zero all available memory during the boot process.

Zero compression allows empty pages to be reclaimed automatically, even before `vmmemctl` is running. Valid values for this option are 0 (disabled) and 1 (enabled). This setting defaults to 1(enabled).

`/proc/vmware/config/MemRelaxAdmit`
This experimental ESX Server option relaxes the admission control policy for memory overcommitment. When enabled, it allows a new virtual machine to be started if sufficient memory is available to reserve its explicitly-configured minimum size; normally enough memory must be available for its maximum size (see the cautions section below).

When this option and `MemZeroCompress` are both enabled, it should be possible to boot a large virtual machine with a maximum size that exceeds available physical memory. Enabling this option without also enabling the `MemZeroCompress` option is strongly discouraged. Valid values for this option are 0 (disabled) and 1 (enabled). This setting defaults to 0 (disabled).

`/proc/vmware/config/MemDriverTimeout`
This ESX Server option specifies the time period, in seconds, after which a warning is logged for a virtual machine that has not yet started running `vmmemctl`. The valid range is 1-600 seconds, or 0 to disable. This setting defaults to 180 seconds.

`/proc/vmware/config/MemBalancePeriod`
This ESX Server option specifies the periodic time interval, in seconds, for automatic memory reallocations. The setting defaults to 15 seconds.

`/proc/vmware/config/MemSamplePeriod`
This ESX Server option specifies the periodic time interval, measured in seconds of

virtual machine time, over which memory activity is monitored in order to estimate working set sizes. The setting defaults to 30 seconds.

`/proc/vmware/config/MemIdleCost`
This ESX Server option specifies the amount charged to a virtual machine for idle pages, expressed as a ratio to the amount charged for actively used pages. The setting defaults to 4.

Examples

Suppose that we are interested in the memory allocation for the virtual machine with ID 204. To query the current memory allocation information for virtual machine 204, simply read the file:

```
% cat /proc/vmware/vm/204/mem/status
 vm status   wait  shares   min      size  %active  target     max overhd
204 dynamic  no     1280   98304  124924   58 56   124924  131072  32768
```

This indicates that virtual machine 204 has a maximum size of 131,072KB (128MB), a minimum size of 98,304KB (96MB), and a current size of approximately 124,924KB (122MB), since the overall system is slightly overcommitted. The virtual machine is also using an additional 32,768KB (32MB) because of virtualization overhead. The status reading of dynamic indicates that the virtual machine is running the `vmmemctl` driver to support dynamic memory allocation. The active percentages indicate the amount of its allocated memory that the virtual machine was actively using during recent `<MemSamplePeriod>` intervals; the short-term estimate is 58 percent, with a longer-term average of 56 percent.

To reduce the minimum amount of memory allocated to virtual machine 204 and enable a greater level of overcommitment, simply write the desired size to the `min` file, expressed in megabytes. Note that you need root privileges in order to change memory allocations:

```
    # echo 64 > /proc/vmware/vm/204/mem/min
```

The allocation change can be confirmed by reading the file again:

```
    % cat /proc/vmware/vm/204/mem/min
    64
```

Cautions

Unless the `MemRelaxAdmit` option is explicitly enabled, the current release refuses to start a virtual machine if there is insufficient memory available to initially reserve its maximum size (which may be up to three times as large as its minimum size). This effectively reduces the maximum overall level of memory overcommitment, although it is not very restrictive for systems running several virtual machines.

To avoid imposing too much load on guest operating systems, memory is reclaimed from each virtual machine at a conservative maximum rate of approximately 4MB/sec. Depending on the number of running virtual machines and the overcommitment level, it may take up to several minutes to reclaim enough memory to start a large virtual machine.

Network Bandwidth Management

VMware ESX Server supports network traffic shaping with the `nfshaper` loadable module. A loadable packet filter module defines a filter class; multiple filter instances may be active for each loaded class. The current release supports only one filter class — `nfshaper`, which is a transmit filter for outbound bandwidth management that can be attached to virtual machines using either a procfs interface or the Web-based management interface.

Using Network Filters

This section describes how to attach, detach and query filter instances from the console operating system's command line. You can also use the Web-based management interface to attach and detach `nfshaper` and obtain statistics from it.

`/proc/vmware/filters/status`
This file contains network filtering status information, including a list of all available filter classes and, for each virtual machine with attached filters, its list of attached filter instances. Read the file with `cat` to see a quick report on network filtering status.

`/proc/vmware/filters/xmitpush`
Command file used to add a new transmit filter instance to a virtual machine. Writing `<id> <class> [<args>]` to this file attaches a new instance of filter `<class>` instantiated with `<args>` to the virtual machine identified by `<id>`.

`/proc/vmware/filters/xmitpop`
Command file used to detach a transmit filter from a virtual machine. Writing `<id>` to this file detaches the last filter attached to the virtual machine identified by `<id>`.

`/proc/vmware/filters/xmit`
This directory contains a file for each active filter instance. Each file named `<class.n>` corresponds to the `<n>`th instance of filter class `<class>`.

Reading from a file reports status information for the filter instance in a class-defined format. Writing to a file issues a command to the filter instance using a class-defined syntax.

Cautions

The current release allows only a single network packet filter to be attached to each virtual machine. This restriction will be removed if VMware distributes multiple filter classes; currently the only supported filter class is the `nfshaper` traffic shaping module.

Receive filters are not yet implemented.

Traffic Shaping with nfshaper

You can manage network bandwidth allocation on a server from the Web-based management interface or from the console operating system's command line.

Using a Web browser, you may change settings from the Resource Editor page of the Web-based management interface. Be sure the virtual machine you want to change is powered on. Then, on the server's Overview page, click Manage Resources. The Resource Monitor page appears. Click the link under the name of the virtual machine for which you want to change settings. Enter the desired settings, then click Save Changes.

You must log in as root in order to change resource management settings using either the Web-based interface or the command line.

The shaper implements a two-bucket composite traffic shaping algorithm. A first token bucket controls sustained average bandwidth and burstiness. A second token bucket controls peak bandwidth during bursts. Each `nfshaper` instance is parameterized by average bps, peak bps, and burst size. The procfs interface described in Using Network Filters is used to attach `nfshaper` instances to virtual machines and detach them. A separate procfs entry is automatically created for each instance. The procfs entry can be read to query status information or written to issue dynamic commands.

The procfs interface described in Using Network Filters is used to attach an `nfshaper` instance to a virtual machine, detach an `nfshaper` instance from a virtual machine, query the status of an `nfshaper` instance or issue a dynamic command to an active `nfshaper` instance.

Commands

`config <bpsAverage> <bpsPeak> <burstSize> [<periodPeak>]`
Dynamically reconfigure the shaper to use the specified parameters: average bandwidth of `<bpsAverage>` bits per second, peak bandwidth of `<bpsPeak>` bits per second, maximum burst size of `<burstSize>` bytes and an optional peak bandwidth enforcement period `<periodPeak>` in milliseconds. Each parameter may optionally use the suffix k (1k = 1024) or m (1m = 1024k).

`maxq <nPackets>`
Dynamically set the maximum number of queued packets to `<nPackets>`.

`reset`
Dynamically reset shaper statistics.

Examples

Suppose that you want to attach a traffic shaper to limit the transmit bandwidth of the virtual machine with ID 104. To create and attach a new shaper instance, issue an `xmitpush` command as described in Using Network Filters (page 144). Note that root privileges are required to attach a filter.

```
# echo "104 nfshaper 1m 2m 160k" > \
/proc/vmware/filters/xmitpush
```

This attaches a traffic shaper with average bandwidth of 1Mbps, peak bandwidth of 2Mbps and maximum burst size of 160Kbps.

Note: This command should be entered on a single line. Do not type the backslash.

To find the number of the attached `nfshaper` instance, query the network filtering status, which contains a list of all filters attached to virtual machines:

```
% cat /proc/vmware/filters/status
```

Suppose the reported status information indicates that the filter attached to virtual machine 104 is `nfshaper.2.104`. The procfs node for this filter can be used to obtain status information:

```
% cat /proc/vmware/filters/xmit/nfshaper.2.104
```

The same procfs node can also be used to issue commands supported by the `nfshaper` class. For example, you can dynamically adjust the bandwidth limits by issuing a `config` command:

```
# echo "config 128k 256k 20k"> \
/proc/vmware/filters/xmit/nfshaper.2.104
```

Note: This command should be entered on a single line. Do not type the backslash.

When a virtual machine is terminated, all attached network filters are automatically removed and destroyed. To manually remove a shaper instance you can issue an `xmitpop` command as described in Using Network Filters (page 144). Note that root privileges are required to detach a filter.

```
# echo "104" > /proc/vmware/filters/xmitpop
```

11

Glossary

***Append disk mode* —** All writes to an append mode disk issued by software running inside the virtual machine appears to be written to the disk, but are in fact stored in a temporary file (`.REDO`). If a system administrator deletes this redo-log file, the virtual machine returns to the state it was in the last time it was used in persistent mode.

***Configuration* —** See Virtual machine configuration file.

***Console* —** See Remote console.

***Disk mode* —** A property of a virtual disk that defines its external behavior but is completely invisible to the guest operating system. There are four modes: persistent (changes to the disk are always preserved across sessions), nonpersistent (changes are never preserved), undoable (changes are preserved at the user's discretion) and append (similar to undoable, but the changes are preserved until a system administrator deletes the redo-log file). Disk modes may be changed from the Web-based management interface.

***Event log* —** A page in the Web-based management interface that displays the most recent actions or events recorded in a virtual machine.

***Guest operating system* —** An operating system that runs inside a virtual machine.

***Headless* —** A virtual machine that runs in the background without any interface connected to it runs headless.

***Host machine* —** A physical computer (as opposed to a virtual machine).

***Nonpersistent disk mode* —** All disk writes issued by software running inside a virtual machine with a nonpersistent disk appear to be written to disk, but are in fact discarded after the session is powered down. As a result, a disk in nonpersistent mode is not modified by ESX Server.

***Persistent disk mode* —** All disk writes issued by software running inside a virtual machine are immediately and permanently written to a persistent virtual disk. As a result, a disk in persistent mode behaves like a conventional disk drive on a physical computer.

***Remote console* —** An interface to a virtual machine that provides non-exclusive access to the virtual machine from workstations with a network connection to that host machine.

***Resume* —** The way to return a virtual machine to operation after it has been suspended.

Suspend — The way to save the current state of a virtual machine. Use the resume feature to restart a suspended virtual machine, with all running applications in the same state they were at the time the virtual machine was suspended.

Undoable disk mode — All writes to an undoable disk issued by software running inside the virtual machines appear to be written to the disk, but are in fact stored in a temporary file (`.REDO`) for the duration of the session. When the virtual machine is powered down, the user is given three choices: (1) permanently apply all changes to the disk; (2) discard the changes, thus restoring the disk to its previous state; or (3) keep the changes, so that further changes from future sessions can be added to the log.

Virtual disk — A virtual disk is a file on a file system accessible from the server. To a guest operating system, it appears to bea physical disk drive. This file can be on the server where the virtual machine is running or on a remote file system.

Virtual machine — A virtualized x86 PC environment on which a guest operating system and associated application software can run. Multiple virtual machines can operate on the same server machine concurrently.

Virtual machine configuration — The specification of what virtual devices (disks, memory size, etc.) are present in a virtual machine and how they are mapped to host files and devices.

Virtual machine configuration file — A file containing a virtual machine configuration. It is created when you set up a virtual machine. It may be modified from the Web-based management interface or by editing the file in a text editor.

VMware authentication daemon — The process, named `vmware-authd`, that ESX Server employs to authenticate users.

VMware Tools — A suite of utilities that enhances the performance of your guest operating system; VMware Tools includes the SVGA driver, VMware guest operating system service, the `vmmemctl` module, a network driver and the VMware Tools control panel.

Web-based management interface — A browser-based tool that allows you to control (start, suspend, resume, reset, and stop) and monitor virtual machines and the server they run on, modify a virtual machine’s configuration, and set up a new virtual machine.

Index