

VMware® Infrastructure 3

Advanced Technical Design Guide

~and~

Advanced Operations Guide

Two books in one!



Ron Oglesby
Scott Herold
Mike Laverick

This book is the domain of Mike Laverick of RTFM Education. Mike's unmatched documentation on the internet led us (Ron and Scott) to seek him out and ask him to participate in this book. While we felt that our "Design Guide" hit a great target audience, we knew that the addition of a companion "Operations Guide" would make it a much more valuable resource. We, as a group, thought that the two books in one concept was a winner, and thus this second book –inside- a- book was born.

The Operations Guide

The purpose of this book is to guide you through the most popular and recommended configurations of VMware's ESX and VirtualCenter. It is not intended as complete reference. However, we do intend to add additional updates during the lifetime of this book. These will be released for free on the vi3book.com website. After all, it's not uncommon for procedures to change and for new features to be added to ESX or VirtualCenter within a release.

I used a mix of hardware when writing this guide: some old and some new. My equipment has two CPUs, four network cards, and basic connectivity to a Storage Area Network (SAN). In some cases, I had a fully redundant connection to the SAN for the purposes of capturing a real-world configuration in dialog boxes and command-line tools.

The structure of this guide was taken from the VMware authorized course "Virtual Infrastructure 3: Install and Configure" which I teach on a regular basis; the idea being that this guide could act as a companion to the course manual you might have if you attended the authorized course. It adds additional information and tips and tricks along the way. It also tries to explain various features and options in a new light which might assist you. I thought by mimicking the structure of the official VMware course this might help people whose aim is to use this guide as part of their preparation for the VMware VCP Test.

Additionally, I thought that this structure might be of benefit to those studying the product on their own.

Chapter 1: Installing ESX 3.x

You will be pleased to know that ESX is one of the easiest operating systems you will ever install. You are asked even fewer questions than the ESX 2.x installation, and the biggest part of the installation is the file copy event.

Confirm your source code ISO is good

You can download ESX 3.x from VMware's website once they have provided you with the correct serial numbers and logins.

After downloading the source ISO file, you should really check the ISO before you burn the ISO to a CD. You can do this using the Md5sum value. Md5sums allow you to confirm your ISO file has not been corrupted by the download process. The source Md5sum value is normally put under the link to the ISO or file you are downloading.

If you are running Linux on your desktop PC you will already have access to md5sum on the command-line. If, on the other hand, you are running Microsoft Windows, there are some free tools to check that your file is not corrupted. Nullriver Software has a very neat and free Windows Md5sum checker.

<http://www.nullriver.com/>

Additionally, rather than burning a CD, you could always use your ISO with the "virtual media" features of your server's ILO, if it has one. In the past, users have had problems with the installer engine and ILOs; however, these issues appear to have been resolved in this release. If you are using an ILO to do your installation, I would personally recommend using the "text" mode for the ESX install as I frequently find that the mouse is simply not there or too slow or disorientated. This is due to the limitations of some ILOs as opposed to the ESX installer itself. If you do use the graphical installer and your mouse does not work you can still use the keyboard:

- Tab – to move forwards

-
- Shift-Tab – to move backwards
 - Space-Bar – open pull-down lists
 - Cursor Keys – to navigate lists
 - Enter – to select

Anaconda the Installer

The installation engine behind ESX is “Anaconda,” which the install engine used for Redhat Linux. In fact, VMware’s installer is just a lightly modified version of the same installer. Anaconda also supports scripted installations from a cd-rom, boot floppy disk or using the Pre-eXecution Environment (PXE) available on most modern network cards. We will cover scripted installation from a PXE boot server appliance in the chapter *ESX on the command-line*.

There are only two areas during installation which could cause you problems: selecting the correct network card for use with the Service Console and building a robust and reliable partition table.

The Partition Scheme

Like most operating systems, the reliability of the build will come from the installer’s decisions. Critically, we want a partition scheme which will protect itself from rapidly filling event log files and “users” copying large files to the wrong location. Almost no one in the VMware Community uses the “automatic” partition scheme called “Recommended” in the installer. Nearly everyone in the VMware Community uses their own manual partition scheme based on user’s personal experience.

The VMware Community has debated the advantages and disadvantages of various approaches in a very long and interesting forum thread. It was started by Steve Beaver, a very good friend of ours. If this interests you, then pop along and have a read. There are as many partition schemes as there are people on the Forum and ours is just one example. The thread is called “Taking a poll of the manual partitions people are using for ESX 3.0” and the thread ID is 425022.

<http://www.vmware.com/community/thread.jspa?messageID=425022>

Rather than using drive letters (C: D: E) to address these partitions, folders are used because ESX is based around the Linux/UNIX world. You have been able to do something similar to this with Microsoft operating systems since Windows 2000 was released. As for the physical disks themselves, I usually recommend two 36GB or 72GB disks in a mirror.

In this example I am going to guide you through an installation to local LUNs. Later I will show you how to set-up a boot from SAN configuration. If you are installing ESX to the local storage, I recommend disconnecting the SAN cables if you can. This prevents any chance of installing ESX to SAN unintentionally – it also reduces your chances of accidentally destroying terabytes of data on the SAN while installing ESX!

Installing the ESX Operating System Locally

1. Boot to the ESX 3.x CD

Note: Text Mode or Graphical?

2. At the welcome screen you will be given two choices. Press the [Enter] key to enter the graphical installation. Type the word "text" and press [Enter] to enter the text mode installation. If you simply wait 60 seconds, the installer will enter the graphical mode by default.
3. Choose to Skip the CD Media Test

Note: Media Tests

The media test checks the integrity of your CD – if you wish you may test the physical CD – but if you have already done a md5sum you should be good to go.

4. Click Next.
5. Select your Keyboard Type.

-
6. Select your Mouse Type.

Note: Mouse Type

Selecting the mouse type isn't dreadfully important, as ESX does not run with a graphic interface such as X-Windows or KDE. The question about mouse type is largely due to the installer's basis in Anaconda.

Note: Install or Upgrade?

The system will check for previous ESX installations. If you are upgrading or re-installing you will be given the options for an "upgrade" or installation. If you have completely blank hard-drives then you should receive no message at all.

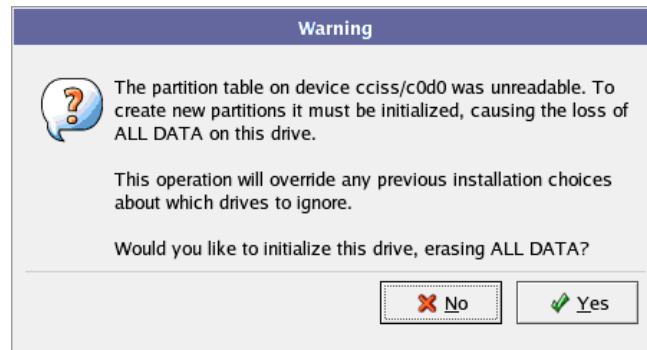
7. Agree the EULA.

Note: Initializing LUNs

After agreeing the EULA you may receive a warning that the `/dev/sdaN` is unreadable. This reference to `/dev/sdN` indicates the SCSI Disk (SD), which is found first (A, B, C). Also, the dialog box should indicate on which controller this disk was found. In Figure 1.1 we can see that the controller is actually a cciss. This is a Compaq Computer Smart Raid Array 6i on my HP Proliant DL385. `/c0d0` is the first controller (c0) and the first lun (d0).

As the partition table doesn't exist, the Anaconda installer will wish to initialize the disk. This operation is potentially fatal if there is data on the drive. In this case, it is fine to accept "yes" because I know it is blank. If your ESX host is connected to SAN, be very careful in accepting yes, as this could be very dangerous.

Figure 1.1



8. Select Advanced as your partition scheme.

Below is a summary of a recommended partition scheme, followed by notes which explain their size and purpose. This partition scheme only uses about 15GB of space. You could easily increase these values and add more partitions. Generally, I create three primary partitions, and the rest are treated as logical drives in an extended partition. The disk management utility (called Disk Druid) automatically creates an extended partition for you using the remainder of the disk once you have created the primaries.

Mount Point	File System	Fixed Size	Size in MB	Force to Primary	Purpose
/boot	ext3	X	250	X	Core Boot (img) files
n/a	swap	X	1600	X	Swap for Service Console
/	ext3	X	5120	X	Main OS location
/var	ext3	X	2048		Log files
/tmp	ext3	X	2048		Temporary Files
/opt	ext3	X	2048		VMware HA Logging
/home	ext3	X	2048		Location of users storage
NA	vmkcore		100		VMkernel Panic Location
	vmfs	Fill to remaining disk			Local storage is only required for VM Clustering

ext3

EXT3 is the primary file system of Linux. We will use this one since the Service Console is based on Redhat Linux. There are two other propriety file systems which are only accessible by the VMkernel. These are vmkcore and VMFS version 3. After the install has completed we will cover VMFS in more detail in the storage chapter.

/boot

This is where core boot files with an img extension are stored. After powering on ESX, the master boot record is located and boot loader is run. In the case of ESX 3.x, this is now the Grand Unified Bootloader (GRUB). GRUB then displays a menu which allows you to select what .img to execute. Img files are image files and are bootable. They are analogous to ISO files which can also be bootable. I've doubled the VMware recommendation to err on the side of caution. Previous VMware recommendations have made this partition too small. This gave people problems when upgrading from ESX 2.x to ESX 3.x through lack of disk space.

/swap

This is where the Service Console swaps files if memory is low. I've chosen to over-allocate this partition. The default amount of memory for the Service Console is 272. VMware usually takes this number and doubles it to calculate the swap partition size (544MB). The maximum amount of memory you can assign to the Service Console is 800MB. This is how I derived the 1600MB value. This means if we ever choose to change the default amount memory assigned to the Service Console we do not have to worry about resizing the swap partition. It doesn't have a mounting point as no files from the administrator are copied there.

/ (referred to as the “root” partition)

This is the main location where the ESX operating system and configuration files are copied. If you have a Windows background, you can see it a bit like the C: partition and folders coming off that drive like C:\Windows or C:\Program directory. If this partition fills, you may experience performance and reliability issues with the Service Console, just like you would with Windows or any other operating system.

/var

This is where log files are held. I generally give this a separate partition just to make sure that excessive logging does not fill the / file system. Log files are normally held in /var/log, but occasionally hardware vendors place their hardware management agent log files in /var.

/tmp

In the past, VMware has recommend using a separate partition for /tmp which I have always done in ESX 2.x, as well. As I have plenty of disk space I have made this larger than it really needs to be.

/opt

Several Forum members have seen the /opt directory fill up very rapidly and then fill the / partition. This location is also *sometimes* used as a logging location for hardware agents. In VMware, HA has been seen to generate logging data here as well, so I create a separate partition for it to make sure it does not fill the / partition.

/home (Optional)

Technically, you don't need a separate partition. In the past VMware recommended one for its ESX 2.x in production. This was due to the fact that VM's configuration files-the vmx, nvram and log- were stored in /home. In ESX 3.x, all the files that make up a VM are more likely to be located on external storage. I still create it for consistency purposes and if I have users on the local ESX server they are more likely to create files here than in a directory coming off the / partition.

Vmkcore

This is a special partition used only if the ESX VMkernel crashes, commonly referred to as the "Purple Screen of Death." If that happens then ESX writes debugging information into this partition. After a successful boot the system will automatically run a script to extract and compress the data to a "zip" file in /root. This file with tar.gz extension can be sent to VMware Support who will work to identify the source of the problem. These PSODs are normally caused by failures of RAM or CPU. You can see a rogue's gallery of PSODs at:

http://www.rtfm-ed.co.uk/?page_id=246

vmfs

VMFS is VMware's ESX native file system which is used for storing all the files of the VM, ISOs, and templates. Generally, we use external storage for this. The only case for using local storage for your VMs is when you do not have access to external storage. Here I am assuming you have access to external storage. Therefore, you have no need for a local VMFS partition.

GOTCHA:

There is one major qualification to this statement: if you want to run clustering software, such as Microsoft's Clustering Software, inside a VM you will need local storage. VMware does not support storing virtual disks or Raw Device Mappings (RDMs) used in VM clustering scenarios on SAN or iSCSI based storage.

/vmimages

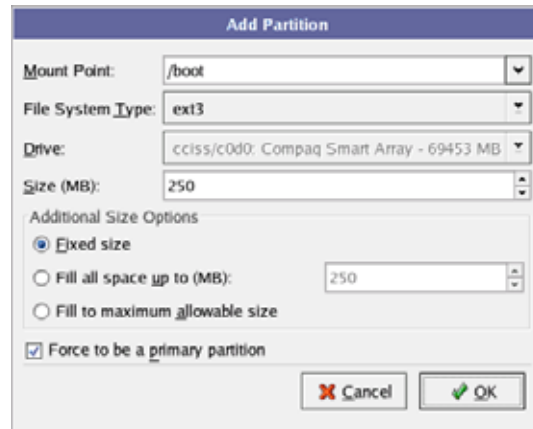
In ESX 2.x, we used to create a /vmimages partition or mounting point to a network location. This storage was used primarily for templates and ISOs. This partition is no longer required, as we now have more effective and easier ways of storing this data. Of course, if you are an ESX 2.x veteran who wishes to keep using /vmimages for consistency purposes then that is fine. It's just no longer required or recommended. Personally, I still like to have a portion of disk space given over to this partition location as a "working area" when I am dealing with large files. I've found my recent purchase of a SAN has made this something I use much less.

If you have done a clean install you will have a directory called /vmimages, even if you haven't created a /vmimages, which contains files used internally by VMware ESX. I will discuss this more in chapter 5 when we create VMs.

1. Click New and this will begin the process of creating partitions.

Figure 1.2 shows the new dialog box. In this case, I select /boot from the pull down list (you can type in this area). I chose ext3 as the partition type and entered 250MB as the size. I left the option as “Fixed” size and indicated with the tick that I wished this partition to be a primary partition. If I left this unchecked then the system would begin creating logical drives in an extended partition.

Figure 1.2



GOTCHA:

After you have built the partition table, take a few moments to confirm you created the right number of partitions of the right size. The partition table is not committed to the disk until you click “Next.” This would be an ideal opportunity to spot an error and correct it. Partition errors can be corrected afterwards using fdisk, but it’s much easier to get it right the first time. In my experience, a bad partition scheme usually results in wasted time spent re-installing ESX.

2. In The Advanced Options Page

Generally, you can click Next here. But before you do make sure that the option “Install MBR to a drive” is the *same LUN* you selected earlier when you partitioned the disk. Occasionally, I have seen people select a SAN LUNs presented as the location for the MBR, and this has caused problems when their intention is to complete a local installation.

The other options aren't especially relevant to us. Force LBA32 (Large Block Addressing) would allow /boot to operate above 1024 cylinder limit (around 8GB of disk space) and is normally enabled on legacy hardware if the BIOS supports it. Only a very old server needs the LBA32 code.

The "From a Partition" is also for legacy hardware that stores BIOS information in a partition. This used to be seen on some very old Compaqs that had an Extended Industry Standard Architecture (EISA) partition. This is not the generation of equipment to which we normally install ESX server.

These options are a throwback to the fact that this is a re-engineered Linux installer. They are unlikely to be options you would use in a production environment.

3. Network Configuration

Select your NIC which will be used by the Service Console

Set your IP Address

Subnet Mask

Default Gateway

Primary & and Secondary DNS and FQDN

Set the VLAN ID, if you're using VLAN

Disable option Create a default network for virtual machines

Note: Use a FQDN

A Fully-Qualified Domain Name and DNS name resolution are required for two core features – the VMware License Server and VMware HA. I recommend you have your DNS infrastructure set-up and in place before beginning the ESX installation so the Licensing and HA are easily configured.

Note: Default Networking Options

I wouldn't recommend the option "Create a default network for Virtual Machines." This would allow a VM's network traffic and Service Console management traffic to co-exist on the *same* net-

work. It is recommended you separate this traffic for security and performance reasons.

GOTCHA:

How do you know you have selected the right network card during the install? If a server has mix of networking speeds/duplex, then ideally I would want to use a 100mps/Full-Duplex card with the Service Console, dedicating my 1000mps interfaces to the systems that really need the bandwidth, namely my VMs.

Aside from that you would have to consult documentation of your system to know which PCI Bus:Slot:Function number was related to each card. By default the install merely scans the PCI Bus and locates the card in its order of the Bus. If after installing the ESX product you find you cannot even ping the Service Console interface it could be you selected the wrong NIC during the install. If this happens to you, you may wish to consult the end of Chapter 2 entitled “Managing Service Console Networking.”

4. Set your Time Zone and, if appropriate, UTC Settings.

Note: Time Zones and UTC

UTC stands for Universal Time Co-ordinate. Despite its name it is the same as Greenwich Mean Time (GMT) where zero hour begins at the Greenwich Meridian in London, United Kingdom. If you're in the UK like me you might like to know that even we don't strictly obey UTC. We partially obey GMT and BST (British Summer Time) depending on the time of the year. Sometimes we are in synch with UTC and other times we are not. For this reason, if you enable UTC and select the Europe/London your system clock will be 1hr adrift, depending on the time of the year. Also, you may wish to check the UTC tab to make sure you are correctly setting where you are, relative to GMT.

Getting time right on ESX is pretty critical. Firstly, because in most configurations this is where VMs get their time. Secondly, VMs can suffer from “clock drifting” meaning the virtual machines “clock” is out of synch with the ESX system clock. Resolving this clock drift begins with getting your time settings correct in the first place. We can synchronize the time of ESX with Network Time Protocol

(NTP) server on the internet for greater accuracy. I will cover this later in Chapter 12, *ESX on the Command-Line*.

5. Set the root password.

Note: Creating other users on the ESX Host

The root account is the user with highest privileges. Currently, there is no method to create other ESX users within the installer – this has to be done with the “VI client.” The main reason for doing this is so you can remotely connect to the ESX host with a Secure Shell (SSH) client to get a command-line view of your ESX hosts without resorting to the ILO or RAC card on your server.

6. Watch the Status Bars.
7. You will then be left with a summary of your choices. At this stage you can still go back and change some settings, such as your IP address, if you feel you have set them incorrectly. Clicking “Next” initiates the copy process. At the end of the install clicking Finish will normally cause the CD-ROM to eject and when the system reboots you get your first boot of ESX server.
8. If you are using virtual media to access the ESX 3 ISO, remember to disconnect it before the reboot. You might find that BIOS settings cause the server to boot to the CD again.

Creating a Local ESX User Account for SSH/PuTTY Access

There is a protocol called SSH (Secure Shell) which allows us to gain access to ESX’s command-line interface without the use of an ILO. A very popular tool used to create a remote SSH session is PuTTY, created by Simon Tatham. This can be found at:

<http://www.chiark.greenend.org.uk/~sgtatham/putty/>

Why would you want this command-line style access to ESX? Firstly, there are some tasks that *only* can be done via the command-line. In other situations, because you have a mix of command-line and GUI tasks, it is easier to choose one interface rather than work with two.

However, in ESX 3.x there are new security settings which disable SSH access to the root account. By default, in a clean install there is no access to the ESX Service Console for the root account except via the ILO. So I create a local account on ESX, use PuTTY to gain access, and then finally switch to the root account. This security change was introduced to enforce more user account traceability and an audit trail in ESX 3.

Note: User activity such as logons and changes in privilege are logged in /var/log/messages.

In the installation there is no way to create this user account, so we must use VMware's Virtual Infrastructure client to gain access.

Download and Install the Virtual Infrastructure Client

The main graphical tool used to administer either ESX server directly or via VirtualCenter is called the Virtual Infrastructure Client (VI client). We can download and install this tool from the ESX host. We can then use this tool to create a local ESX user suitable for login in remotely with SSH.

1. From your Management PC open a web-browser.
2. Type in the FQDN of your ESX host such as <https://esx1.vi3book.com>

Note:

Choose Yes to continue to access the ESX host despite the fact its certificate is not trusted. ESX creates a server certificate for each installation you do. These are auto-generated using OpenSSL and are not signed from recognized root certificate authority.

3. Select the link Download the Virtual Infrastructure Client.
4. Save or Open the VMware-viclient.exe file and install it to your management PC.

Note: After installing the VI client you should be able open it from the shortcut on your desktop logging onto your ESX host by its FQDN, root and root's password

Create a Local User on an ESX Host

1. Click the User & Groups tab.
2. Right-Click the Window and Choose Add.
3. Fill in the dialog box as outlined in Figure 1.3.

Figure 1.3

The screenshot shows a 'User Information' dialog box with the following fields and values:

- User Information:**
 - Login: lavericm
 - UID: (empty)
 - User Name: Mike Laverick
 - Below User Name: User Name name and UID are optional
- Enter password:**
 - Password: (masked with asterisks)
 - Confirm: (masked with asterisks)
- Shell Access:**
 - ☒ Grant shell access to this user
- Group membership:** (empty)

Note: Grant Shell Access

You must enable "Grant shell access" to this user for the account to have access to the Service Console via SSH. Otherwise they will only access to the ESX host with VI client.

4. Click **OK**.

Gaining Command-Line Access as ROOT with SSH

1. **Open a SSH/PuTTY Session to the ESX Host.**
2. **Login** with your recently created account.
3. To levitate to a higher plane of privileges type:

su -

4. **Enter the root's password assigned during installation.**

Note: Using the SU Command

The SU command allows you to “switch user” and assumes, unless otherwise specified, that you would like to change to “root.” The minus sign indicates you would like to use the root account's environmental settings, such as path variables. This is very important if you want to run any commands properly and not receive “command not found” error messages.

There is a more sophisticated way to allow elevated access without even using the su command or having knowledge of the root account's password. There is a utility called “sudo” which allows you say which commands an ordinary user can run. This stops you from having to disclose even the root password; to run commands normally used by root. However, for now the method above does work. It is easy to use and still provides for traceability. I will cover a simple sudo configuration in the chapter *ESX on the Command-Line*.

Enabling SSH from your ESX host to other hosts

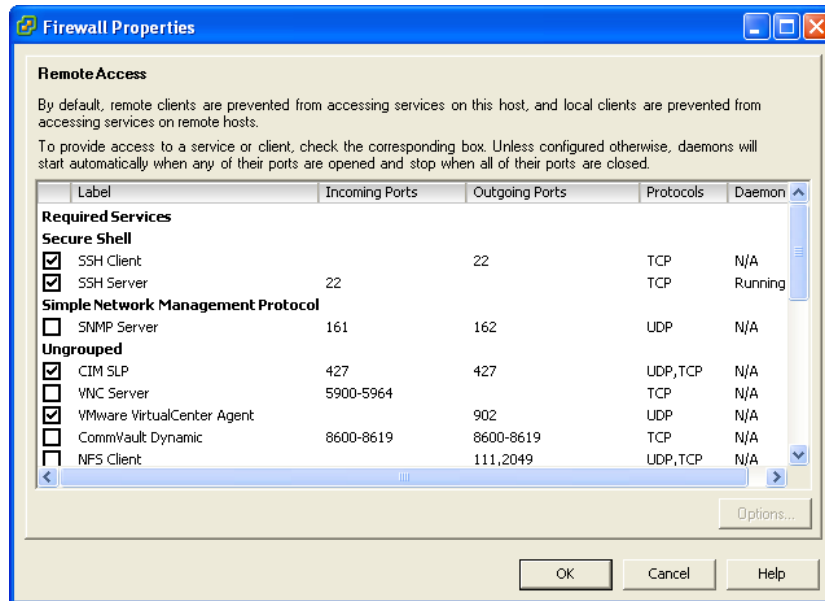
One of the things I like to do is connect to one ESX host and then use the SSH at the Service Console to get to my other servers – this saves me having to open repeated PuTTY sessions. However, you will be unable to use the SCP (Secure Copy) command to copy files to ESX servers from an ESX server. Under the default settings this is not allowed in ESX 3.x as the firewall denies the client (although every ESX host is an SSH server), with an error like this:

```
ssh: connect to host esx2.vi3book.com port 22: Connection refused
```

To enable this kind of access we need to adjust the firewall settings:

1. In the VI client click the Configuration tab.
2. Under the Software Panel, select Security Profile.
3. Click the Properties link in the far right-hand corner.
4. Enable the SSH Client option and click OK.

Figure 1.4



Note: Using SSH

This is all that's required for SSH. So now you could type `ssh -l lav-ericm esx2.vi3book.com` and then use `su -` to elevate your privileges to root rights. If you ever get confused about who you are, try the command `whoami`, and if you are unsure which host you're connected to, try the command `hostname`. You can use the command `exit` to leave ssh sessions.

Note: If you were using the command-line you could have enabled the SSH client using the command-line version of the ESX firewall. Please note this command, as with *all* commands, *is* case-sensitive.

```
esxcfg-firewall -e sshClient
```

GOTCHA:

Many `esxcfg` commands require restarting the `hostd` daemon if you want the VI client to refresh and show your command-line changes in the GUI. You can do this with `service mgmt-vmware restart`.

Configuring ESX 3.x for SAN Booting

Firstly, if you are unfamiliar with SAN technology you might wish to skip this section and proceed to Chapter 3, *Storage*. There you can learn more about conventional ESX deployments, leveraged SAN based storage. You could then return to this chapter fully armed with the background knowledge to safely continue.

Since ESX 2.x VMware has supported booting from both local storage and from SAN based storage, SAN based booting is a tempting option, especially if you run ESX servers from blades. It allows you to quickly remove a failed blade with a working one, and bring that ESX host back online quickly. Another advantage is that you can leverage your SAN snapshot features as a way of backing up your ESX server build. Before I begin an overview of preparing ESX for SAN-based booting, it's important to know some key restrictions and what is actually supported by VMware:

- Booting from SAN and using VM clustering (clustering software from Microsoft or VERITAS for example) is not supported.
- With ESX 2.x you could not use a feature called "Raw Device Mappings" (RDMs). RDMs allow a VM access to native LUNs on the SAN for existing data or storing your data using the guest operating systems native files system on a SAN LUN rather than creating a virtual disk on VMFS partition. RDM files and SAN booting are now supported together.
- ESX server can see up to a maximum of 256 LUNs (0-255). However, the Anaconda installer only displays the first 128 LUNs (0-127).
- In the past, VMware only supported booting for the "lowest order" LUN's, so if an ESX host had access to LUN's 10, 12, and 15. The LUN selected as the boot LUN would be 10. This restriction no longer applies to ESX 3 installations.

Along with these soft issues there are a number of physical hardware requirements that must also be met:

-
- The Fibre Channel card used should sit highest in the PCI bus for performance reasons and should be plugged into an “active” storage processor on the SAN.
 - Correct LUN masking and zoning should be implemented so only that ESX host can see the correct LUN. This has some implications when you replace a blade in the enclosure. The new blade will present a different World Wide Name (WWN) value. The SAN will need reconfiguring to present this LUN to the new ESX host.
 - Boot from SAN is only supported in conjunction with fibre channel switch – direct connections without a switch and Fibre Channel Arbitrated Loop are not supported.
 - IBM eServer Bladecenters that ship with IDE drives onboard need these drives disabling on each blade.

Generally, the install procedure for boot-from-SAN differs little from an install to local storage. The only real difference is, rather than selecting local storage when you partition the disk or set the location of the Master Boot Record, you select a LUN on the SAN.

Firstly, before the installation I would recommend that you set the main system BIOS to boot from the CD-ROM. In the main BIOS we set the fibre channel storage controller to be above the internal RAID controller card. This stops the server booting to local internal storage.

Secondly, once the main BIOS has been correctly configured you will have to enter the configuration tool for your fibre-channel card. The fibre-channel device will need to be enabled as a boot device – and the LUN where installation files were copied selected as the boot LUN. Clearly, these settings are very hardware specific. The following instructions were developed using HP Proliant DL 385 using a Qlogic 2200 fibre channel card. If this is not your hardware I would recommend consulting your vendor’s procedures for enabling SAN based booting.

GOTCHA:

One of the appeals of boot from SAN is that nothing about the ESX host itself is physically stored locally. This makes the ESX product more like an appliance, merely an engine for running VMs. There is, however, one drawback to this boot from SAN approach which is especially appealing with blade technology. If you want to setup clustering software such as Microsoft or VERITAS Clustering Services, you would need local storage on blades. VMware does not support virtual machine clustering with the boot disks of the VMs stored on the SAN. If you are configuring VM clustering and wish to be supported by VMware, the virtual disks of the VM must be on *local storage*.

Configuring the Fibre Channel BIOS Options

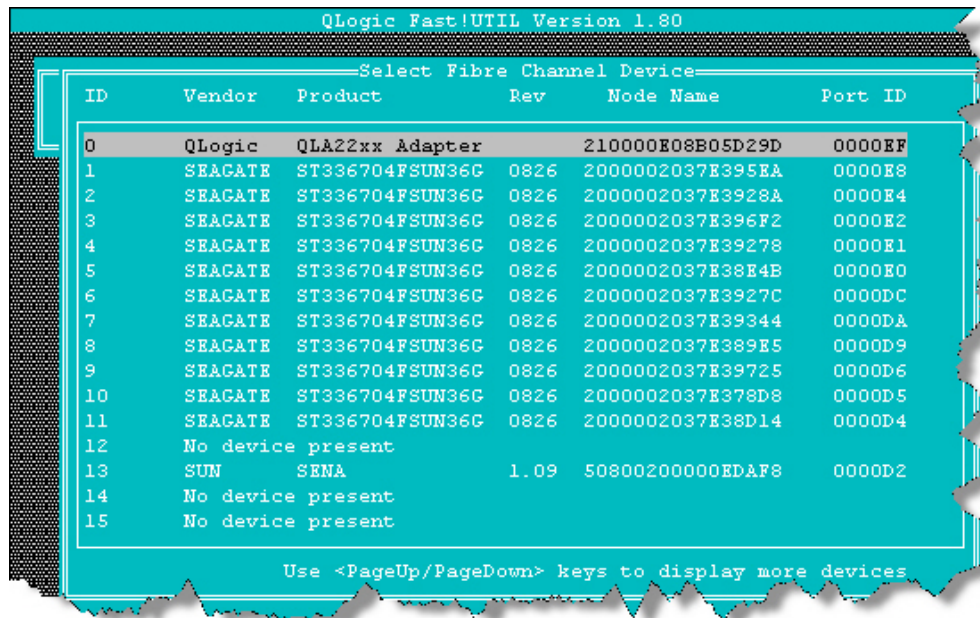
TIP:

Before you begin I recommend either removing physical disks or using your RAID controller card to remove any LUNs. This will ensure the only storage you see during the install is SAN based.

1. Power on your server.
2. Press **[Alt+Q]** to enter **Qlogic Cards BIOS settings**.
3. Choose **Configuration Settings** [Enter].
4. Choose **Host Adapter Settings** [Enter].
5. Set the **Host Adapter BIOS:** option to be **Enabled**.
6. Press **[ESC]** to exit the **Host Adapter Settings** menu.
7. In **Configuration Settings** choose **Selectable Boot Settings**.
8. Set **Selectable Boot Device** option to be **Enabled**.
9. Cursor down to select **Current Boot Node Name:** and press **[Enter]**.

Figure 1.5 shows the disks displayed to my server. I'm using quite an old Sun Microsystems SAN with Qlogic 2200 fibre channel card.

Figure 1.5



QLogic Fast!UTIL Version 1.80

Select Fibre Channel Device

ID	Vendor	Product	Rev	Node Name	Port ID
0	QLogic	QLA22xx Adapter		210000E08B05D29D	0000EF
1	SEACATE	ST336704FSUN36C	0826	2000002037E395EA	0000E8
2	SEACATE	ST336704FSUN36C	0826	2000002037E3928A	0000E4
3	SEACATE	ST336704FSUN36C	0826	2000002037E396F2	0000E2
4	SEACATE	ST336704FSUN36C	0826	2000002037E39278	0000E1
5	SEACATE	ST336704FSUN36C	0826	2000002037E38E4B	0000E0
6	SEACATE	ST336704FSUN36C	0826	2000002037E3927C	0000DC
7	SEACATE	ST336704FSUN36C	0826	2000002037E39344	0000DA
8	SEACATE	ST336704FSUN36C	0826	2000002037E389E5	0000D9
9	SEACATE	ST336704FSUN36C	0826	2000002037E39725	0000D6
10	SEACATE	ST336704FSUN36C	0826	2000002037E378D8	0000D5
11	SEACATE	ST336704FSUN36C	0826	2000002037E38D14	0000D4
12	No device present				
13	SUN	SENA	1.09	50800200000EDAFA8	0000D2
14	No device present				
15	No device present				

Use <PageUp/PageDown> keys to display more devices

Note: Use LUN Masking

Here my ESX host can see many LUNs/Disks. You might find it easier to mask all LUNs away from the server except the SAN boot LUN. This will make selecting the right LUN much easier and clearer.

10. **Select the appropriate LUN or disk** from the list. I will use ID9 which has the Node name that ends with 20nnnnnnnnnnr9725.
11. Press **[ESC]** and **Save your changes**.
12. Finally, **Exit Fast!UTIL** and **Choose to Reboot** the system.

Configure the Main System BIOS

Different systems use different BIOS providers and different keyboard strokes to gain access to the main keyboard settings. For example, on most HP systems it is F10 to enter the BIOS, but on Dell systems it is normally F2.

Below is the procedure for HP Proliant:

1. Press [**F10**] at the prompt.
2. Choose the option **Setup Utility** and press [**Enter**].
3. Choose **Boot Controller Order (IPL)** and press [**Enter**].
4. Select the **Qlogic Card** and Press [**Enter**].

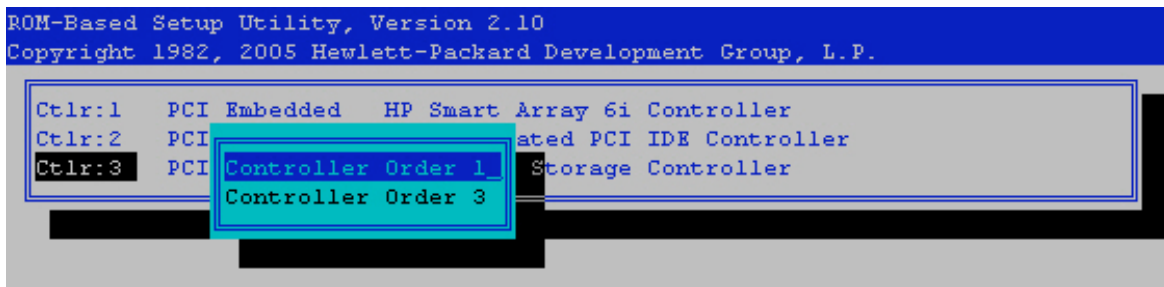
Note:

On my system this appears as Ctlr:3 PCI Slot 3 SCSI Mass Storage Controller.

5. Choose **Control Order 3**.

Figure 1.6 shows configuring the Qlogic Card to be the primary device for boot purposes.

Figure 1.6



Note: In a DELL BIOS

If you were doing this in a Dell BIOS you would choose the option called "Hard Drive Sequence" and use the +/- keys to move the Qlogic card to the top of the list.

6. Press [**ESC**] and [**F10**] to save your changes.

TIP:

Normally, I have a personal preference for the boot order on my ESX hosts which is:

- Hard Drive

-
- CD-ROM
 - Floppy
 - USB Keydrive
 - PXE

This is to prevent me accidentally leaving a CD, Floppy or USB key in drive and finding a reboot boots to removable media. You might notice that the ESX CD does not ask “Press any key to boot from this CD” as some operating systems do. When it comes to boot from SAN setups I change this order to:

- CD-ROM
- Hard Drive
- Floppy
- USB Key
- PXE

This is to ensure that when I reboot the server I get to the CD the first time, and the system doesn't try to boot from a blank LUN before I have had the opportunity to install ESX there.

Installing ESX to a SAN LUN

1. Insert the ESX CD.
2. Choose **Graphical** or **Text**.

Note: Choosing the Correct LUN

During the installation choose the correct LUN from the SAN when you come to partition the LUN, and remember to put the MBR on the *same* LUN. Figures 1.7 and 1.8 shows the significant dialog boxes in the boot from SAN installation.

Figure 1.7

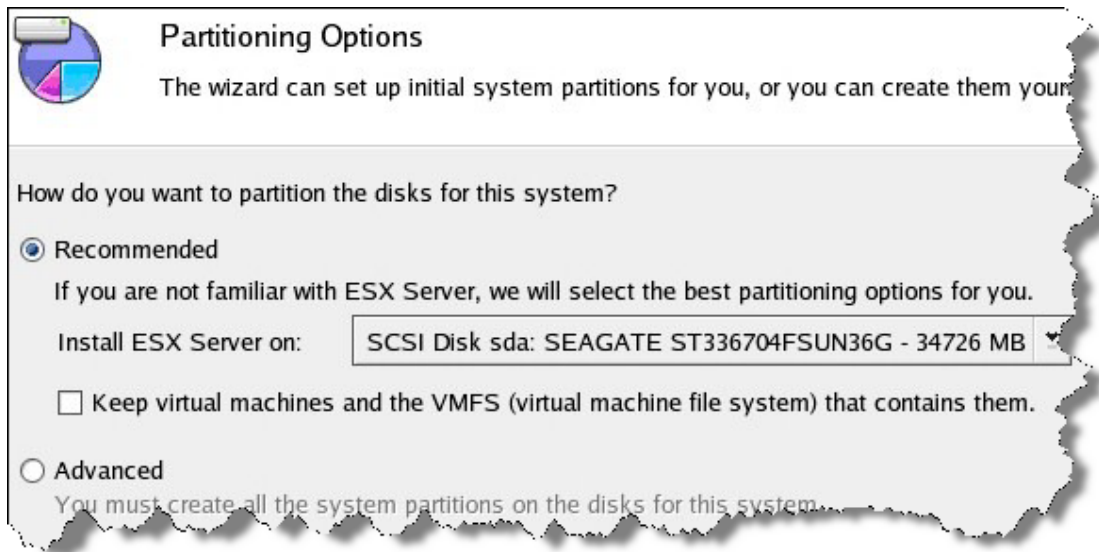
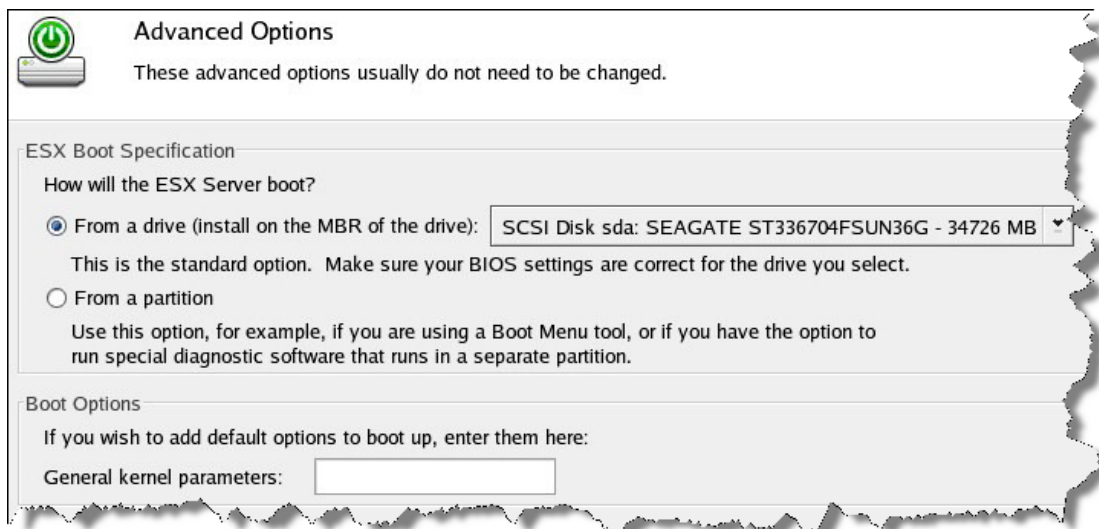


Figure 1.8



Summary

By now, I hope you are fully au fait with the ESX install procedure. In time you will probably do a few ESX installations, and all of this will become second nature to you. Eventually you will get truly bored with the idea of manual installations from a CD-ROM. Manual installations take time and only address a small

part of the overall configuration of the ESX host for use in a live environment. That's where the next two chapters on Networking and Storage are taking us.

The post-configuration stages include:

- Network Configuration
- Storage Configuration such as NAS and iSCSI
- Creating ESX Users and Controlling Their Rights
- Installing 3rd Party Hardware Agents such as HP Insight Manager, Dell OpenManage and IBM Director
- Patching ESX
- Adding into VirtualCenter and Licensing

You'll be pleased to hear that much of this configuration can be automated with deployment tools, some of which are free. If you're interested in this subject then you may wish to proceed to the chapter titled *ESX on the Command-Line*. Here I cover scripted installations with "Kickstart" scripts with the source code delivered across the network with a free PXE "virtual appliance." Personally, I would advise reading the Networking and Storage chapters if you are new to the product so you are completely familiar with the post-configuration of an ESX host before you consider automating the process.