

Professional Expertise Distilled

Implementing Citrix XenServer Quickstarter

A practical guide to getting started with the Citrix XenServer Virtualization technology with easy-to-follow instructions

Gohar Ahmed

www.hellodigi.ir

[PACKT] enterprise 
PUBLISHING professional expertise distilled

Implementing Citrix XenServer Quickstarter

A practical guide to getting started with the
Citrix XenServer Virtualization technology
with easy-to-follow instructions

Gohar Ahmed



BIRMINGHAM - MUMBAI

Implementing Citrix XenServer Quickstarter

Copyright © 2013 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: June 2013

Production Reference: 1110613

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham B3 2PB, UK.

ISBN 978-1-84968-982-3

www.packtpub.com

Cover Image by Suresh Mogre (suresh.mogre.99@gmail.com)

Credits

Author

Gohar Ahmed

Project Coordinator

Sneha Modi

Reviewers

Ferdinand Feenstra

Daniele Tosatto

Proofreader

Mario Cecere

Acquisition Editor

Kunal Parikh

Indexer

Rekha Nair

Commissioning Editor

Yogesh Dalvi

Graphics

Ronak Dhruv

Technical Editors

Prasad Dalvi

Pushpak Poddar

Production Coordinator

Manu Joseph

Copy Editors

Brandt D'Mello

Aditya Nair

Alfida Paiva

Laxmi Subramanian

Cover Work

Manu Joseph

About the Author

Gohar Ahmed is an all round expert from Lahore, Pakistan. He works with different technologies related to virtualization, networking, system securities, server-side application development, and on top of all deals with Voice Over IP technologies and unified communication protocols using SIP protocol on both open source and proprietary software and hardware. Aside from keyboard, screen, and servers, he loves to do painting and gardening. Gohar maintains his blog at <http://saevolgo.blogspot.com>.

I'd like to thank my parents, Lall Khan Nasir (late) and Sajida Shaheen, who encouraged me through all the hurdles and struggles in life. My sweet wife Sairah, who constantly reminded and helped me to publish this book, deserves all my love and gratitude. I'd like to mention my dear colleagues, Abdul Basit, Salman, Shahzad, Haroon, and Imran Iqbal, who provided me with their possible support in writing this book.

About the Reviewers

Ferdinand Feenstra is a Citrix Certified Architect and senior specialist for Microsoft environments, based in The Netherlands. He is working in the IT branch since 1998 and has experience in many complex environments with different customers in different functions.

His experience is categorized in build and design Citrix environments, implementations and migrations projects, and consultancy projects. Since he started working with Citrix in 2004, a new world of solutions, working on any device combined with a great user experience, has come his way. This makes IT more dynamic and easier to adapt for users. You can find his blog at www.CitrixGuru.net or check his tweets on Twitter, [@f_feenstra](https://twitter.com/f_feenstra).

This is his third review. He has already reviewed the books *Citrix XenServer 6.0 Administration Essential Guide*, *Packt Publishing* and *Citrix XenDesktop 5.6 Cookbook*, *Packt Publishing*.

Ferdinand works for Icento. Icento is a Citrix Partner Solution Advisor with the Silver status. Icento is also a V-Alliance member; the virtualization collaboration between Microsoft and Citrix. Icento is located in Rotterdam, The Netherlands, and delivers solutions for the desktop, unified communications, and virtualization and systems management. Icento delivers state-of-the-art ICT solutions for a broad set of international customers. You can find more information at www.icento.nl.

Daniele Tosatto is a senior systems engineer based in Venice, Italy. He is a Microsoft Certified IT Professional, Microsoft Certified Technology Specialist, Microsoft Certified Solutions Expert, Microsoft Certified Solutions Associate, and Citrix Certified Administrator, and has been working with Microsoft products since 2000 as a system administrator. In February 2008, he started working for the first Italian Citrix Platinum Partner. He is focused on Active Directory, application virtualization and delivery, and IT infrastructure management.

Daniele is the author of *Citrix XenServer 6.0 Administration Essential Guide*, Packt Publishing.

Visit his blog at www.danieletosatto.com.

www.PacktPub.com

Support files, eBooks, discount offers and more

You might want to visit www.PacktPub.com for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.PacktPub.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at service@packtpub.com for more details.

At www.PacktPub.com, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.



<http://PacktLib.PacktPub.com>

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why Subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

Free Access for Packt account holders

If you have an account with Packt at www.PacktPub.com, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Instant Updates on New Packt Books

Get notified! Find out when new books are published by following [@PacktEnterprise](https://twitter.com/PacktEnterprise) on Twitter, or the *Packt Enterprise* Facebook page.

Table of Contents

Preface	1
Chapter 1: Getting to Know Xen	5
Life without virtualization	5
What is virtualization?	6
Types of virtualization	6
History of Xen and XenServer	9
How XenServer works	9
Benefits of Citrix XenServer	9
Requirements for XenServer	10
System requirements	10
What is XenCenter?	10
System requirements	11
Summary	11
Chapter 2: Installing and Starting XenServer	13
Downloading and installing XenServer	13
Enabling virtualization in BIOS	14
Installing XenServer	14
Installing XenCenter	22
Connecting XenCenter and XenServer	23
Summary	24
Chapter 3: Licensing XenServer and XenCenter	25
XenServer licensing	25
Activating the free license	26
Activating the non-free license	28
Managing the XenServer host	30

Resource pools	34
Summary	35
Chapter 4: XenServer Storage Repositories	37
Storage technologies and protocols	37
Direct attached storage	38
Network attached storage	38
Storage area network	38
Storage protocols	39
Network file system (NFS)	39
Internet Small Computer System Interface (iSCSI)	39
Fibre Channel (FC)	40
Storage objects in XenServer	40
Storage repository (SR)	40
Virtual Disk Images (VDIs)	41
Physical Block Devices (PBDs)	41
Virtual Block Devices (VBDs)	42
Storage attributes	42
Creating a storage repository	43
Creating an NFS VHD storage repository	44
Creating a software iSCSI Storage Repository	45
Creating a hardware HBA storage repository	47
Creating a StorageLink repository	48
Creating an ISO library	49
Managing the storage repositories	51
Summary	51
Chapter 5: Playing with VMs	53
Virtual machines using templates	53
Selecting a template for a new virtual machine	54
Selecting the operating-system-bootable ISO	55
Selecting the host XenServer for a virtual machine	56
Selecting storage space to create the vHDD	57
Network interface and network connectivity	58
XenServer Tools	60
Creating new templates	62
Converting an existing VM to a template	62
Making templates from snapshots	65
Importing templates	66

Cloning VMs and templates	68
Preparing and cloning Windows VMs	70
Summary	71
Chapter 6: Managing Virtual Machines	73
<hr/>	
Managing storage, memory, and networking	73
Managing VM CPU and memory	74
Dynamic memory control	76
Managing virtual machine storage	77
Managing network interfaces	79
Importing and exporting virtual machines	80
Exporting a virtual machine	80
Importing a virtual machine	82
Virtual machine snapshots	84
Types of snapshots	85
Disk-only	85
Quiesced	85
Disk and memory	85
Introduction to vApps	85
Summary	88
Chapter 7: Networking in XEN	89
<hr/>	
The XenServer networking concept	89
Network types and objects	90
The networking pool	90
Network types	90
Using virtual networks	92
VLANs with virtual machines	95
NIC bonds	97
Creating an NIC bond	98
Deleting an NIC bond	99
Virtual switches and routers	100
Summary	100
Chapter 8: Advanced Concepts	101
<hr/>	
Introduction to XenConvert	101
System requirements	102
Converting a physical machine to a virtual machine	102
Preparing the host machine	102
High Availability	106
Failure planning	107

Table of Contents

Distributed vSwitch Controller	108
Workload Balancing	108
Role-based administration control	110
Roles	111
Summary	112
Index	113

Preface

Virtualization has made a big name in today's technologies and it is a growing market for vendors as well as engineers. The main concept behind this book is to make anyone with basic IT administrative skills a virtualization-ready administrator. This book mainly covers the Citrix XenServer and virtualization concepts in a balanced mode. Readers can go through the entire book and either theoretically know what to do and how to start with virtualization using XenServer or can start creating their own virtualization environment following the examples in this book. The content and language of the book has been kept as simple as possible so that users can follow and understand easily.

What this book covers

Chapter 1, Getting to Know Xen, covers basic conceptual knowledge about the virtualization, brief history of XenServer, and introduction to Xen and XenServer.

Chapter 2, Installing and Starting XenServer, discusses topics such as installing XenServer and XenCenter, and having the XenCenter connected with XenServer host for management purposes.

Chapter 3, Licensing XenServer and XenCenter, covers licensing options in XenServer and overview of XenCenter. Different types of XenServer licenses have improved features, so we'll see how to install the license.

Chapter 4, XenServer Storage Repositories, discusses the different storage types and their benefits for the XenServer, and explains how to create ISO libraries.

Chapter 5, Playing with VMs, introduces templates for virtual machines, cloning, and creating virtual machines using different methods.

Chapter 6, Managing Virtual Machines, illustrates how to manage virtual machine's resources such as CPU, memory, storage, and networking.

Chapter 7, Networking in XEN, discusses the networking concepts used in XenServer, VLANs creation, and networking options in XenServer.

Chapter 8, Advanced Concepts, introduces advanced technologies such as XenConvert, high availability, RBAC, and distributed VSwitch.

What you need for this book

This book is designed to keep the conceptual and practical implementations at a balanced level, hence anyone interested in starting with XenServer or virtualization technologies can benefit from this book. However, for the practical person, it is recommended to have a virtualization technology-enabled system with minimum recommended specifications as mentioned in *Chapter 1, Getting to Know Xen*.

For XenCenter, a Windows-based machine is recommended.

XenServer installation requires its ISO to be downloaded and burnt to appropriate CD/DVD to begin with the installation.

XenServer ISO can be downloaded for free from <http://www.citrix.com/products.html>.

Who this book is for

If you want to become a virtualization expert, or if you've basic knowledge about virtualization and want to get started practically along with the concepts behind the technology, don't hesitate to go through this book. This book covers all the mandatory details to prepare a system's administrator for XenServer virtualization.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles are shown as follows: "The Xen hypervisor launches the most privileged domain, `dom0`."

A block of code is set as follows:

```
watchdog_timeout=300 cpuid_mask_xsave_eax=0 lowmem_emergency_pool=1M
crashkernel=64M@32M console=vga vga=mode-0x0311 dom0_max_vcpus=1-4
--- /boot/vmlinuz-2.6-xen root=LABEL=root-fegvhmfg ro xencons=hvc
console=hvc0 console=tty0 quiet vga=785 splash --- /boot/initrd-2.6-
xen.img
```


When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in bold:


```
label xe
# XenServer
kernel mboot.c32
append /boot/xen.gz mem=1024G dom0_mem=952M,max:952M
```

Commands will be shown like this:

```
[root@xens ~]# cat /proc/xen/balloon
Current allocation:  974848 kB
Requested target:   974848 kB
```

New terms and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "Click on **XenServer** under the category **Cloud Platforms**."

 Warnings or important notes appear in a box like this.

 Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book – what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to feedback@packtpub.com, and mention the book title via the subject of your message.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on www.packtpub.com/authors.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books – maybe a mistake in the text or the code – we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting <http://www.packtpub.com/submit-errata>, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from <http://www.packtpub.com/support>.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at copyright@packtpub.com with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at questions@packtpub.com if you are having a problem with any aspect of the book, and we will do our best to address it.

1

Getting to Know Xen

Virtualization and cloud are the latest keywords wherever we go in technical world these days. In this chapter we will discuss the following items:

- What is virtualization anyway?
- Different types of virtualization
- History of XenServer
- How it works and its benefits
- Citrix XenCenter
- System requirements

Life without virtualization

Back in the days when I started my job, I had to do a lot of installations, learn new software, and get familiar with their behavior on different operating systems. I spent most of my time installing a particular operating system on the one and only big server I was allocated for this learning process. I progressed slowly, but then one day I had to ask my senior manager to provide me with at least one more medium-scale server so that I could do some test work simultaneously. Unpredictably and unfortunately, I was granted an old desktop.

I was happy; I did some quick network cabling, adjusted power, installed a fresh new operating system on it, spent more than a couple of hours on it to get updated from the Internet, and by the end of the day I was really happy for what I had done. This continued five to six more times before I got tired of all the hectic exercise just to create a testing development machine.

What did I miss there? I wasted a lot of good time; I wasted time in setting it up; I wasted time in doing the cabling and networking; I wasted time even installing the OS and updating it from the Internet. On top of all that, I wasted both the physical space and the hard disk space. Of course, I wasted the processing power of the server and desktop.

Then I came across the idea of *hardware virtualization* while browsing through the Internet. And so my journey with virtualization began.

What is virtualization?

So what exactly is virtualization? In very basic technical language, it means splitting one physical resource into as many logical pieces as possible. One slot of RAM with 8 GB of capacity can be split into smaller pieces of varying sizes. Similarly, the hard disk can be virtually used as multiple hard disks and so it is the case with processors. This can be viewed as hardware partitioning.

Virtualization should not always be thought of as splitting resources down; rather it can be thought of as adding up resources. One simple example is using multiple hard disks in a virtualized server as one big disk split into many others.

This partitioned hardware can then be organized to create virtual machines; a controlling framework is obviously required for this. We are specifically talking about hardware virtualization.

Types of virtualization

There are three such types of hardware virtualization frameworks: operating-system-level virtualization, paravirtualization, and full virtualization. A main server with such a framework running on its own is known as a **host** and the virtual machines created on that host are termed as **guests**.

Full virtualization is a complete simulation of hardware to allow guest operating systems to be used unmodified.

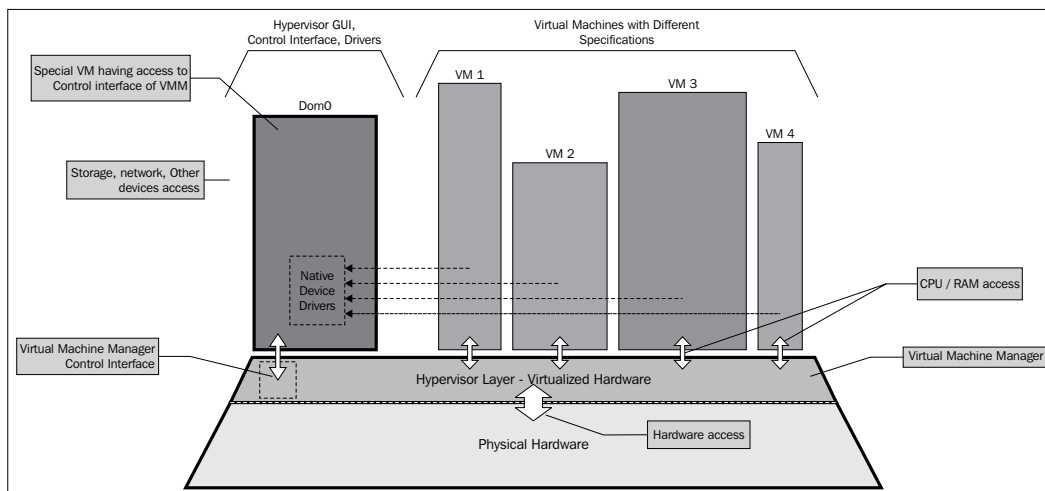
In paravirtualization, the hardware environment is not simulated; however, the guest programs are specifically modified to run in their own isolated domains, as if they are running on a separate system.

In between full virtualization and paravirtualization, there also exists partial virtualization, which is a blend of both full virtualization and paravirtualization. Not all hardware is simulated from the host server and so some programs or a part of the guest operating system needs to be modified to work in this environment.

Operating-system-level virtualization has a base operating system whose kernel is modified to run multiple separate user operating systems. These separate user operating systems are known as containers, and look and feel like a real server. The base operating system is the one responsible for hardware resource management between the guest operating systems.

The firmware or software that lets the user create a virtual machine is known as **hypervisor** or **virtual machine manager**. In general, there are two broader categories of **hypervisors**.

A Type-1 hypervisor is the only layer between the host machine hardware and the guest operating systems. It is also commonly referred to as a bare-metal hypervisor. Popular software in this category are Microsoft's Hyper-V hypervisor, VMWare's ESX, and Citrix XenServer. These are mostly used in production server virtualization and we will focus on Citrix XenServer.



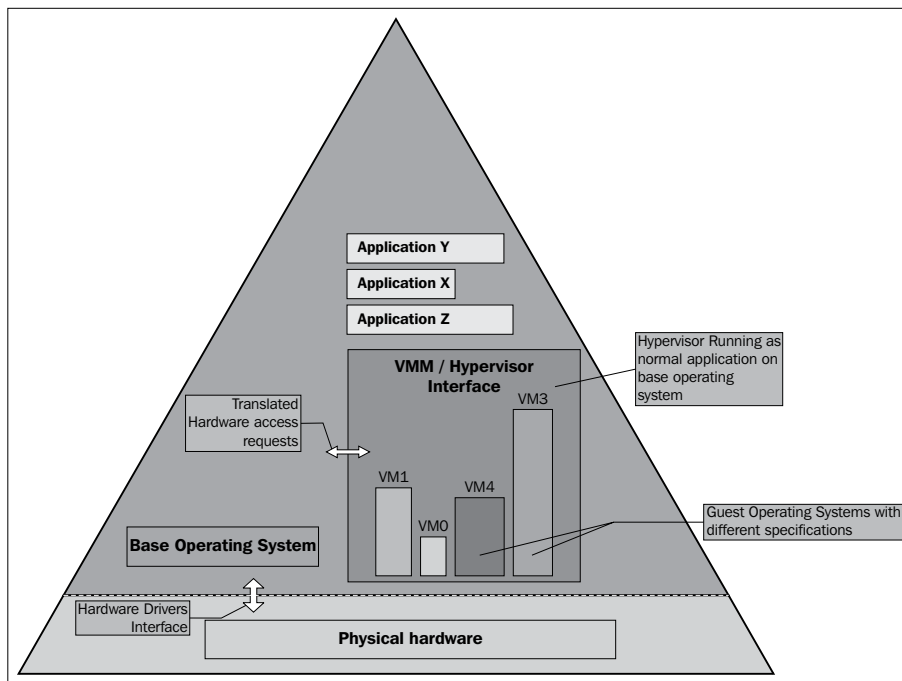
Bare-metal XenServer hypervisor architecture

Type-2 hypervisors are the software that run on an already running base operating system as an application and then let users create smaller virtual machines. These are mostly used in regular desktops for common tasks. I, for example, prefer to use Oracle's VirtualBox and VMware's Workstation. Nowadays, Windows 8 (Professional or Enterprise) offers the Hyper-V hypervisor as part of the operating system.



In the mid 1960s, the IBM Watson Research Center was home to the M44/44X Project, whose goal was to evaluate the then emerging time-sharing system concept. The architecture was based on virtual machines.

IBM had provided an IBM 704 computer, a series of upgrades (such as to the 709, 7090, and 7094) and access to some of its system engineers to MIT in the 1950s. It was on IBM machines that the **Compatible Time Sharing System (CTSS)** was developed at MIT. The supervisor program of CTSS handled console I/O, scheduling of foreground and background (offline-initiated) jobs, temporary storage and recovery of programs during scheduled swapping, monitoring of disk I/O, and so on. The supervisor had direct control of all trap interrupts.



Hypervisor architecture

Our specific requirement is Type-1 bare-metal para-virtualization so that we can have maximum hardware utilization. There are numerous comparisons between the different virtualization technologies available over the Internet. Different hypervisors have been compared and we have decided to use Citrix XenServer 6.

History of Xen and XenServer

Before going into further details, some history should be revised in brief.

Xen originated as a research project at the University of Cambridge as part of a project called *The XenServer wide-area computing project*. The name derives from the Greek word *xenos*, which means foreign or unknown. In 2003, the first public release of Xen was introduced. Xen is a native or bare-metal hypervisor. It runs in a more privileged CPU state than any other software on the machine.

How XenServer works

The Xen hypervisor launches the most privileged domain, `dom0`. This is the only virtual machine that by default has direct access to hardware and can manage user domains, such as `domU`, which are the unprivileged guest virtual machines. The hypervisor is responsible for the memory management and CPU scheduling of the guest machines.

XenServer is built on an open source Xen hypervisor. It's like an operating system and uses a combination of paravirtualization and hardware-assisted virtualization. XenServer allows its guests' operating systems to know that they are being run on a virtualized environment and hence the guests' operating systems sometimes need to be modified to reduce performance penalties. For any operating system that can't be modified for paravirtualization, for example, Microsoft Windows, hardware-assisted technologies are utilized.


Benefits of Citrix XenServer

Citrix XenServer is installed directly on bare-metal servers, requiring no dedicated host operating system. This increases server and storage utilization and reduces costs of equipment, power, cooling, and physical space. Since guest machines do not directly communicate with hardware, running virtual machines can be migrated to new servers with no service interruption, allowing essential workloads to get the necessary resources and enable zero-downtime maintenance. More sophisticated features include automatic recovery from hardware failure and failover capabilities in disaster situations.

A complete feature list and comparison of Citrix XenServer editions is available at <http://www.citrix.com/products/xenserver/features/editions.html>.

Requirements for XenServer

Since XenServer is going to host all the virtual machines, it's essential that it is capable and powerful enough to carry the load. The requirements are discussed in the following sections.

 A Hardware Compatibility List is available at <http://hcl.xensource.com>. It is recommended, but not mandatory, to use the hardware from this list.

System requirements

The following are the minimum XenServer host system requirements:

- 64-bit x86 server-class system
- CPU: 1.5 GHz minimum; 2 GHz or faster and multi-core recommended
- Intel VT or AMD-V required for support of Windows guests
- 2 GB to 1 TB physical memory
- Up to 64 logical processors
- 100 Mb/s or faster NIC
- Up to 16 physical NICs
- Local, Fiber Channel, or SAN boot disk with 16 GB of minimum space; 60 GB or more recommended
- Fiber Channel, iSCSI, or NFS LUN with 356 MB of minimum space for high availability (HA) heartbeat storage repository

What is XenCenter?

XenCenter is the management GUI for XenServers that runs on Windows. This is the front end that lets users configure new virtual machines and enables full virtual machine installation, configuration, administration, and lifecycle management. Using this tool we configure the remote storage and manage networks, including VLANs and internal networks and bonded and dedicated NICs.

XenCenter displays the VMs' performance stats and lets us take snapshots of guest machines and provide access to the virtual machines' consoles. These are just some basic functions of XenCenter.

There are a lot of things we will be doing in the following chapters using this tool.

System requirements

XenCenter can be installed on any system running the Windows operating system. The following are the minimum requirements of a XenCenter machine:

- XenCenter Client
- x86-based system
- Microsoft Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008, Windows Vista, or Windows 7 (any edition or version)
- .NET Framework 2.0 SP1 or later
- CPU speed: 750 MHz minimum; 1 GHz or faster recommended
- RAM: 512 MB minimum
- Disk space: 100 MB minimum
- Network interface card

Summary

So, we've learned a lot of basic concepts about virtualization, Xen, and XenServer. As mentioned earlier, our preferred choice was XenServer. Deploying and playing with XenServer will be covered in the following chapters. We will also discuss types of virtualization, types of hypervisor (virtual machine manager), Xen hypervisor domains, and Citrix XenServer components.

In the next chapter we'll go through the steps to install XenServer and connect to it with XenCenter.

2

Installing and Starting XenServer

In this chapter we'll see the steps to download, install, and configure our first XenServer. Although connecting XenServer with XenCenter is a small thing, it's the main thing we'll be dealing with in all the chapters ahead.

Downloading and installing XenServer

Downloading Citrix XS is easy. Just go to the Citrix **Products** page at <http://www.citrix.com/products.html> and click on **XenServer** under the category **Cloud Platforms**; this will take you to the **Products** home page. Next, click on **Try it**, and on the following page, select how you'd like to start working with XenServer.

I chose the **Do-it-yourself** approach and started a premium XenServer with advanced features trial. They'll ask for some quick user information, which will be verified. A trial number will be generated there as well.

Once the ISO download is complete, just use the ISO for the network PXE boot installation method and have XenServer installed from over the network, or burn a DVD and boot up XenServer with it. It's easier to choose the second option if XenServer is physically available. It is strongly recommended that we enable the virtualization extension in BIOS.

Enabling virtualization in BIOS

Perform the following steps to enable virtualization in BIOS. Please note that many of the steps may vary depending on motherboard, processor type, chipset, and OEM. Refer to your system's accompanying documentation for the correct information on configuring your system.

1. Reboot the computer and open the system's BIOS menu. This can usually be done by pressing the *Delete* key, the *F1* key, or *Alt + F4*, depending on the system.
2. Open the **Processor** submenu. The processor settings menu may be hidden in the **Chipset**, **Advanced CPU Configuration**, or **Security** tabs.
3. Enable **Intel Virtualization Technology** (also known as Intel VT) or **AMD-V**, depending on the brand of the processor. The virtualization extensions may be labeled **Virtualization Extensions**, **Vanderpool**, or various other names, depending on the OEM and system BIOS.
4. Enable **Intel VTd** or **AMD IOMMU**, if the options are available; they are used for PCI passthrough.
5. Select **Save & Exit**.

Installing XenServer

Once the boot media is detected as **CD/DVD ROM**, a prompt appears with the XenServer **Install** menu. For the normal installation procedure, just press *F1*; if you need to go in for advanced installation options, press *F2*.

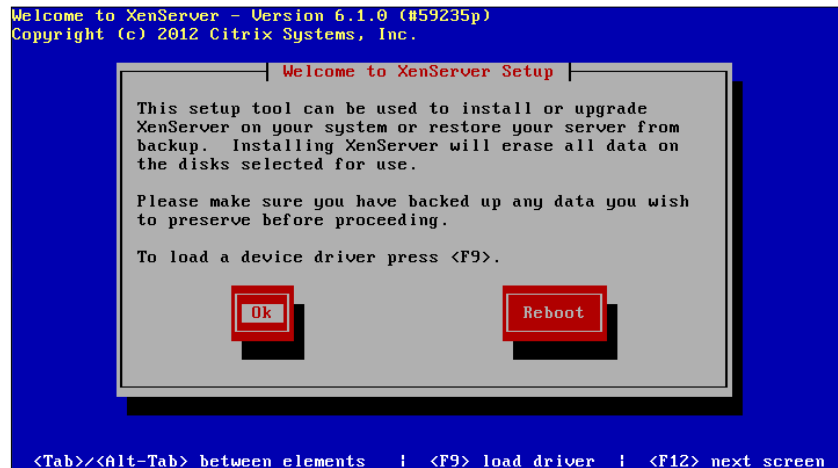


The advanced installation menu

Continue with the standard installation procedure since we don't need any of the discussed advanced options now.

The following screens will look familiar to anyone who has installed any Unix operating system, such as Ubuntu and CentOS.

The first screen inquires about the keyboard type; the next one warns us to save any data before the installation erases everything.



Data backup and driver loading screen

The second screen is important if we've any other devices that we may need to load drivers for.

On pressing *F9*, it will give three options to load the supplemental pack, which contains the drivers **Local media** (CD/DVD ROM), **HTTP or FTP**, and **NFS**.

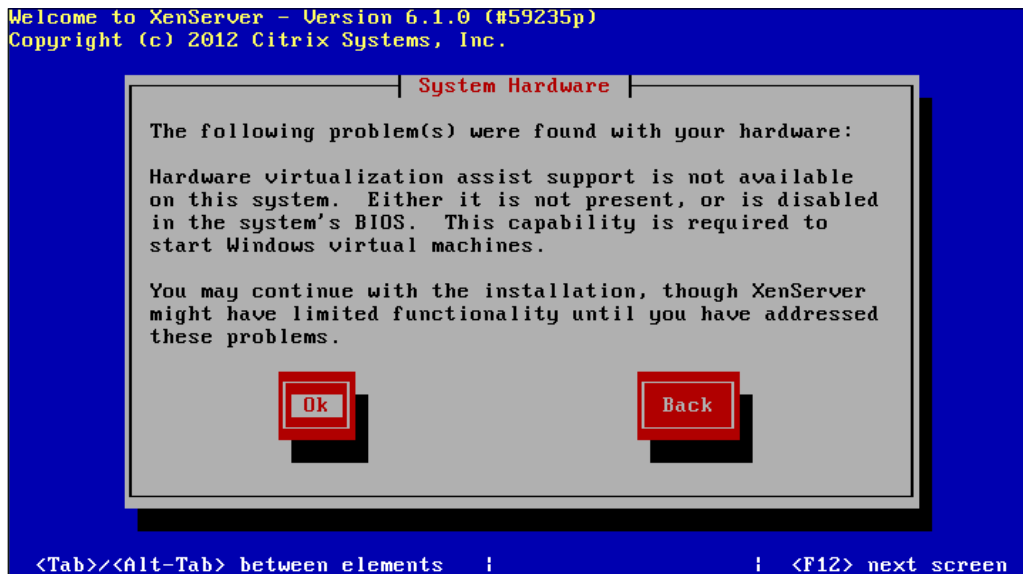


What are supplemental packs?

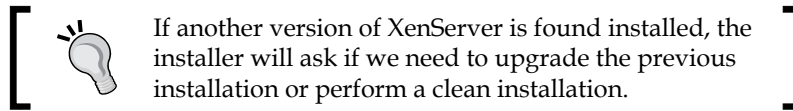
Supplemental packs are used to modify and extend the functionality of a XenServer host by installing software into the control domain, `dom0`. Users can add supplemental packs either during initial XenServer installation or at any time afterwards.

Supplemental packs are created using the XenServer **Driver Development Kit (DDK)**. This has been extended to allow the creation of not only supplemental packs containing just drivers (also known as driver disks) but also packs containing userspace software to be installed into `dom0`.

I did not need any extra drivers to be loaded, so I proceeded to the next step and accepted the **End User License Agreement (EULA)**. After this, we may see any hardware issues that XenServer installation has detected. An example is **Support for Hardware virtualization support not detected**.

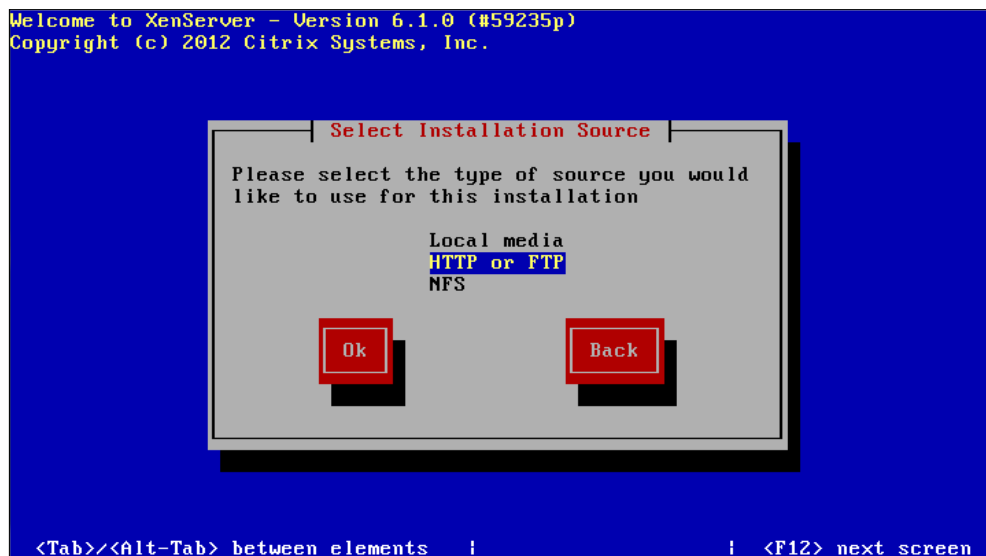


Hardware issue detected



The next menu will ask whether supplemental packs need to be installed. After that, the virtual storage disk selection will be made. Choose the connected hard disks that we need to use as storage space for XenServer and virtual machines.

Next, select the source of installation; this again provides us with three options: **Local media**, **HTTP or FTP**, and **NFS**.

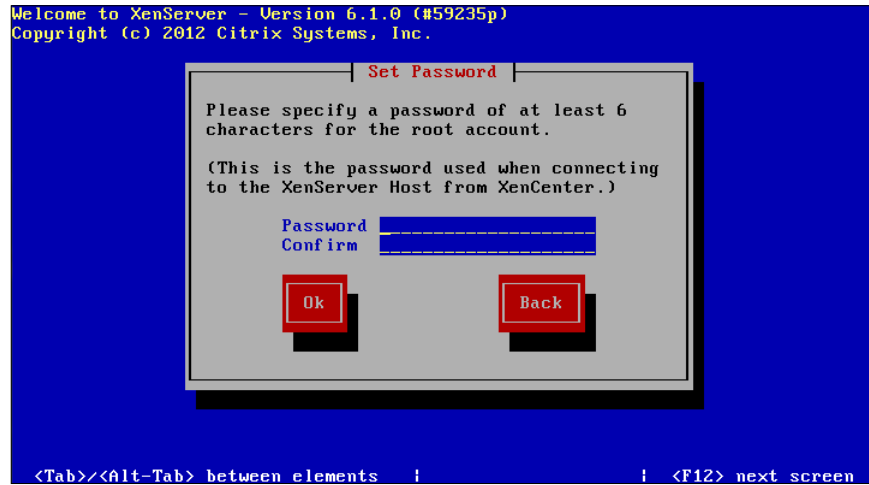


Selection of the installation source medium

For the other installation options, that is, HTTP, FTP, or NFS, networking configurations will be set up. The location URL or path of the source files via HTTP, FTP, or NFS should be provided in the installation menu along with username/password, if any. I had DVD as the installation medium, so I chose **Local**.

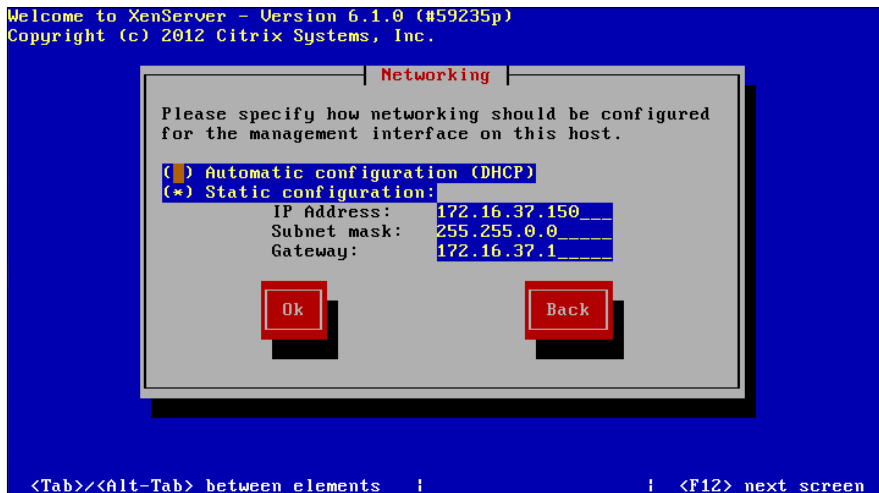
Then, it will ask you to choose whether a supplemental pack needs to be installed. I chose **No**. On the next screen, we can choose to verify or skip the installation source.

The next step will ask us to enter the root password of the XenServer host. This password will be used in XenCenter to get connected; also, this password will be used to log on for any direct console-based configurations.



Root password for xsconsole

Once the password is set, we will need to configure the network interface. Either set it to use DHCP and set the IP address automatically, or use the static IP address manually. It's important to use a short hostname in manual configurations, as giving a **fully qualified domain name (FQDN)** may cause external authentication to fail.

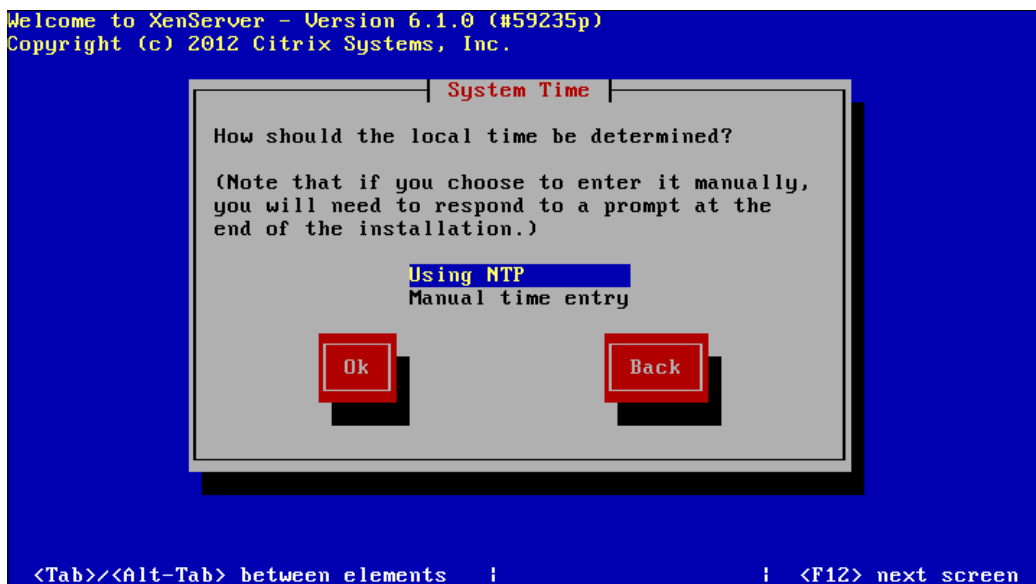


The network configuration menu

If we choose to give network configurations via DHCP, DNS and hostname configurations can be provided via DHCP or manually in the next menu. I had all the networking settings done via DHCP.

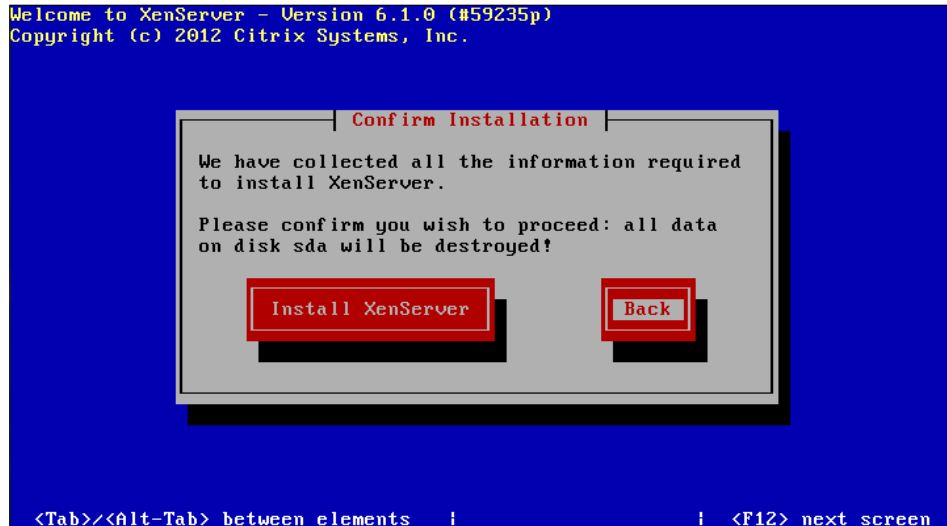
After the time zone is determined, select the geographical area. The major cities of that region are enlisted. Select the region and proceed.

Once a region is selected, we are asked to configure system local time. Here, you have two options: either get the date and time by a list of NTP servers (that are entered manually or by DHCP) or enter the date time string manually later on during installation.



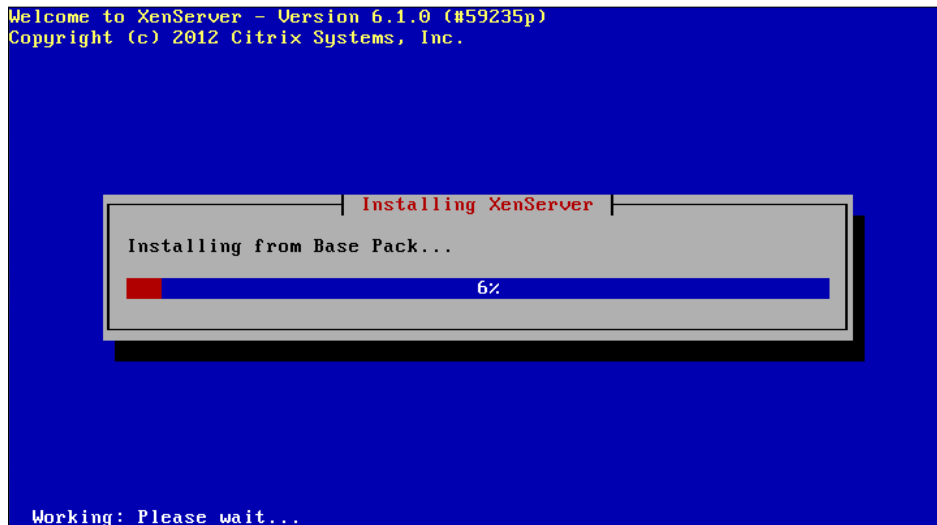
System time configuration

Once time settings are configured, we are asked to confirm all the inputs to the previous menus. Select **Install XenServer** and press *Enter*.



The information confirmation menu

Installation will begin afterwards. Once everything is done, remove the installation medium from the CD/DVD ROM and click on **OK**.



The installation progress bar

Once the medium is removed and the server is rebooted, one should see the XenServer loading screen, and finally, a XenServer configuration console will appear.

```

XenServer 6.1                               15:51:21                               root@saevolgo
----- Configuration -----
Network and Management Interface
Configure Management Interface
Display DNS Servers
Network Time (NTP)
Test Network
Display NICs
Emergency Network Reset
Open vSwitch

Configure Management Interface
Device           eth0
MAC Address      08:00:27:60:8a:ec
DHCP/Static IP   Static
IP address       172.16.37.150
Netmask          255.255.0.0
Gateway          172.16.37.1
Hostname         saevolgo

NIC Vendor
Advanced Micro Devices [AMD]

NIC Model
79c970 [PCnet32 LANCE]

<Esc/Left> Back <Up/Down> Select    <Enter> Reconfigure <F5> Refresh

```

The XenServer xsconsole interface


Note down the IP address and other details, if required. This screen is **xsconsole**. To access local shell from xsconsole, press *Alt + F3*; to exit from the shell back to xsconsole, press *Alt + F1*.

That is it. Our very first XenServer is up and running. Now we need to move to XenCenter, the GUI interface of our hypervisor.

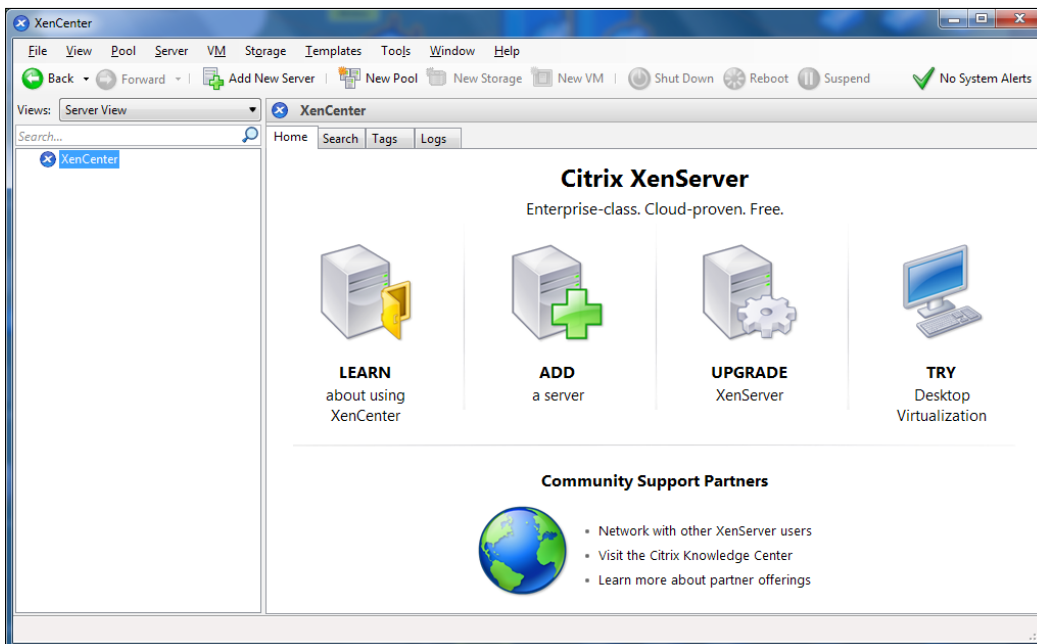
Installing XenCenter

XenCenter is very easy to install. We now know the IP address of our XenServer, which is 172.16.37.150. Open up a browser on the Windows machine that will be hosting XenCenter, write down this IP address in the URL bar, and press *Enter*. Ensure that this machine has network connectivity with XenServer.

A plain page will appear, linking to the installer for XenCenter. Click on **XenCenter Installer**, and it will begin downloading the executable installer. After that, it's as simple as installing any application in Windows; click on **Next** and follow the onscreen instructions. It's just a three-step installation process.

 **Another way of installing XenCenter:**
XenCenter can be found in the same installation DVD in the folder `client_install`; the executable file to run is `XenCenter.msi`.

Start XenCenter, where we will be spending most of our time from now onwards.



The XenCenter application

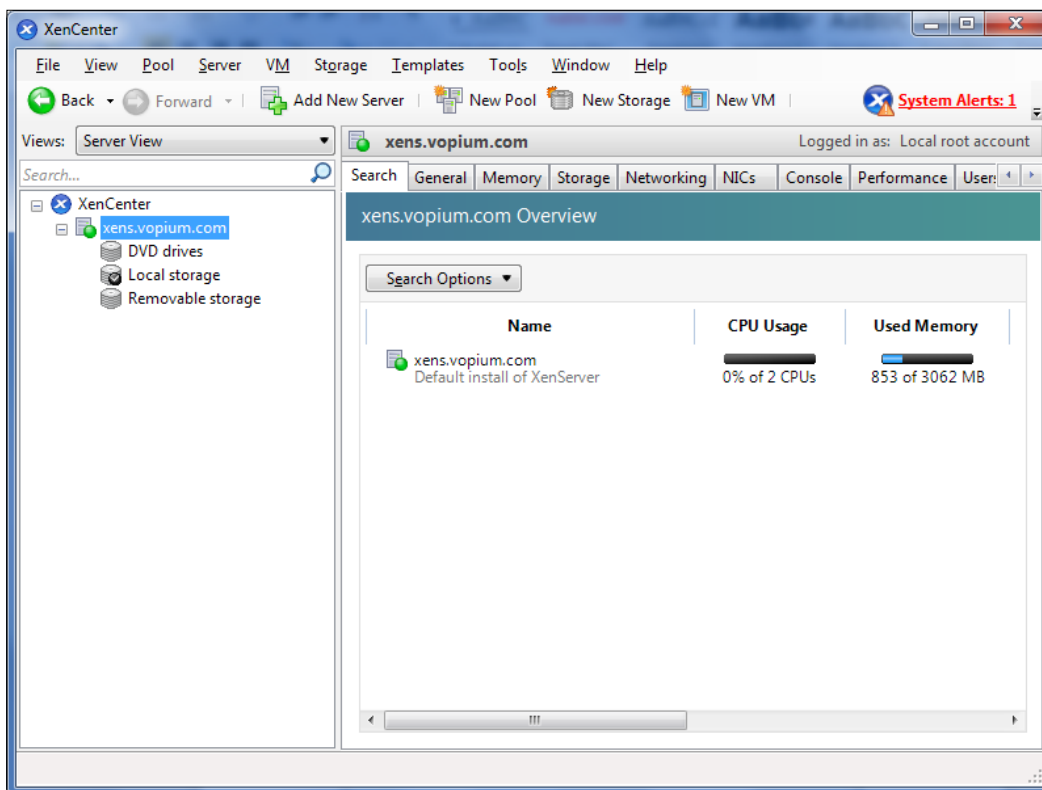
Connecting XenCenter and XenServer

With reference to the previously seen XenCenter application, in order to connect XenCenter with the XenServer host, click on **ADD**.

As soon as the server IP address and root password are entered, XenCenter will connect to the XenServer host server, authenticate the root user, and synchronize with that host machine.

For the very first time, XenCenter will ask for license activation. Licensing will be seen in detail in the next chapter. We can close that pop up and can insert the license later on.

That is it; we are done connecting XenCenter with our first XenServer host.



XenCenter with a server added

Summary

We went through the following topics in this chapter:

- Enabling virtualization support from BIOS
- Installing XenServer and XenCenter
- Interconnecting XenCenter and XenServer

In the next chapter, we will take a look at different types of licenses for XenServer and their different features. We will activate a free license and start using XenServer right away.

3

Licensing XenServer and XenCenter

This chapter is about the different versions of XenServer and their features. We'll look at the following:

- Licensing XenServer
- Basic configuration management using XenCenter
- Using xsconsole
- XenServer resource pools

XenServer licensing

There are four editions of Citrix XenServer: Free, Advanced, Enterprise, and Platinum. The differences are given here. We can start using Citrix XenServer for free for as long as we want. Initially, Citrix gives a 30-day limit to activate XenServer.

The following features are available in all four editions:

- XenServer hypervisor
- Conversion tools
- Management integration with Microsoft System Center VMM
- Resilient distributed management architecture
- VM disk snapshot and revert
- XenCenter management console
- XenMotion live migration

The following features are available in all non-free editions; that is, Advanced, Enterprise, and Platinum:

- Automated VM protection and recovery
- Live migration with Storage XenMotion™
- Distributed virtual switching
- Heterogeneous pools
- High availability
- Memory optimization
- Performance alerting and reporting

The following features are available in the Enterprise and Platinum editions:

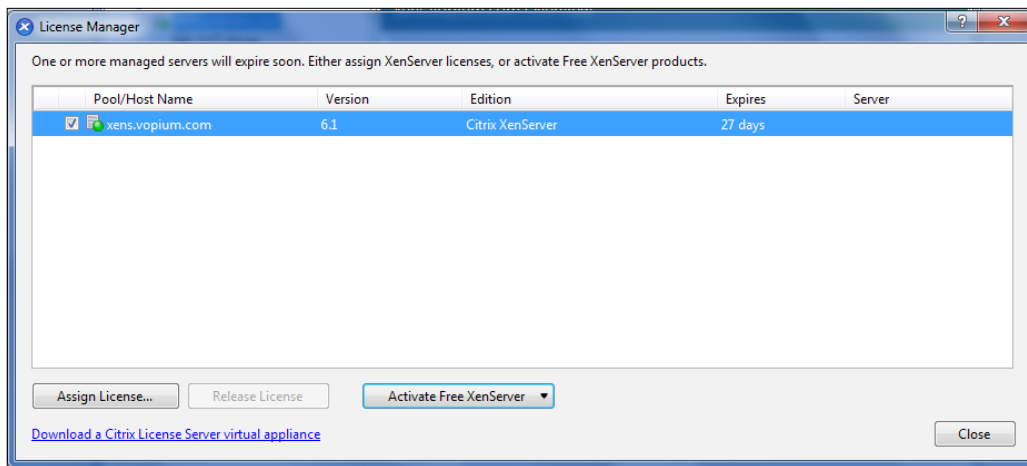
- Dynamic workload balancing
- GPU passthrough
- Host power management
- IntelliCache
- Live memory snapshot and revert
- Provisioning services (virtual)
- Role-based administration
- StorageLink
- Web management console with delegated admin

The Platinum edition has these extra features as compared to the Enterprise edition:

- Provisioning services (physical)
- Site recovery

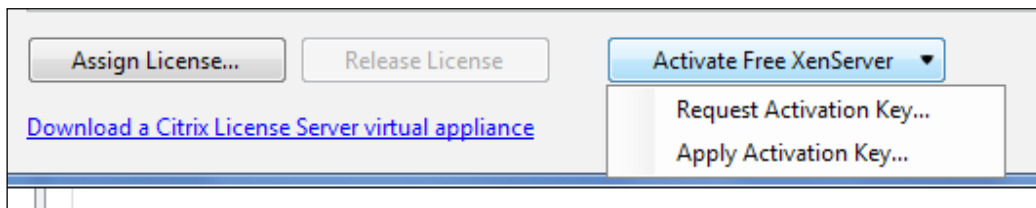
Activating the free license

Starting up with the basic Free edition is pretty simple without activating the free license every time. Open XenCenter; it automatically pops up the **License Manager** window. To request for a free activation key, just select the XenServer host you wish to request a key for and select **Activate Free XenServer**.



Automatic License Manager pop up

Click on **Activate Free XenServer** and select **Request Activation Key...**

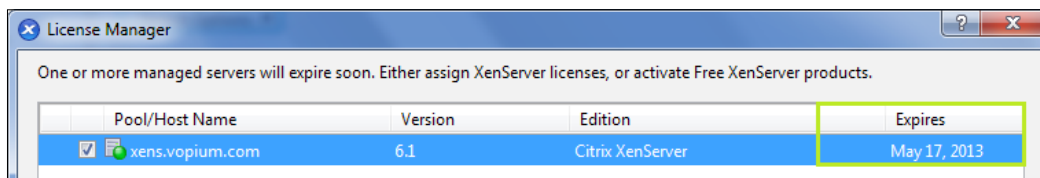


Automatic License Manager

This will open up a new browser page that will gather user information such as name, address, e-mail contact, and other things. Once the form is submitted, Citrix automatically sends the license file to the specified e-mail address. The extension of the license file is `.xslic`.

As soon the license file is downloaded, apply the application key to the License Manager window. If you have multiple XenServer hosts added on the same XenCenter, don't forget to select the host for which you're applying this license.

As soon as we browse and select the license, the expiry status of the host XenServer will change and should be something like the following screenshot. The free license gives a one year time period before we need to request another free license.



Free license activated

We applied a license that I requested on May 16th, and hence we now have a validation date of one year.

In case the XenCenter Windows machine has no Internet connectivity, we are given the option to save the free license request in a file locally, and then using the same request file later on, or from some other computer, submit the license request file to <https://activate.vmd.citrix.com>.

Activating the non-free license

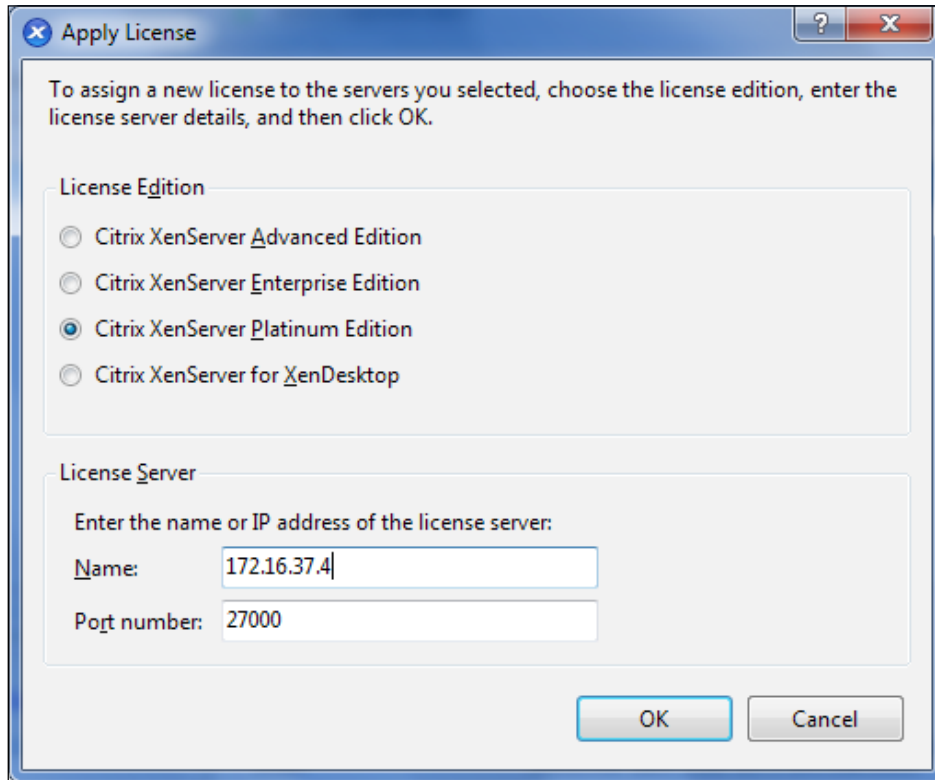
To activate a purchased license, all we need is a License Server. The License Server can be installed on the XenServer host. Installing the License Server is easy. It is a ready-to-go virtual machine that can be imported as a virtual appliance on XenServer. The virtual appliance is available from the Citrix downloads page, <http://support.citrix.com/article/CTX124501>. The License Server can be shared across all Citrix products. All the license files from Citrix should be uploaded into the License Server. This will be a central license repository.

While licensing must be configured for each XenServer host, the XenCenter License Manager allows us to apply the same settings to multiple hosts at once. Licenses are managed and controlled from XenCenter by applying or releasing a license from the XenServer hosts.

To apply or release a license, just open the XenCenter interface and click on **Tools** in the menu bar and select **License Manager**.

The **License Manager** window will pop open as shown in the following screenshot. Select the server we need to apply/release the license to. In case we are applying a license, a new window will ask us which edition we are applying a license to.

Also, we need to provide the address of the License Server. The installed License Server listens on port 27000 by default for any incoming connections. We can change this port for the License Server.



Applying license

After a license is checked out by a XenServer host, the host and the License Server exchange "heartbeat" messages every five minutes to indicate to each other that they are still up and running. If the XenServer and the License Server fail to send or receive heartbeats (for example, due to problems with the License Server hardware or software, or network failures), XenServer enters into the licensing grace period of 30 days and licenses itself through cached information.

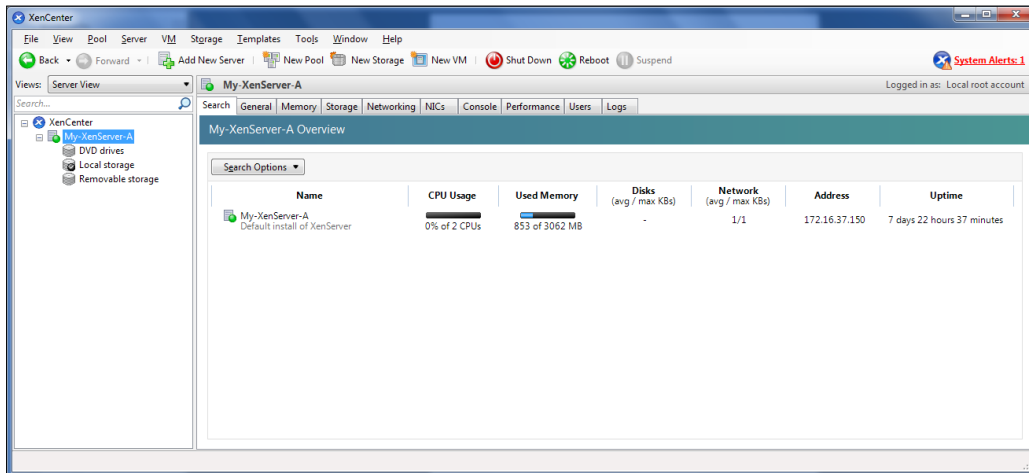
If the grace period expires, the VMs running on that host do not stop. However, we cannot start new VMs on that host until its connection is restored with the License Server.

Managing the XenServer host

We have our first host server connected with XenCenter (done in the previous chapter). Now let's focus on managing the XS host from XenCenter. It is already known that we use XenCenter to manage the virtual machines created in the host, but here we are looking at XenServer itself; that is, what can be done with XenServer.

As soon as we connect XenCenter with the host server, it will show us a general summary of the server. We can see what percentage of the host CPU is being utilized. The memory utilization, disk and network average, and maximum speeds are shown.

All the host servers connected to XenCenter display their general stats here.

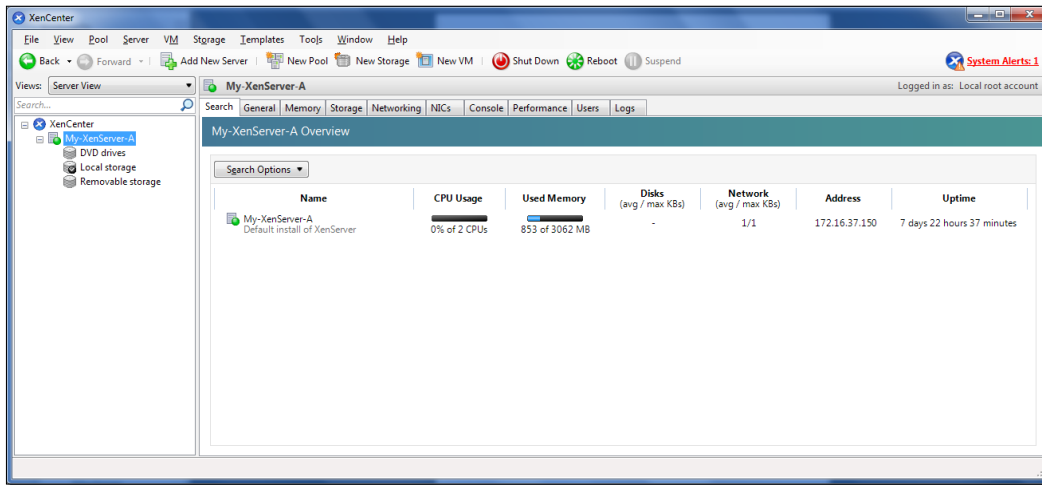


General information

The **General** tab on the right-hand side displays all the properties of the XenServer hardware as well as the XenServer version and license information.

In the **Storage** tab, we can add a new **Storage** pool for XenServer, which will be discussed in the next chapter.

Another important tab for us is **Networking**; in this tab, we can create virtual networks; but most importantly we can also change the physical management network interface configurations as well.



Network interface management

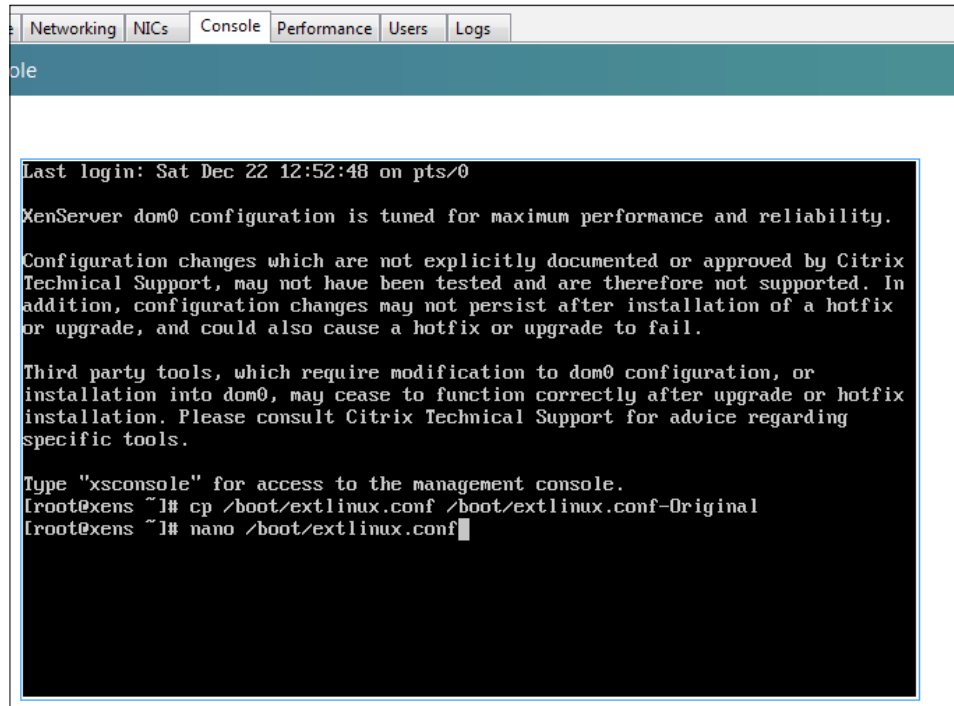
The **NICs** tab provides us with the option to create bonding between the physical NICs.

The **Console** tab gives us root access to the XenServer host. We can administer everything from the console as well. As an example, let's change the memory used by the XenServer control domain `dom0`.

The two major components of XenServer using memory are the Xen hypervisor and the Control domain. The hypervisor uses a fixed amount of memory, set to 128 MB. The Control domain uses a variable amount of memory; the minimum is 200 MB and the maximum is 752 MB. By default, XenServer might utilize memory from 328 MB to 880 MB.

The maximum memory utilization is defined in the file `extlinux.conf` located in the `/boot` folder. We can maximize this value to up to 2.94 GB. We may want this when we have a large number of virtual machines running on XenServer.

To change the maximum memory, let's go to the console and open up the file `extlinux.conf`. Make sure to create a backup copy of the original file.



The screenshot shows a terminal window with tabs for Networking, NICs, Console, Performance, Users, and Logs. The console output includes a login message, a warning about configuration changes, and instructions for using the management console. The user has executed the following commands:

```
[root@xens ~]# cp /boot/extlinux.conf /boot/extlinux.conf-Original
[root@xens ~]# nano /boot/extlinux.conf
```

XenServer console

See the highlighted text in the file, which is changed from 752 to 952 as an example.

```
label xe
  # XenServer
  kernel mboot.c32
  append /boot/xen.gz mem=1024G dom0_mem=952M,max:952M watchdog_
timeout=300 cpuid_mask_xsave_eax=0 lowmem_emergency_pool=1M
crashkernel=64M@32M console=vga vga=mode-0x0311 dom0_max_vcpus=1-4
--- /boot/vmlinuz-2.6-xen root=LABEL=root-fegvhmfg ro xencons=hvc
console=hvc0 console=tty0 quiet vga=785 splash --- /boot/initrd-2.6-
xen.img

label xe-serial
  # XenServer (Serial)
  kernel mboot.c32
```

```

append /boot/xen.gz com1=115200,8n1 console=com1,vga mem=1024G dom0_
mem=952M,max:952M watchdog_timeout=300 cpuid_mask_xsaves_eax=0 lowmem_
emergency_pool=1M crashkernel=64M@32M dom0_max_vcpus=1-4 --- /boot/
vmlinuz-2.6-xen root=LABEL=root-fegvnmfg ro console=tty0 xencons=hvc
console=hvc0 --- /boot/initrd-2.6-xen.img

```

Save, exit, and reboot XenServer.

After rebooting the server, reconnect XenCenter with the host and visit the console again.

The maximum usable memory size can be confirmed by the following command on xsconsole:

```

[root@xens ~]# cat /proc/xen/balloon
Current allocation:  974848 kB
Requested target:   974848 kB
Minimum target:     229376 kB
Maximum target:     983040 kB
Low-mem balloon:    0 kB
High-mem balloon:   0 kB
Driver pages:       4096 kB

```

Previously, before restarting the XenServer, it was:

```

[root@xens ~]# cat /proc/xen/balloon
Current allocation:  770048 kB
Requested target:   770048 kB
Minimum target:     203776 kB
Maximum target:     778240 kB
Low-mem balloon:    0 kB
High-mem balloon:   0 kB
Driver pages:       4096 kB

```

There are manuals and other tutorials available for the commands and what can be done from the console. We can create virtual machines from this console as well as manage them through the xsconsole. This console is also accessible from an SSH connection. Make sure that the SSH access is well protected.

The other management tab is **Performance**, which displays the graphical stats of XenServer, such as CPU, memory, and network-utilization stats.

XenServer supports Microsoft Active Directory as a directory and authentication service. In order to add new users, assign them properties to restrict the actions they can perform. By default, we have a root user that is authenticated locally. So to configure the active directory, we can look at the options available in the **Users** tab.

Resource pools

So far we only have one XenServer host connected to XenCenter, and we haven't added any virtual machines. So this is a good time to look at adding another XenServer host to XenCenter and take a look at the resource pool. One of the basic conditions for adding servers in a pool is that there should be no running virtual machines or suspended virtual machines in the host that is being added into the pool. Also, we don't have any shared storage configured for our host.

We can combine multiple XenServer hosts into a combined big entity on which virtual machines can be created, and using XenMotion a live virtual machine can be migrated to another host in the pool with almost zero downtime. If one of the hosts in the pool fails, its VMs can be restarted on another host in the same pool.

If we're running XenServer Advanced Edition or higher, the VMs from a failed host in a pool are moved to another host of the same pool automatically. The High Availability feature needs to be enabled for this. We will learn more about the HA feature in *Chapter 8, Advanced Concepts*.

A pool always has one Pool Master, and the others are slaves. The Pool Master server maintains communication with the slaves and is the only point of contact for all the servers in its pool. If a Pool Master fails, we can't connect to the pool until the master is online or until we select a new XenServer host as the new Pool Master. The virtual machines will remain active on the other hosts.

In XenServer 6.1, there are two types of resource pools: **heterogeneous** and **homogeneous**.

When the hosts added in a pool have the same CPU (vendor, model, and features) and the version of the XenServer host is the same (the same patches are applied to hosts), such a pool is called homogeneous.

XenCenter won't let us create a pool if the CPUs of the hosts differ. If we are using the Advanced edition or above, we can leverage its heterogeneous pools feature, which enables us to pool servers with different CPU features; the vendor should be the same though.

A heterogeneous pool requires Intel's **FlexMigration** and AMD's **Extended Migration** technology. These technologies mask the CPU features and we can configure them to appear to be providing a different make, model, or functionality. Again, the XenServer running on such hosts should have the same processor and the same patch level.

Summary

We dealt with activating XenServer using a free or a non-free license and a general overview of the different tabs on XenCenter to manage and monitor the XenServer(s).

We discussed the different feature sets of different versions of XenServer. A brief explanation and usage scenarios of most of these features will be discussed in the upcoming chapters.

4

XenServer Storage Repositories

Storage plays a vital role in XenServer setups. We need big storages with properties that meet the needs to create a reliable service, and this can't be neglected. We will look at the following topics:

- Different types of storage
- Benefits of different storage types
- Creating different storages for XenServer

This will help us save our virtual machines' hard disks and other files, which we might need from time to time in our XenServer environment.

Storage technologies and protocols

In XenServer, the virtual machines behave like real machines with locally attached hard disks. In reality, these hard disks are allocated a chunk of real disk space of XenServer. The hypervisor and the control domain (`dom0`) have made this chunk of real disk space appear as a virtual hard disk for the VMs.

In this chapter we will be looking at the various methods and server space technologies that are supported by Citrix XenServer. Starting with storage technologies, there are different ways to attach and use storage devices, but typically there are three main architectures.

- Direct attached storage
- Network attached storage
- Storage area network

Direct attached storage

As the name suggests, these are the storage devices that are directly connected to the servers using the host bus. These are enclosures containing a number of disks within. **Direct attached storage (DAS)** systems offer fault-tolerance features and are able to "hot swap" failed disks and rebuild disks from parity on other disks.

Serially attached SCSI (SAS) and SATA are commonly used for directly connected storage devices.

Network attached storage

NAS are typically devices or computers that have large disk spaces; these are shared through a network to be commonly used to share files. The clients accessing this shared storage device are tablets, laptops, desktops, or servers, which only need to understand network-file-based sharing protocols such as NFS and CIFS/SMB. Initially, this can be thought of as a DAS system with its storage repository accessible over the network, but a specialized NAS server is far better in terms of efficiency and performance than an ordinary system with access shared on the network.

Storage area network

A **storage area network (SAN)** is a specially designed network architecture for attaching remote storage devices to servers in such a way that the devices appear as local disks attached to the operating systems. Any common device connected to the network can't simply access SAN. Only after the filesystems are built on the SAN, can they be utilized as NAS or file-sharing servers, based on whichever service is configured. Special equipment such as Fibre switches and directors are needed in order to connect more devices to the SAN network.

Storage protocols

A few popular storage protocols need to be briefly reviewed before we proceed to create storage repositories.

Network file system (NFS)

The NFS protocol is a simple protocol that only allows access to files over an Ethernet network. NFS is most frequently used within Linux or Unix environments, but is also available for other platforms such as Windows (Windows Services for Unix/NFS Client or Server) or Mac OS.

Initially, NFS was based on UDP for performance reasons. Starting with Version 3, NFS added support for networks based on TCP/IP.

NFS consists of at least two main parts: a server and one or more clients. The client remotely accesses the data that is stored on the server machine. In order for this to function properly, a few processes have to be configured and run.

NFS means we access a file share such as `\\MyServerAddress\MyFolder` and put files on it. In Windows, this is a mapped network drive. We access folders and files there but we don't see the network-mapped drive in Computer Manager as a local drive letter. We don't get exclusive access to NFS drives.

Internet Small Computer System Interface (iSCSI)

Internet Small Computer System Interface (iSCSI) is a SCSI mechanism, but over the Ethernet network. Its TCP/IP nature allows for a server/client mechanism in which a client who requires access to data is termed as **initiator** and the server that provides the data is termed **target**. The initiators can both be software-based clients or hardware devices. For authentication, iSCSI initiators and targets prove their identity to each other using the **Challenge-Handshake Authentication Protocol (CHAP)**, which includes a mechanism to prevent clear text passwords from appearing on the wire.

Fibre Channel (FC)

FC is a transport protocol that transports SCSI commands over Fibre Channel networks. Despite its name, Fibre Channel signaling can run on both twisted-pair copper wire and fiber-optic cables. To access the Fibre Channel network, host-based adapters need to be installed and configured. These HBAs are NIC-like cards that are installed on the server and bear a unique **World Wide Name (WWN)** or IEEE assigned **Organizationally Unique Identifier (OUI)**, similar to an Ethernet MAC address.

Fibre Channel can be used in three different topologies:

- **Arbitrated loop:** Similar to a token ring network, all devices are connected in a loop.
- **Point-to-Point:** Two devices are connected directly to each other.
- **Switched fabric:** Just like any Ethernet network, a switch is where all FC devices connect and from where they communicate to each other. This topology can be used to extend the Fibre network and hence the resulting bigger network is called **fabric**.

Storage objects in XenServer

The preceding brief introduction of storage techniques and protocols is important in order to understand the capabilities that Citrix XenServer provides us to facilitate a variety of storage containers within itself.

Concepts about the following should be briefly reviewed:

- **Storage repository (SR)**
- **Virtual Disk Images (VDIs)**
- **Physical Block Devices (PBDs)**
- **Virtual Block Devices (VBDs)**

Storage repository (SR)

This is the basic container in which XenServer holds the virtual machines' hard disks, known as VDIs. Storage repositories can be shared between multiple XenServers if storage technology allows it; for example, commonly accessible SAN space.

In order to create a storage repository, XenServer needs any one of the following, at least:

- Local storage on the host
- A LUN on an iSCSI SAN
- A LUN on a Fibre Channel SAN
- Storage through a NetApp
- The Dell EqualLogic storage appliance
- Network File Systems
- Any other externally connected devices, such as USB Flash Disks.

XenServer can have multiple SRs and each SR of a different type simultaneously.

Virtual Disk Images (VDIs)

VDIs are the fundamental unit of virtualized storage in XenServer, which are used as hard disks by virtual machines. VDIs are persistent, on-disk objects that are interfaced with a storage repository server using VBDs. VDIs have certain properties that are useful functions for better handling.

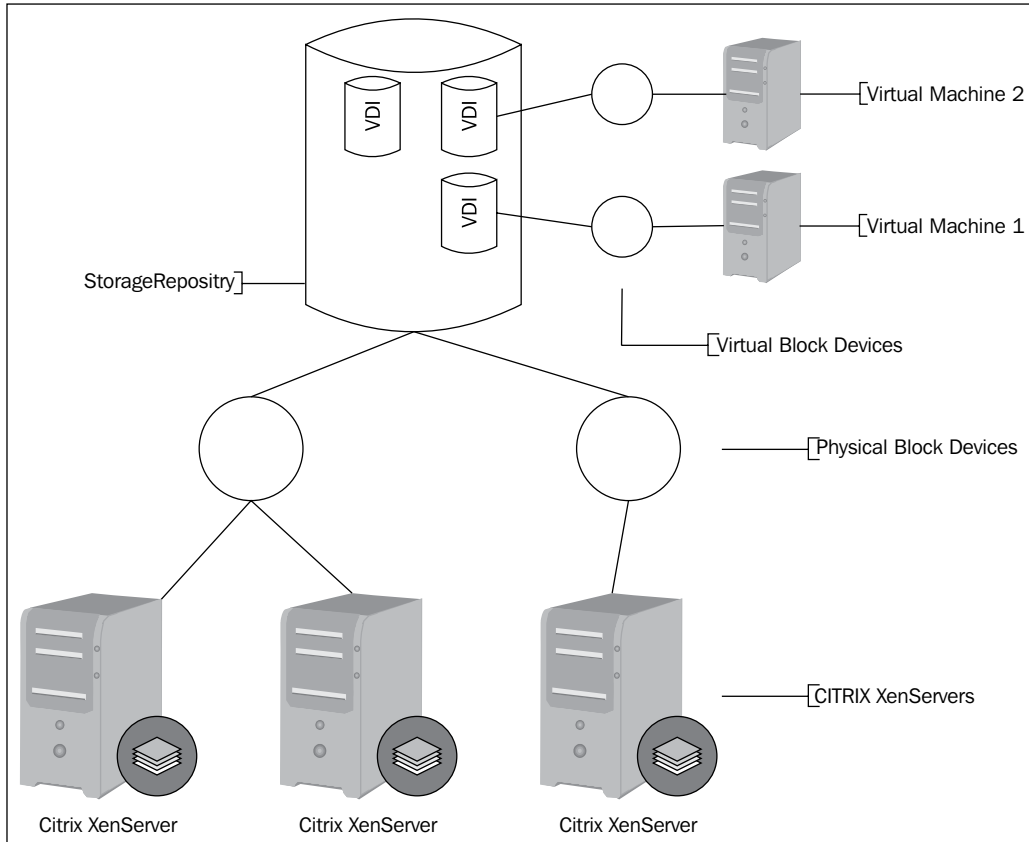
VDIs can be of different formats, such as **VHD (Virtual Hard Disk)**, **LVM (Logical Volume Manager)**, or a supported storage appliance. Each of these formats have their own specific advantages; for example, supported storage appliances give support to sparse allocation/thin provisioning, VDI resizing, and fast cloning. The VDI is also shareable.

Physical Block Devices (PBDs)

The interface between a physical server and an attached storage repository is a **PBD**. PBDs are connector objects mapping a storage repository to a XenServer host. They save the storage-device-specific configuration fields that are used to connect to and interact with a given storage target, which include the IP addresses, ports, username, password, and so on.

Virtual Block Devices (VBDs)

Like PBDs, **Virtual Block Devices (VBDs)** are connector objects that allow mappings between VDIs and virtual machines. In addition to plugging a VDI into a VM, VBDs allow fine-tuning of **QoS (Quality of Service)**, statistics, and the booting parameters of a VDI.



Citrix storage objects

Storage attributes

Citrix defines a few attributes for its VDIs. These properties allow administrators to manage the virtual disks in a very efficient manner. Sometimes, they don't even need to do anything and XenServer gives the flexibility by default. Let's briefly look at these attributes.

Multiple XenServers in a pool can "share" a storage based on a SAN or NFS. This is essential for the virtual machine's performance and migration. So, for any XenServer host having only DAS devices, this property is not available.

Another attribute is **sparse allocation**. The VDI file expands as the virtual machine writes data to it (the VM is writing to what it thinks is a local drive). The VM VDI files take up only as much space as is required. So, even though a total of 100 GB might be assigned to a VM, the actual used disk space might only be as much as the programs and OS installed on it have taken.

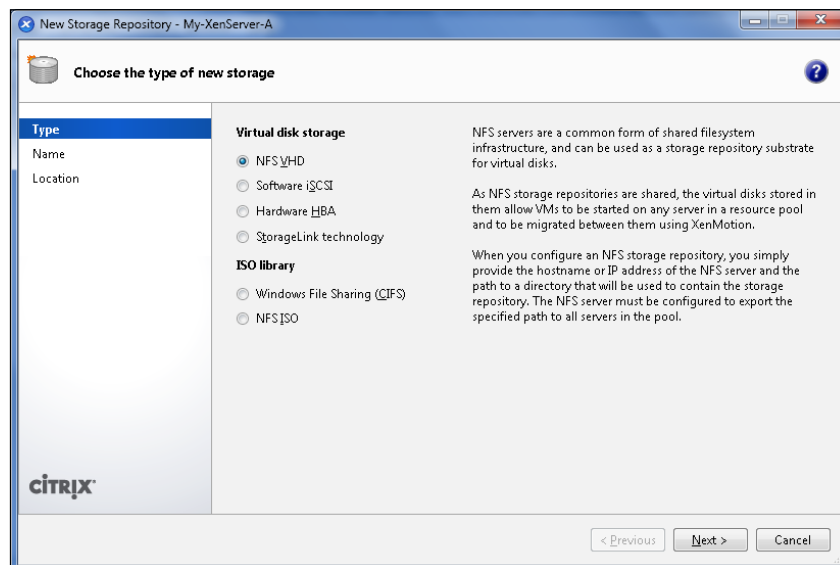
The **resizable** attribute is used on a detached VDI to increase the size of that VDI.

Fast cloning is the last attribute in which a new virtual disk image is created for a virtual machine, but it is in essence a pointer to an existing VDI. Any blocks that are read from the new VDI will actually come from the old VDI, but any changes to blocks or newly added blocks are provided in the new VDI. This means a VM can be cloned almost instantaneously, because the space is allocated on an as-needed basis.

Creating a storage repository

Let's proceed to creating different types of storage repositories. To start with, use the **New Storage** button on the top menu of XenCenter.

The **New Storage Repository** page provides a menu to select the type of storage that we are going to create; these include iSCSI and NFS VHD.



Creating an NFS VHD storage repository

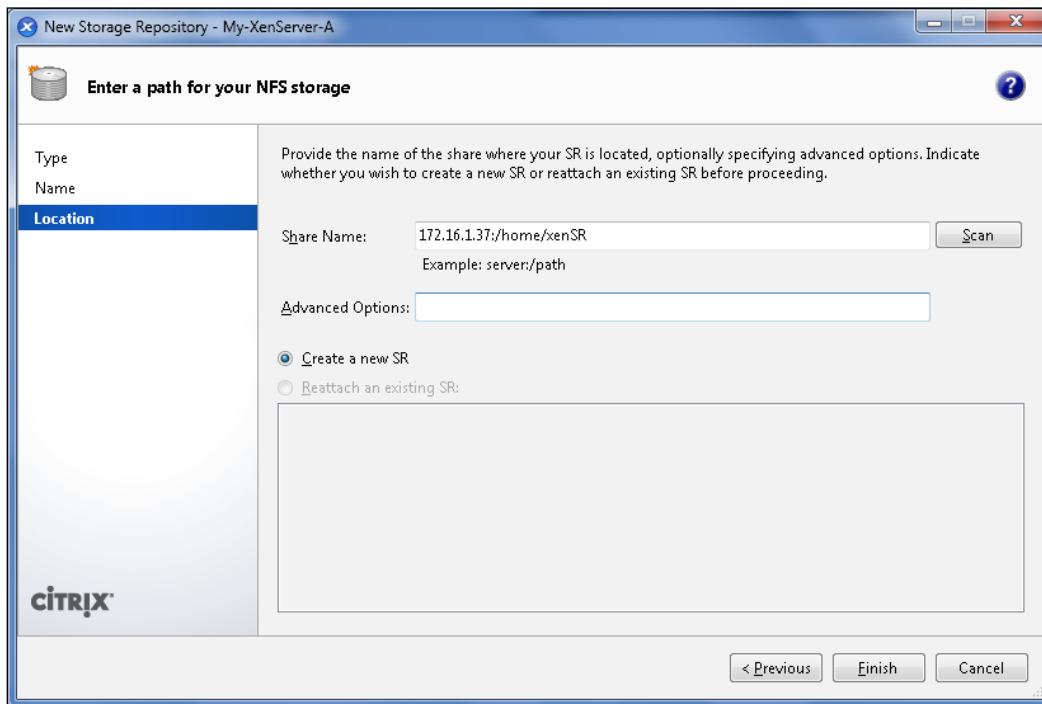
The NFS VHD type stores disks as VHD files on a remote NFS. XenServer can use a network file sharing server to save VHD-format VDIs to an NFS server that uses NFS-V3 over TCP/IP.

VDIs are stored in the Microsoft VHD format only. Since the NFS server is shared and we can share the NFS storage repository between XenServer hosts, we can migrate or start the VMs in a resource pool. VDIs on NFS Storage repo have the sparse attribute.

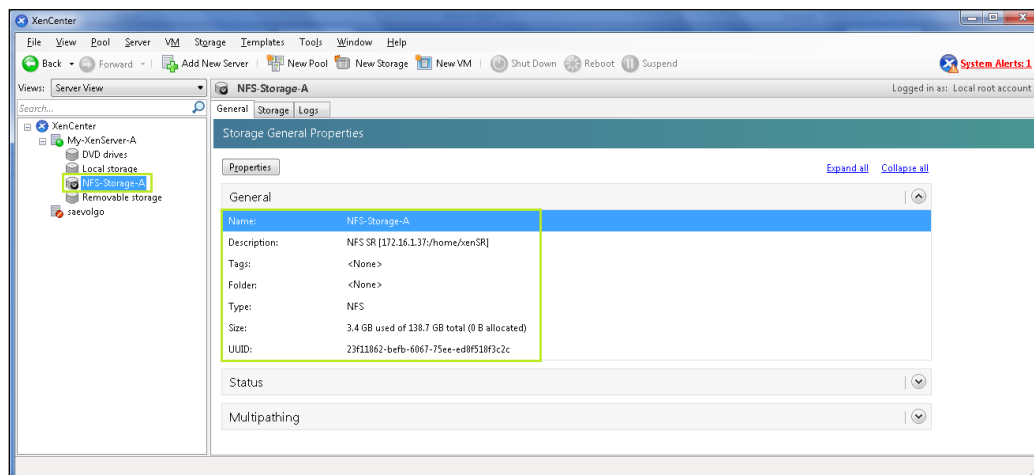
Select NFS VHD and click on **Next**. On the **Name** page, enter the name of the new SR.

By default, a description of the SR will be generated automatically by the wizard, including a summary of the configuration options you select as you progress through the wizard.

Click on **Next** to continue. Finally, on the **Location** page, insert the NFS shared space path and any NFS advanced options.



Click on **Finish** and a new storage repository will be created.



So, our very first NFS VHD storage space is available for us to use.

Creating a software iSCSI Storage Repository

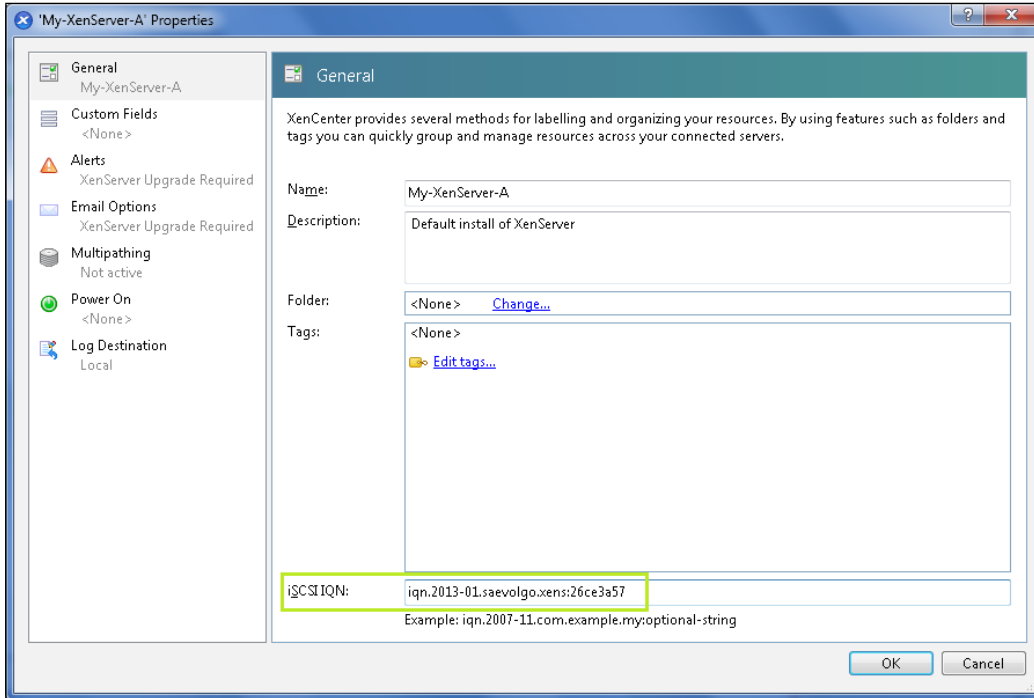
XenServer supports storage repositories using the open-iSCSI software or by using a supported iSCSI host bus adapter on iSCSI LUNs.

The software iSCSI initiator uses the technology based on LVM and has the same advantages as LVM VDIs. Virtual machine migration using XenMotion is also supported on shared iSCSI storage repositories having a software-based host initiator. Remember that all iSCSI initiators and targets must have a unique name to ensure that they can be uniquely identified on the network. An initiator has an iSCSI initiator address and a target has an iSCSI target address. Collectively, these are called **iSCSI Qualified Names (IQNs)**.

XenServer hosts support a single iSCSI initiator that is automatically created and configured with a random IQN during host installation. The single initiator can be used to connect to multiple iSCSI targets concurrently.

So, in order to create an iSCSI storage repository, we need to first configure the default IQN by changing the properties of our host XenServer via XenCenter.

Click on the **Properties** button appearing on top of this tab and in the window that appears next, change the **iSCSI IQN** field at the bottom.



Now perform the following steps:

1. Open the **New Storage Repository** wizard and click on the **New Storage** button on the toolbar.
2. Select **Software iSCSI** as the physical storage type and then click on **Next**.
3. On the **Name** page, enter the name of the new SR and click on **Next** to continue.
4. On the **Location** page, specify the iSCSI target details, including the IP address or DNS name of the iSCSI target.

If the iSCSI target is configured to use CHAP authentication, select the **Use CHAP** checkbox and fill in the following details:

- **CHAP User:** The CHAP authentication username credential that should be applied when connecting to the target

- **CHAP Password:** The CHAP authentication password credential that should be applied when connecting to the target
5. Click on **Discover IQNs** to discover the iSCSI target IQN and choose the IQN for your storage appliance.
 6. Click on **Discover LUNs** to specify the LUN on which the storage repository will be created.
 7. Click on **Finish** to complete the new SR configuration and close the wizard.

New Storage Repository - My-XenServer-A

Enter a path for your iSCSI storage

Type	Name
Location	

Provide a target host for your iSCSI storage, indicating your target IQN and your target LUN before proceeding.

Target Host: 172.16.1.40 : 3260

Use CHAP

User: testuser

Password:

Target IQN: [] Discover IQNs

Target LUN: [] Discover LUNs

CITRIX

< Previous Finish Cancel

Creating a hardware HBA storage repository

Hardware HBA storage repositories connect to Fibre Channel, **Fibre Channel over Ethernet (FCoE)**, or shared SAS LUNs via a host bus adapter.

Before creating an HBA SR, you need to carry out the configuration required to expose the LUN to the XenServer pool.

Refer to your HBA manufacturer for details on configuring the host adapter.

When you start the **New Storage Repository** wizard, it will automatically probe for available LUNs and display a list of all the LUNs found.

1. Open the **New Storage Repository** wizard and click on the **New Storage** button on the toolbar.
2. Select **Hardware HBA** as the physical storage type and then click on **Next**.
3. On the **Name** page, enter the name of the new SR. Click on **Next** to continue.
4. The wizard scans for available LUNs and then displays a page listing all the LUNs found. Select a LUN from the list.
5. Click on **Finish** to complete the new SR configuration and close the wizard.

Creating a StorageLink repository

A StorageLink storage repository can provide direct access to high-performance storage, allowing the VMs in your XenServer environment to benefit from array-side enterprise storage services such as replication, deduplication, thin provisioning, snapshots and cloning, data protection, and performance optimization.

To create a StorageLink repository, follow the given procedure:

1. In the **Resources** pane, select a server or pool and then right-click and click on **New Storage** on the shortcut menu.
2. Select **StorageLink Technology** as the physical storage type and then click on **Next**.
3. On the **Name** page, enter the name of the new SR. Click on **Next** to continue.
4. On the **Storage Adapter** page, select one of the available storage system adapters and then click on **Next** to continue.
5. On the **Storage System** page, select the storage system that will host the storage for this SR.

In the **Array** target box, enter the hostname or IP address of the array management console.

Under **Credentials**, enter the username and password to use for connecting to the array management console.

Click on **Discover** to scan the target array for storage systems. After discovery has completed, select a storage system from the **Storage System** list and then click on **Next** to continue.

In the **Settings** page, define the SR settings. The set of available settings for each new SR depends on the storage system vendor/model and the configuration of the storage pool you've selected.

Click on **Finish** to add this new storage repository.

Here are some fields that need to be set:

- **Storage pool:** Identify the storage pool within the specified storage system to be used for allocating storage. On some types of storage systems, you can use the **Show all** checkbox to list all storage pools here.
- **RAID types:** Select the level of RAID to use for the SR, as supported by the storage array.
- **Provisioning type:** Set the provisioning type (**Default**, **Thick**, or **Thin**).
- **Provisioning options:** Set the provisioning options (for example, deduplication).
- **Protocol:** Set the storage protocol used to connect the managed server to the storage (**iSCSI**, **Fibre Channel**, or **Auto** to have the system choose).
- **Use CHAP:** If the server is configured to use CHAP authentication, select this checkbox and fill in the username and password to be used.

Creating an ISO library

An ISO library means a shared storage area where all the operating system's ISO images are placed and can be accessed to start the installation from there. So, essentially, we don't need to put any OS DVD in the XenServer or place any ISO in the XenServer; we can rather just create an ISO library and use the ISOs from there.

Again, we have two options to connect to our ISO library:

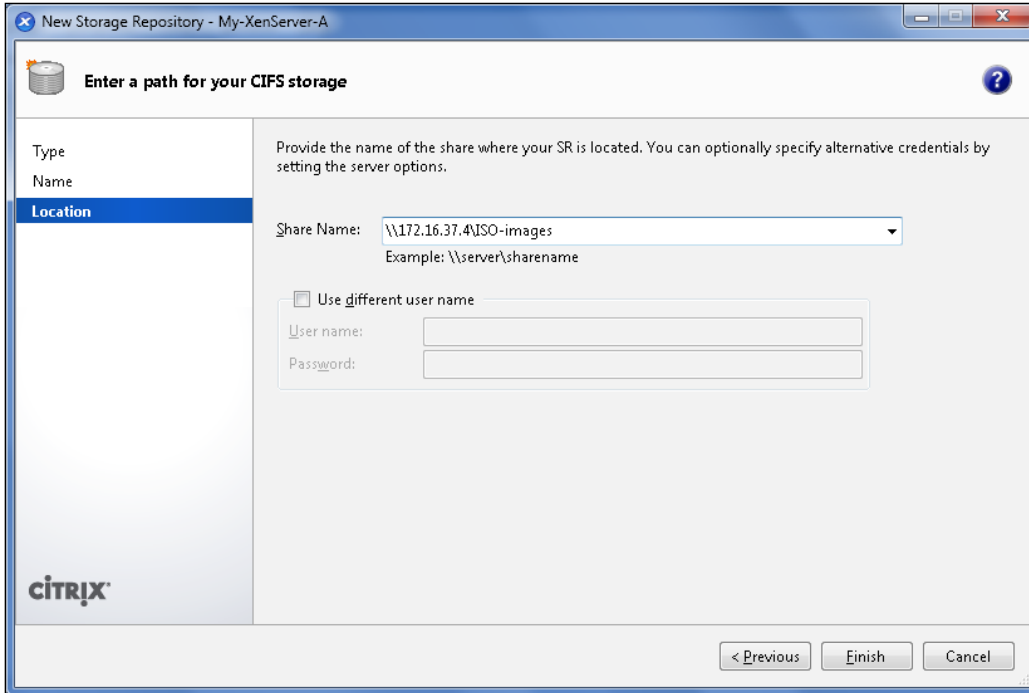
- Windows File Sharing (CIFS)
- NFS ISO

CIFS is a standard Windows-based shared-folder technique while **NFS** is a shared-mount directory similar to the storage repository we had for the virtual machine's VDIs.

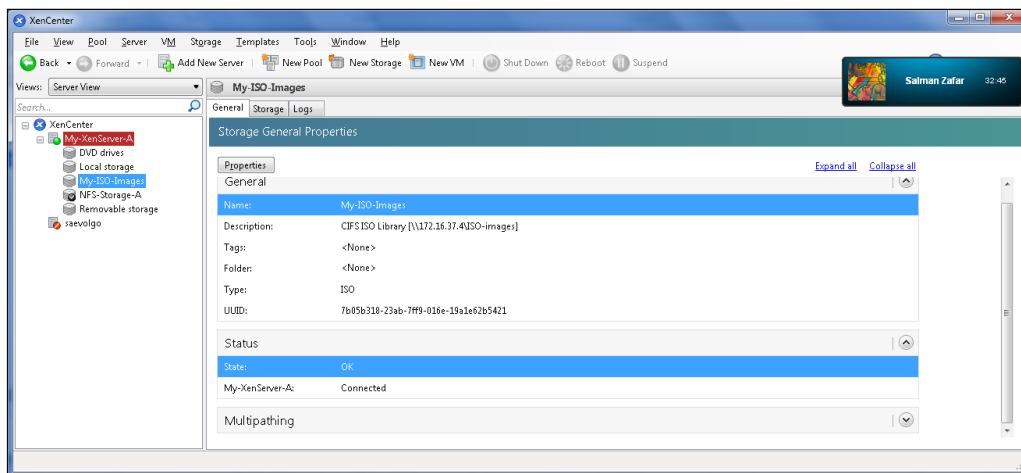
Let's start with the Windows file-sharing method first. Click on **Next** and type in the name of the ISO library; then click on **Next** again.

XenServer Storage Repositories

Provide the path where ISO images will be located, and if a Windows username/ password is required, enter those here as well to access that folder.



Click on **Finish** and the ISO library will be added to XenServer.

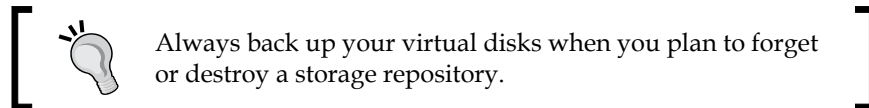


This ISO library will be used in the next chapter to load up the operating system's bootable ISOs and create new virtual machines.

The same steps can be performed for creating an NFS-based ISO library. It involves three steps: selecting the NFS type, naming the NFS library, and giving the location of the NFS mount.

Managing the storage repositories

Using XenCenter or XenConsole, a storage repository can be removed temporarily or permanently.



Detaching can be used when we need to temporarily take a storage repository offline; for example, for maintenance, when the virtual disks become inaccessible.

We can forget a storage repository; the contents of the virtual disks on the SR are preserved but the information used to connect virtual machines to the virtual disks they contain is permanently deleted. The SR is removed from the **Resources** pane. A **Forget** operation cannot be undone.

If we destroy an SR, both the information used to connect virtual machines to the virtual disks on the SR and the underlying virtual disks themselves are permanently destroyed. The SR is removed from the **Resources** pane. A **Destroy** operation cannot be undone.

We cannot remove an SR if it holds the virtual disks of a currently running virtual machine.

Summary

We have now created storage repositories and ISO libraries for our XenServer. In the next chapter we will be creating our first virtual machine using templates and also look at cloning the virtual machines.

5

Playing with VMs

In this chapter we will be learning about the following:

- Making templates
- Using templates to create virtual machines
- Making templates from virtual machines

It is really interesting to understand that we can create snapshots of our running virtual machines and use those snapshots to create templates. These templates can then be used to replicate many new virtual machines.

Virtual machines using templates

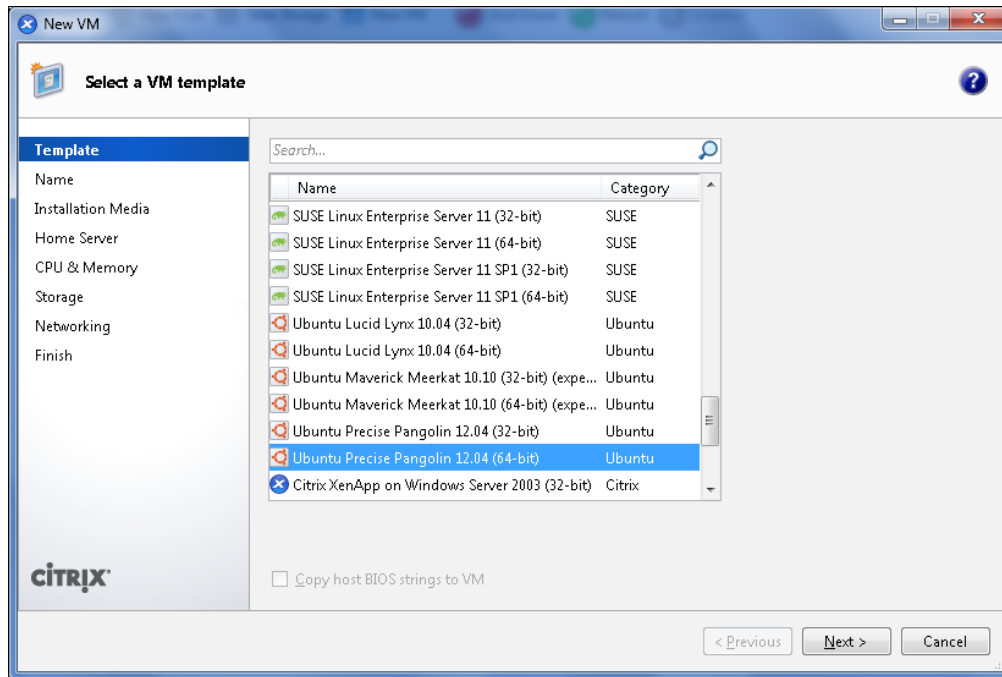
So we've already understood the core concepts of Citrix XenServer's working and storages. Now let's start making the virtual machines. A **Virtual Machine (VM)** is known as a guest machine on the host XenServer and uses the resources we allocate to it. These VMs can have Linux or Windows or any other operating system.

There are different methods to create a virtual machine:

- By using templates
- By custom installation from ISO images
- By importing a virtual machine
- By cloning an existing VM
- By converting a physical machine into a virtual machine

Creating a virtual machine from a template means that some predefined VM settings, such as disk space, CPU, and memory, get loaded for that particular operating system. Though these settings are loaded, we still need to provide a bootable media for installation.

In order to create a VM using templates, select the XenServer host and press the **New VM...** button.

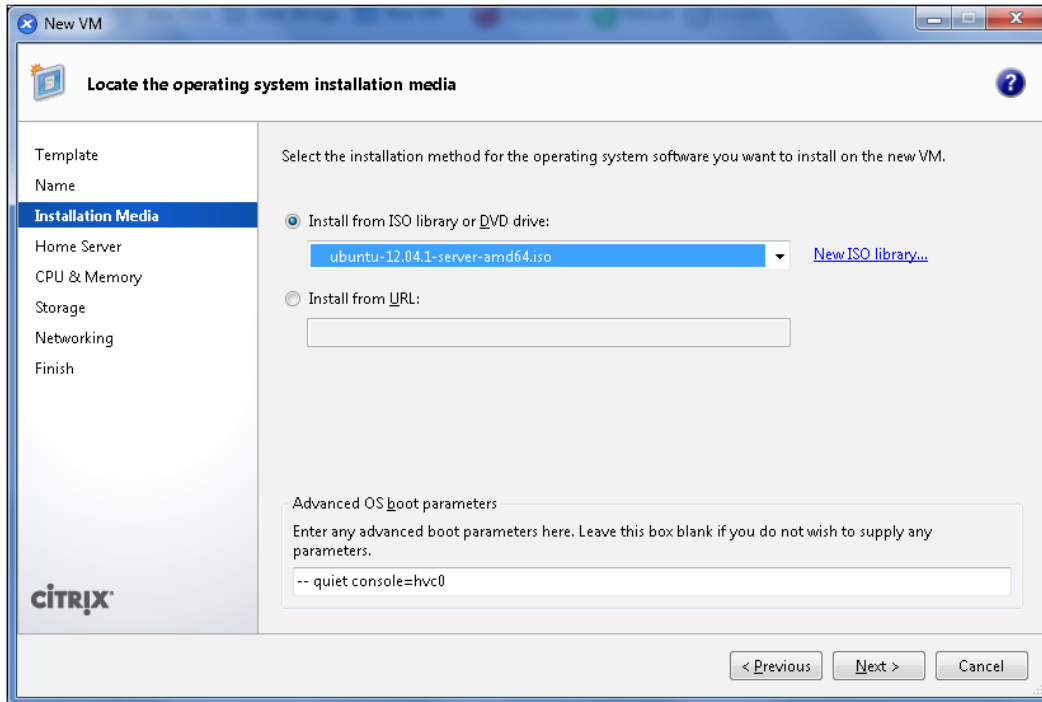


Selecting a template for a new virtual machine

Select the type of OS template we wish to load. If the OS we need to install isn't in the list, we can choose **Other Install Media** at the end.

Note that VMs created using the **Other Install Media** template will be created as HVM guests, which means that the Linux VMs created using this template will not be able to use the native high-performance drivers (PV drivers) included in modern kernels.

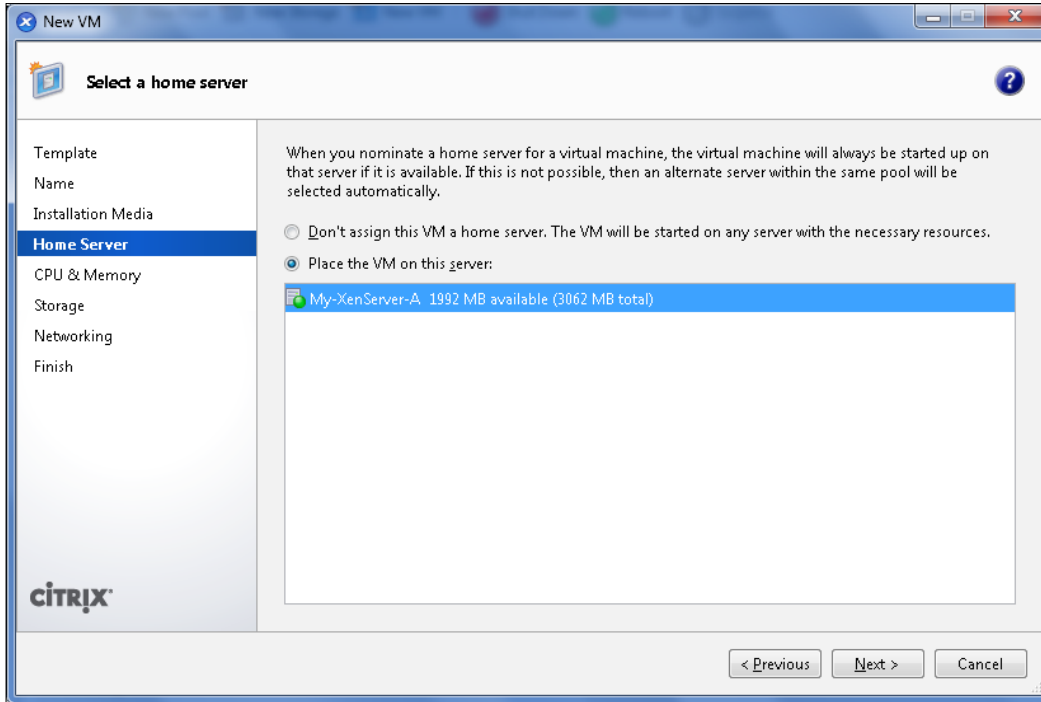
Click on **Next** and fill in the **Name** and **Description** fields. Always put in some meaningful description for VMs.



Selecting the operating-system-bootable ISO

Select the installation media by selecting the ISO from the ISO storage repository we added in the previous chapter. If we have the URL of the ISO from the local network or over the Internet, we can have the OS installed over the network too.

Click on **Next** and select the XenServer host where we need to place this VM.

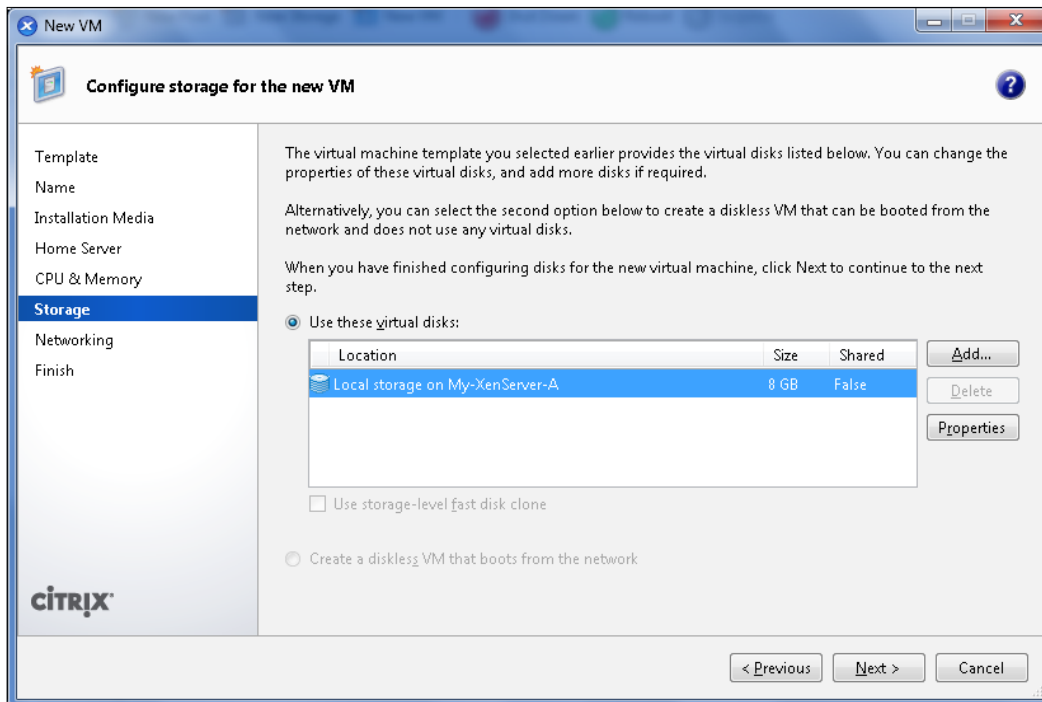


Selecting the host XenServer for a virtual machine

For multiple XenServers and lots of VMs, we can let XenCenter choose where this new VM can fit and be placed. Once the host server is selected, click on **Next** to assign the number of CPUs and memory for this VM. It is recommended not to assign more vCPUs than the physical CPUs on the XenServer.

Assigning memory to the VM needs to be done carefully as well, as that amount of memory will be consumed by this VM once it is started.

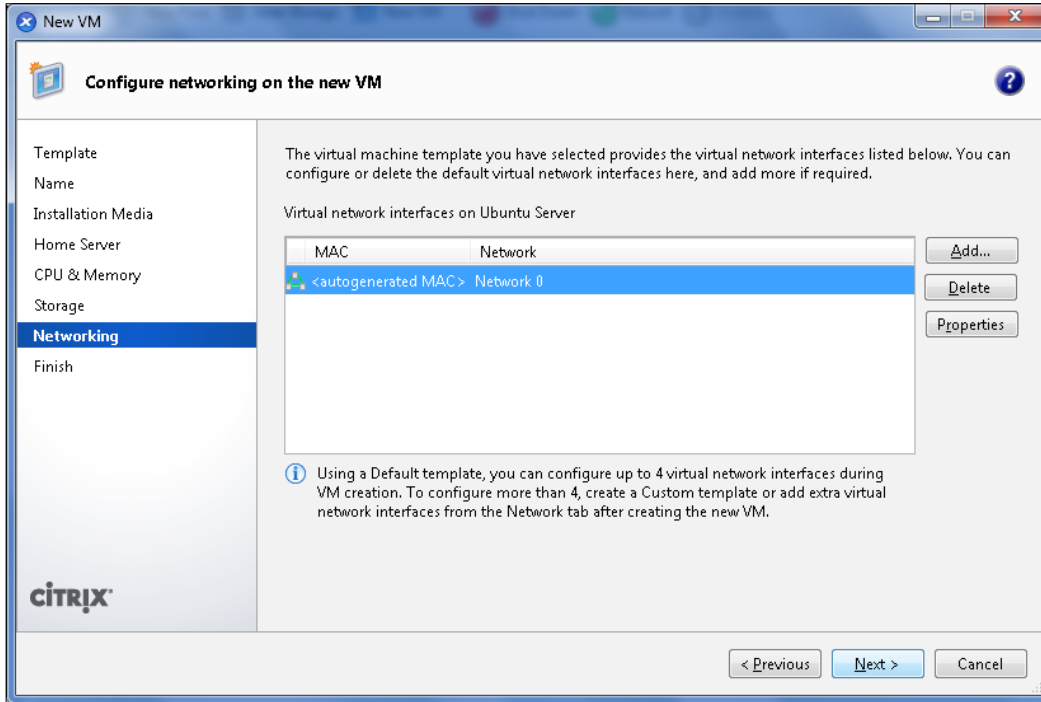
Once memory and the number of CPUs are assigned, click on **Next** to choose the storage where we want this virtual machine's hard disk (VDI) to be created. At this point, remember that the type of storage will display the available storage attributes for this VM's virtual disk.



Selecting storage space to create the vHDD

We can add more hard disks to our virtual machine at this point, or later on, which we'll see in further chapters. We can change the amount of disk space we want to assign to this VM by clicking on the **Properties** button.

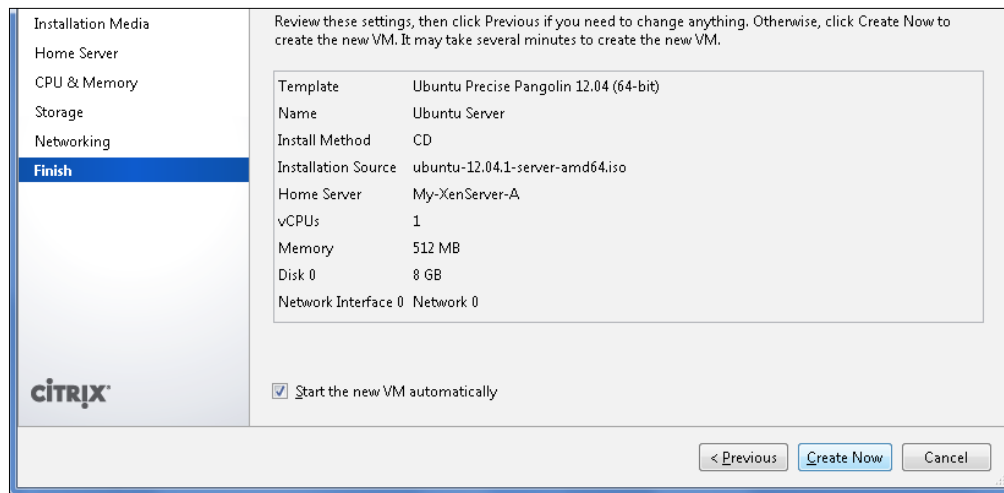
Clicking on **Next** will take us to the page where we can configure the network for this VM. Initially, we have only one default network: 0. All our VMs will get connected to this network.



Network interface and network connectivity

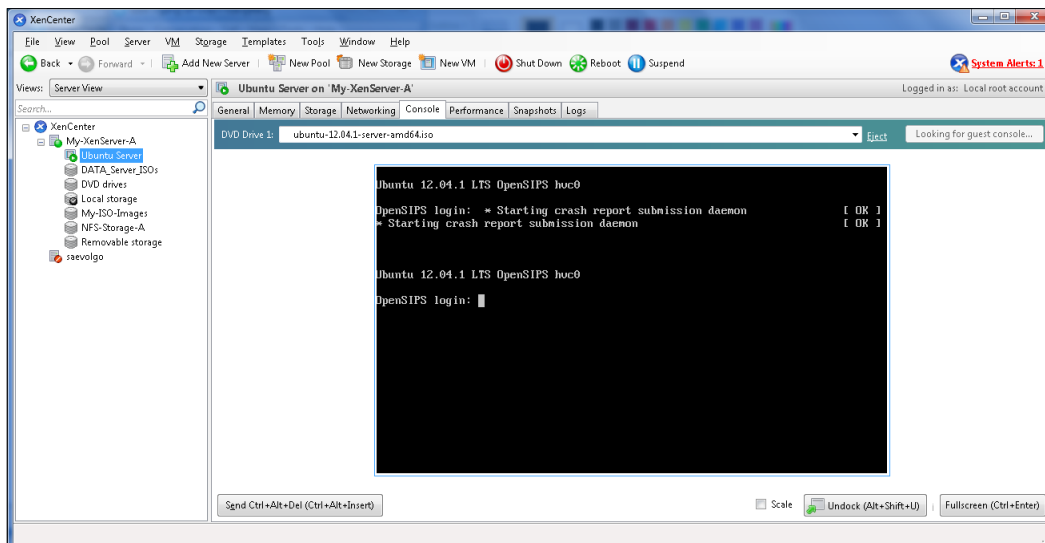
We can edit the MAC address and apply some bandwidth limit by clicking on the **Properties** button for the selected network interface. As mentioned in the previous screenshot, we can configure up to four virtual interfaces at the time of creation of the VM.

That's pretty much it; we're done with our very first VM configuration. Click on **Next**; in the **Finish** step, we'll see the summary of what we've configured. We can still go back and change the settings. Let's go ahead and click on the **Create Now** button.



Summary of the VM to be created

After a few seconds of processing while the hypervisor is instructed to create the virtual machine, we'll see the OS installation steps on the **Console** tab for this new VM because we'd previously selected to start this VM when completed. I'm skipping the OS-specific installation process here, and once that part is done our VM is ready for use.



The virtual machine console after installation

XenServer Tools

For the initial Windows installation, traditional device emulation is used by XenServer to present a standard IDE controller and network card to the virtual machine. Windows uses built-in drivers to complete the installation phase, but this is a slow method and reduces the performance as it is an emulation of controller drivers. To overcome this, Citrix provides high performance I/O services without the overhead of traditional device emulation and also provides high-speed transport between Windows and the XenServer product family software.

Running a VM without installing XenServer Tools is not a supported configuration, so Citrix recommends that we install it immediately after the operating system installation. For some operating systems, XenServer Tools includes a XenServer-specific kernel, which replaces the kernel provided by the vendor. Other operating systems, such as RHEL 5.x, require that we install a specific version of a vendor-provided kernel.

If you are working with a VM that does not have XenServer Tools installed, a **Tools not installed** message in blue hypertext will be visible on the **General** tab in the **Properties** pane. Click on that link and a pop up will ask if we want to install XenServer Tools.

Select the VM in the **Resources** pane, right-click, and then click on **Install XenServer Tools** on the shortcut menu.

Alternatively, on the VM menu, click on **Install XenServer Tools**.

Click on **Install XenServer Tools** on the message dialog to go to the VM's console.

As the root user, mount the image into the VM.

```
mount /dev/xvdd /mnt
```

Execute the installation script as the root user.

```
/mnt/Linux/install.sh
```

```
gohar@OpenSIPS:~$ sudo /mnt/Linux/install.sh
Detected `Ubuntu 12.04.1 LTS' (ubuntu version 12).

The following changes will be made to this Virtual Machine:
* update arp_notify sysctl.conf.
* packages to be installed/upgraded:
  - xe-guest-utilities_6.1.0-1033_amd64.deb

Continue? [y/n] y

Selecting previously unselected package xe-guest-utilities.
(Reading database ... 26125 files and directories currently installed.)
Unpacking xe-guest-utilities (from ../xe-guest-utilities_6.1.0-1033_amd64.deb)
...
Setting up xe-guest-utilities (6.1.0-1033) ...
Mounting xenfs on /proc/xen: OK
Detecting Linux distribution version: OK
Starting xe daemon: OK
Processing triggers for ureadahead ...
ureadahead will be reprofiled on next reboot

You should now reboot this Virtual Machine.
gohar@OpenSIPS:~$ █
```

Installing XS-Tools on a Linux VM

If the kernel has been upgraded, or the VM was upgraded from a previous version, reboot the VM now.

For Windows-based machines, an executable installer will pop up, and from there onwards it's a simple follow-the-wizard process. The installer will also load up if we load the `xs-tools.iso` ISO image into the Windows VM, and Autoplay will launch the installer menu.

In the XenServer environment, the CD-ROM drivers and ISO attached to Linux VMs appear as `/dev/xvdd`, unlike the usual Linux `/dev/cdrom` mount point. Since they are emulated devices, when we eject the CD from XenCenter, it hot-unplugs the device from the VM and the device `/dev/xvdd` disappears. In a Windows VM, this behavior is different, and we can see an empty CD-ROM icon rather than no device for CD/DVD.

XenServer Tools must be installed for each virtual machine (Windows and Linux) in order to be able to use the `xe` CLI or XenCenter; the VM's performance will be significantly lower unless the tools are installed. Without the tools installed, you will not be able to do any the following:

- Cleanly shut down, reboot, or suspend a VM
- View VM performance data in XenCenter
- Migrate to a running VM (via XenMotion)
- Create snapshots or snapshots with memory (checkpoints), or revert to snapshots
- Adjust the number of vCPUs on a running Linux VM (Windows VMs require a reboot for this to take effect)

Creating new templates

Creating new templates is often required in an environment where VMs of particular configurations are needed regularly. We can create new custom templates in several different ways in XenCenter, as follows:

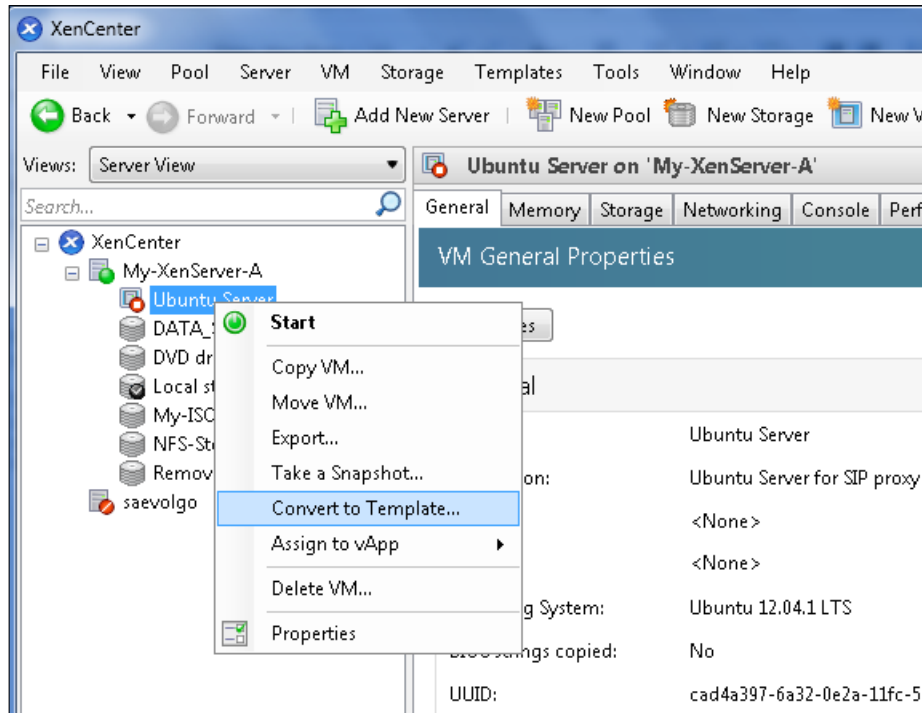
- By copying an existing template
- By converting an existing VM into a new template
- By saving a copy of a VM snapshot as a new template
- By importing a template that has previously been exported from an existing template or VM snapshot as an XVA file

Cloning or copying of existing VMs and templates will be discussed later; right now, let's look at converting an existing VM into a new template. We need to have the required VM in the shut-down state to convert it into a VM; we can't convert a running virtual machine into a template.

Converting an existing VM to a template

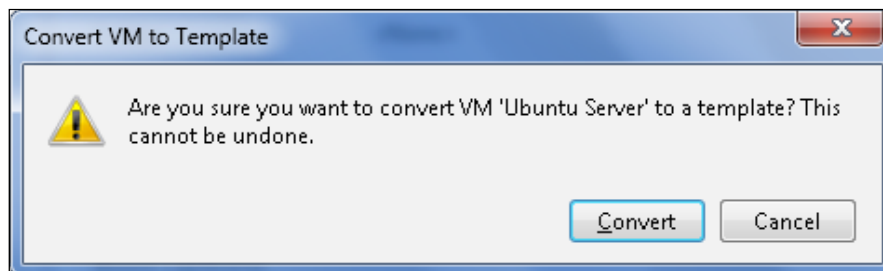
When we create a new template using this method, the VM's disks will be copied to the new template and the original VM will no longer exist.

Right-click on the VM that is shut down and click on **Convert to Template....**



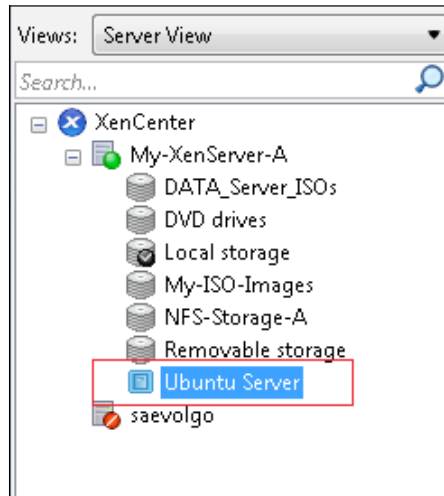
Converting a VM into a template

This will pop up a warning to think about what we're doing, as this change is not reversible. Our VM won't be the same VM anymore.



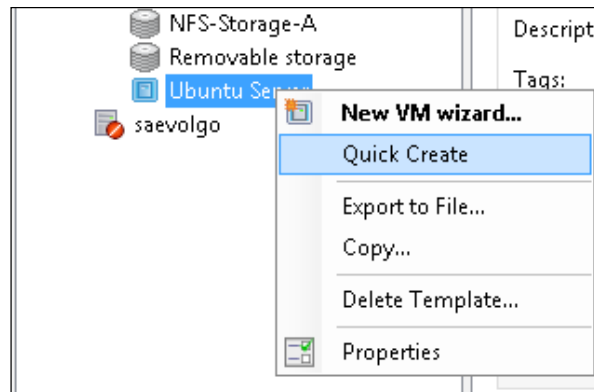
VM template conversion confirmation

Click on **Convert**; XenCenter will show the progress bar in the **Logs** tab and soon we'll see a new icon in our **Resources** panel. Now, creating a new VM will allow us to select this template from the template list.



VM converted to a template

If we want to save the new VM in our template, just right-click on the newly created template and select **Quick Create**, and in few seconds another server is ready.

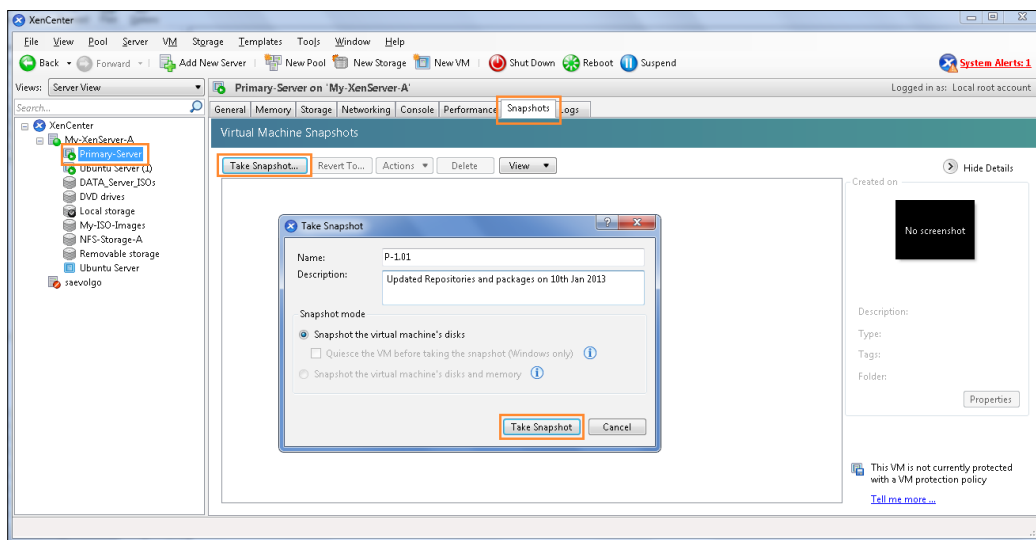


Quick VM creation from a template

Everything we've installed on this VM, from OS to any special packages, users, and so on, is saved in this template. Creating VMs from this template is just like cloning a VM from a saved snapshot. Imagine a Windows desktop machine with organizational-standard software installed in it with all the usual settings; since that template is used to create new VMs, all the troubles of Windows, drivers, updates, and software installation, will only be encountered once. Or imagine a Linux server with all the routine packages and libraries and users defined for once, and then recreating the servers with that template. A lot of time and bandwidth can be saved.

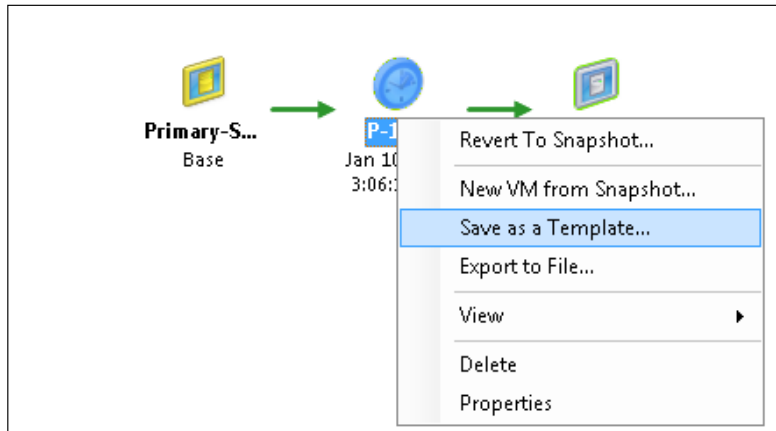
Making templates from snapshots

Another method to create a template can be using a snapshot of a VM. The benefit of this is that we can have updated templates with newer stuff; we've done this in the template-creating VM. An advantage of this technique is that our virtual machine is available to us for use.



Creating a snapshot for template creation

The following options will appear; right-click on the snapshot to convert it into a template.



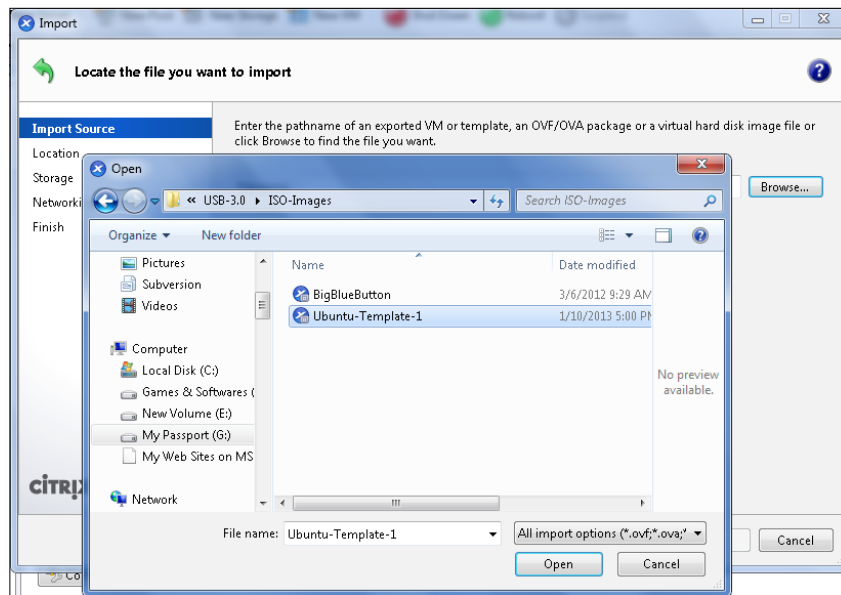
From a snapshot to a template

Once the new template has been successfully created, it appears as a custom template in the **Resources** pane and on the **Templates** page in the **New VM** wizard. Templates created this way will appear in the templates list only when creating new VMs.

Importing templates

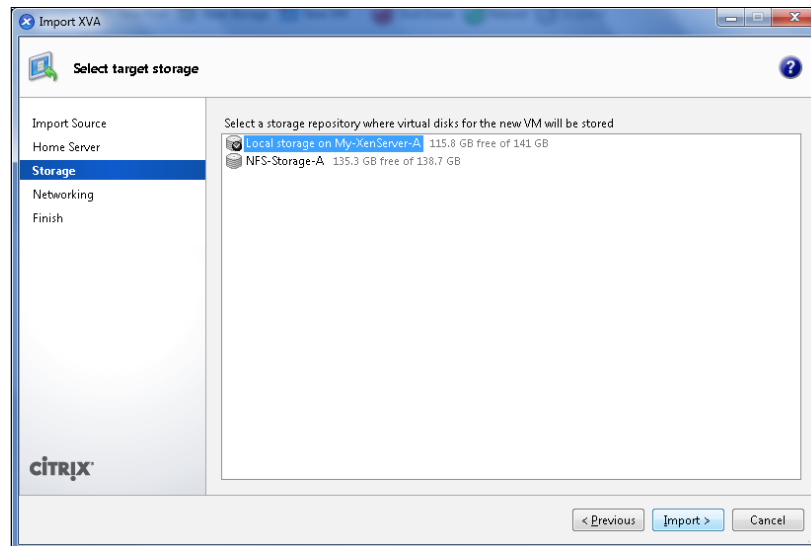
VM templates and snapshots that have been exported as XVA files can be imported using XenCenter's **Import** wizard. On the **File** menu, click on **Import**, select the XVA file containing the template on the first page of the wizard, and then follow the same steps.

Select **Import** from the **File** menu. A file browser will pop up where we need to locate the template we wish to import.



Importing an exported template

In the next step, we need to select which Xen host server we want to place this template onto. Here we are asked to follow almost the same steps we performed when creating a new VM. Select the storage space where we will place this template.



Selecting storage for placing the template

Click on **Import** > to make XenCenter prepare for the import of the XVA file; select the default **Network-0** option in the next step and then review the summary of the import and click on **Finish**.

The import progress is displayed on the status bar at the bottom of the XenCenter window and also on the **Logs** tab. Another template will be visible in the server resource area once the import process is complete. The import process may take some time, depending on the size of the imported VM's virtual disks, the available network bandwidth, and the disk-interface speed of the XenCenter host.

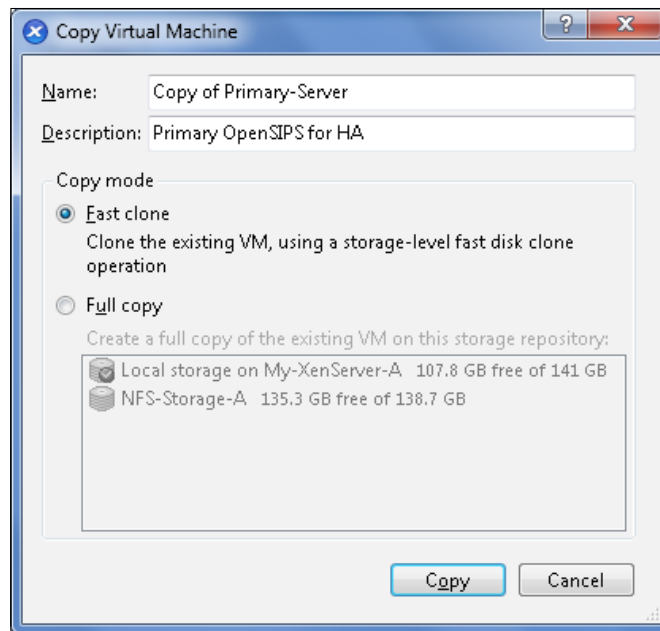
Cloning VMs and templates

Cloning a virtual machine is useful when we need to set up many identical machines in our environment. We can create new VMs and templates by copying (cloning) an existing VM or template. XenServer has two mechanisms for copying VMs and templates: full clone and fast clone.

Normally, cloning a virtual machine or a computer will duplicate the attributes unique to that machine, such as the IP address, SID, or MAC address. To avoid this, XenServer automatically changes some virtual hardware parameters when we clone a Linux VM from XenCenter. If the interfaces are statically configured, we might need to modify their network configurations, but dynamic network configurations will work without any changes. The process of unbinding the machine-specific attributes is termed as "generalizing" the virtual machine.

A Linux virtual machine needs to be shut down for us to be able to see the **Copy VM...** option when we right-click on that virtual machine. Windows-based VMs need a special procedure to generalize and this will be discussed towards the end of this chapter.

Clicking on **Copy VM...** will ask us to choose between two options to specify how to clone this virtual machine.



The options for VM cloning are as follows:

- **Full copy** makes a complete copy of the VM's disks
- **Fast clone** (copy-on-write) writes only the modified blocks to the disk

The fast clone option uses hardware-level cloning features for copying the disks from the existing VM to the new VM. This mode is only supported for file-backed VMs. Copy-on-write is designed to save disk space and allow fast clones, but can slow down normal disk performance slightly.

We can only copy a VM directly within the same resource pool. To copy a VM to a server in a different pool, we will need to `export` the VM and then `import` it to the destination server.

A VM converted to a template can also be copied/cloned with the same options.



If a template is cloned into a VM and the clone is converted back into a template, disk performance can linearly decrease depending on the number of times this has happened. In this event, the `vm-copy` CLI command can be used to perform a full copy of the disks and restore the expected levels of disk performance.

Preparing and cloning Windows VMs


Computers running the Windows operating system are uniquely identified by a security ID (SID). To "generalize" a Windows-based virtual machine, a Windows command utility should be run; `sysprep` is the utility that generalizes the Windows-based VMs. Not performing the necessary steps to clone a Windows system will lead to duplication of SIDs and other problems. For more information, refer to the Microsoft Knowledge Base article 314828, *The Microsoft policy for disk duplication of Windows installations*.

`sysprep` uniquely changes the SID. `sysprep` can be found in the `\support\tools\deploy.cab` file on the Windows installer disc.

We need to have at least one fully running Windows VM with all patches and XenServer Tools installed (with any other application installed that might be wanted in every VM made from this template).

To start cloning, perform the following steps:

1. Copy the contents of `\support\tools\deploy.cab` from the Windows product CD to a new `\sysprep` folder in the VM.
2. Run `sysprep`. This will shut down the VM when it is completed.

 Do not start this virtual machine; if you start it again, `sysprep` has to be executed again. This VM should be converted to a template from this shut-down state.

3. From XenCenter, convert this Windows machine to a template.
4. Clone the newly created template into new VMs, as required.
5. When the cloned VM starts, it will do the following:
 - Get a new SID and name
 - Run a mini-setup to prompt for configuration values, as necessary
 - Restart

Now this cloned virtual machine is ready for use.

Summary

We were able to create our first virtual machine. We also created a template from our virtual machine and imported templates. We now also know the two different ways of cloning a virtual machine.

The next chapter talks about managing our virtual machines and grouping virtual machines together in vApps.

6

Managing Virtual Machines

We've created a couple of virtual machines and discussed how to create clones and use templates. In this chapter we are going to take a closer look at certain hardware configurations in virtual machines.

- CPU configuration
- RAM size allocation
- Disk space
- Network interface card settings

Managing storage, memory, and networking

Every system we use daily around us depends primarily on its processor capability and the amount of memory available to be utilized. Besides these, disk space and network devices play the important roles of storing and communicating with other devices. To have a virtual machine that performs better, we need better and more of these resources. So, let's see how these settings are configured in XenServer.

Managing VM CPU and memory

When calculating the memory that XenServer uses, there are two components to consider. The first is the memory that the XenServer virtualization engine, known as Xen hypervisor, uses. The second is the memory that the control domain uses. A Xen hypervisor uses a fixed amount of memory. This value is set to 128 MB. Dom0 uses a variable amount of memory based on the total physical memory on the server. We have discussed this in previous chapters.

In previous chapters, we've learned how to change the control domain virtual machine memory in order to be able to run a high number of VMs on the host. Now we will see how much memory is necessary on our XenServer host to run virtual machines with best performances.

To find out how much memory is needed for your XenServer host, you can use the following formula:

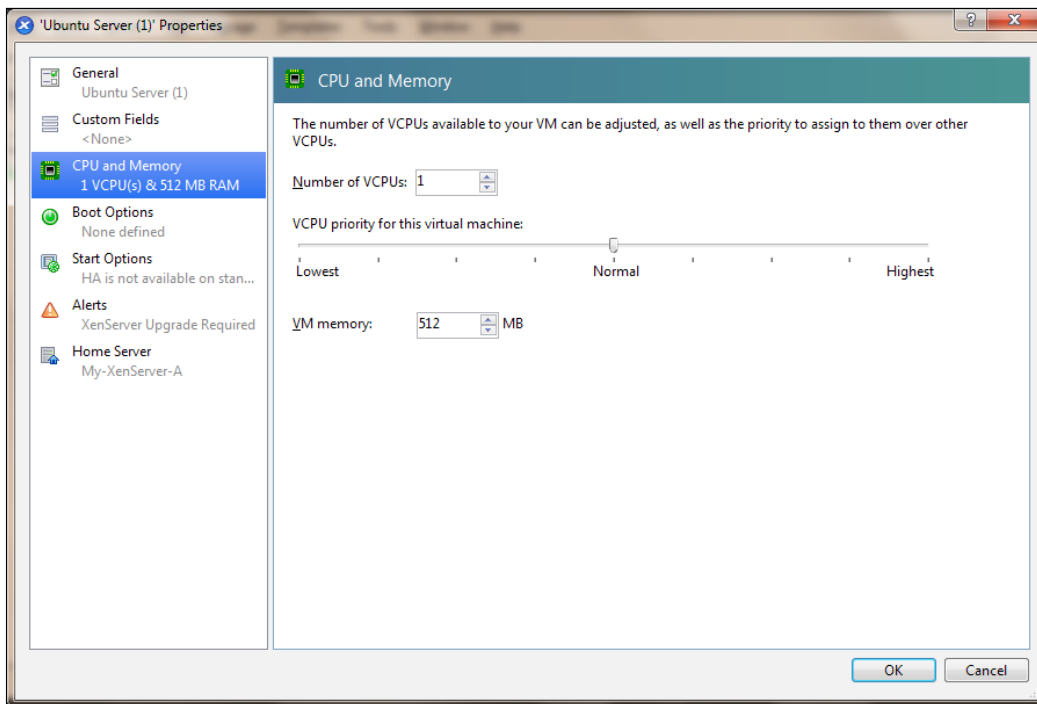
$$[\text{Memory required for hypervisor/control domain}] + [\text{Total memory in all virtual servers}] + [\text{Memory needed to support migrated VMs using XenMotion}] + [\text{Extra}]$$

The *Memory required for control domain* value is the amount of memory we want to allocate for the control domain virtual machine. By default, this value is 880 MB. The *Total memory in all virtual servers* value is the total amount of memory we want to allocate for all the virtual machines we have planned to run on a XenServer host.

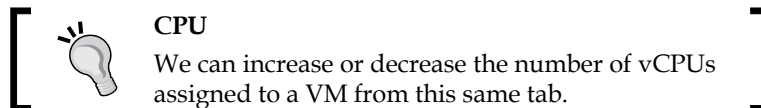
The *Memory needed to support migrated VMs using XenMotion* value is the amount of memory needed to run optimally migrated virtual machines from other XenServer hosts, if we plan to use the XenMotion feature. To better understand this point, imagine we have a XenServer pool with two hosts, each running two virtual machines. If one of the hosts fails, we should have enough free memory to run all the virtual machines on a unique host. This is a well-known rule named "N+1", where N is the total number of XenServer hosts.

The *Extra* value is an additional quantity of memory we want to reserve for future growth capacity planning or for expanding the actual memory of our existing virtual machines.

When we start creating a new virtual machine, we specify how much memory is to be reserved for it. We can change the memory assigned to a specific virtual machine by right-clicking on the VM's name in the **Resources** pane and selecting **Properties**.



Managing vCPU and memory

**CPU**

We can increase or decrease the number of vCPUs assigned to a VM from this same tab.

Usually, we don't want to over-allocate memory to a virtual machine if it is not needed, because it wastes physical memory. In the same way, we should prevent under-allocating memory because it creates poor performance. By default, when a Windows configuration achieves 75 percent of memory usage, it begins to access the page file and start swapping.

The goal is to give the VMs the right amount of memory, monitoring the operating system performance. When we have to deploy new applications on an existing or new virtual machine, we need to take care to check all the applications' requirements in order to verify that a memory bottleneck will not occur on the guest virtual machine.

We can also monitor the memory usage of our virtual machine by using the **Performance** tab in XenCenter.

Dynamic memory control

This is an advanced memory configuration and is only available in XenServer Advanced or higher editions, in which virtual machines are allowed to allocate and free up memory according to their needs. For a Dynamic Storage system, only as much HDD space gets utilized as the actual space required by VM until a particular maximum limit has been reached. Therefore, the Dynamic Memory control allows us to set a minimum and maximum limit on the amount of memory for VMs.

Let's see a particular scenario in which we have 15 GB of RAM in our XenServer and we wish to have VMs with at least 5 GB RAM. We can at max create only three VMs and if they are not using all 5 GB of their RAM, that memory is just there and not being used efficiently.

The DMC allows the machines to relieve the memory that they are not using and their donated RAM can be used on some other VM. This memory will can be made available when required. In this case, we can create more than three VMs; maybe five VMs, each using 5 GB of RAM.

A good question arises: what happens when all the VMs require 100 percent of memory at the same time? As you can imagine, there is no longer enough physical memory. This means that the VMs and the server start swapping memory, and performance degradation occurs rapidly. Also, you cannot start any new virtual machines.

Note that if the amount of memory needed for the VMs equals or is very close to the physical memory of the XenServer host, the server can suffer a lack of memory and enter into a contention state. So, consider increasing your XenServer host memory in order to satisfy the overall memory needs.

Dynamic memory control changes the allocated amount of memory to virtual machines using the concept of dynamic range. This range is based on minimum and maximum values you have configured in the virtual machine's memory properties.

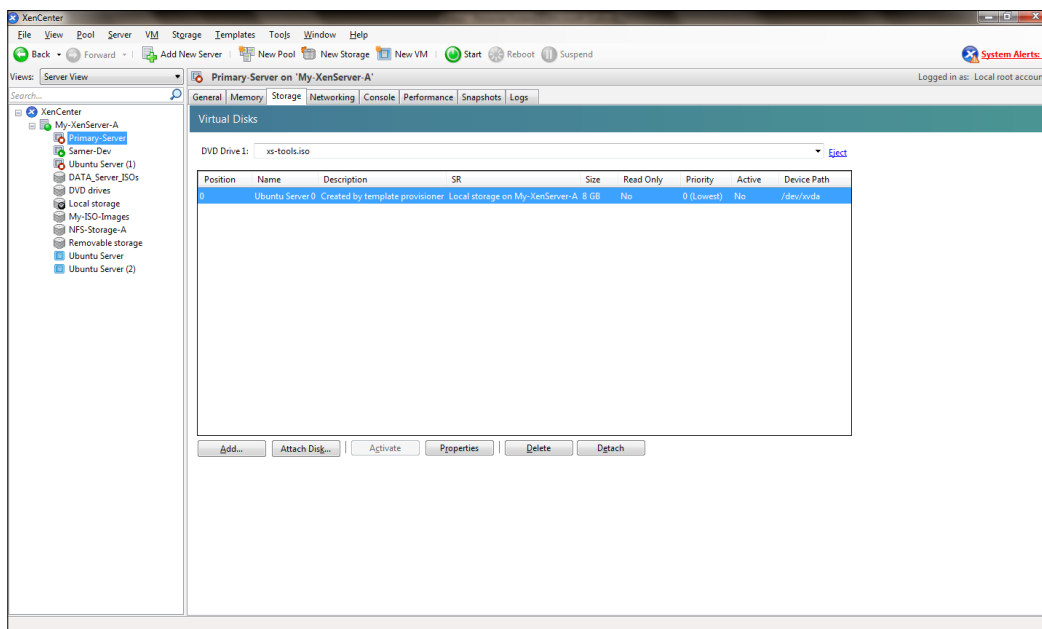
As we have understood from the previous section, XenServer will proportionally add or release free unused memory inside this range according to the needs of the virtual machines and then demand without reboot.

The allocated memory will never fall below the minimum value or exceed the maximum value; virtual machines will use memory within this range. Also remember that XenServer always tries to allocate the amount of memory defined in the dynamic range for all the running virtual machines according to the host's memory availability. If the host's memory is plentiful, XenServer allocates the value you have set on the dynamic maximum level; otherwise, if the host's memory is low, the dynamic minimum level will be used.

Managing virtual machine storage

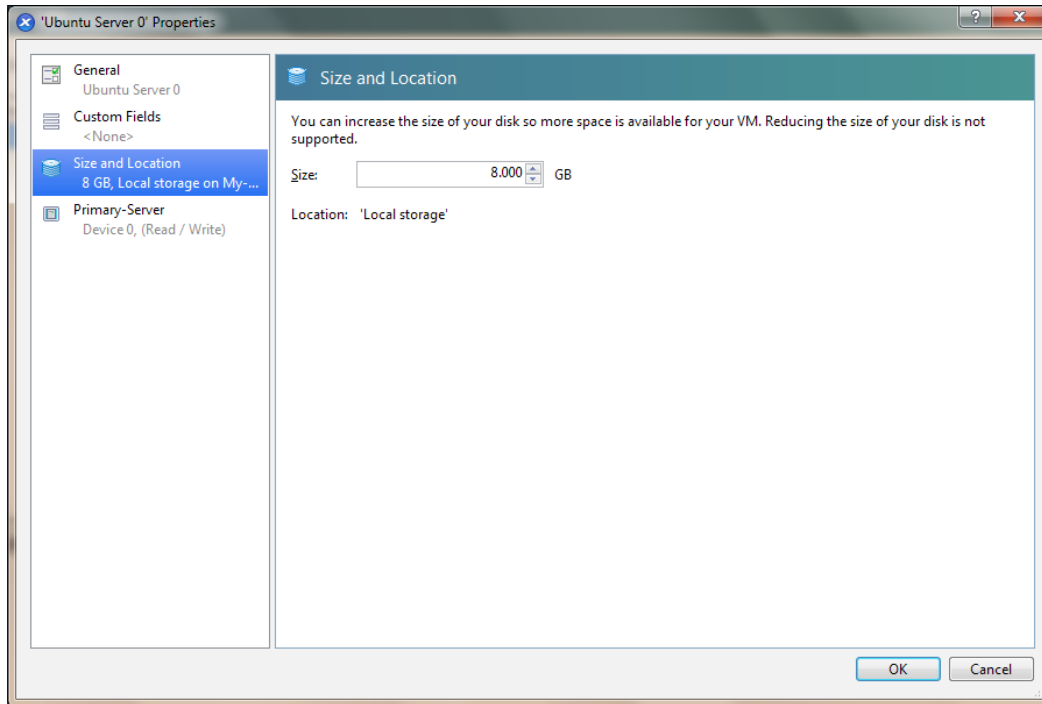
Unlike memory management, where we can increase or decrease memory, storage doesn't allow us to decrease the size of the HDD; we can only remove, increase, or attach a new storage device to our virtual machine.

To manage the storage of a VM, select any virtual machine from the **Resources** panel and click on the **Storage** tab on the right.



Storage management

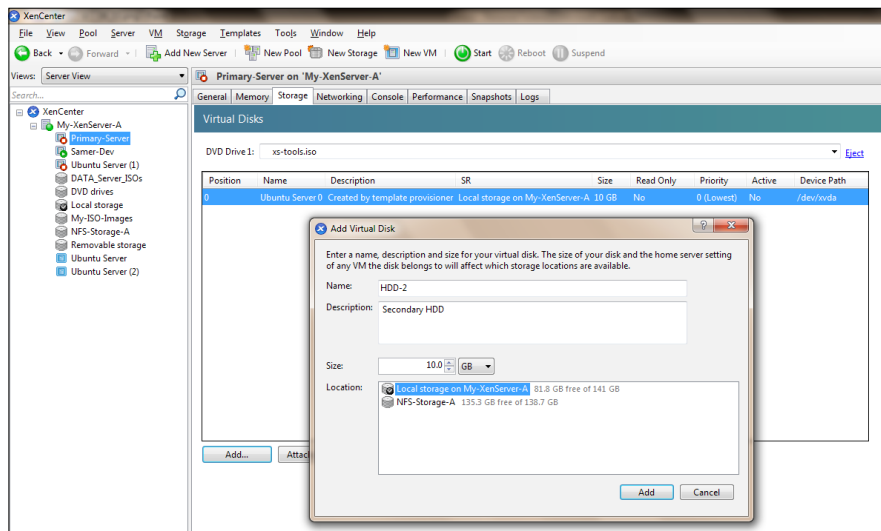
We can add a new device or check the properties of the already attached virtual HDD. Detaching a vHDD only unplugs the device; the data in it won't get deleted. Deleting a virtual machine literally erases the vHDD from the XenServer.



Increasing the disk size of a virtual machine

To increase the disk space on the first disk, start by selecting the disk and clicking on **Properties**; another window will pop up as shown in the preceding screenshot. Reconfigure the disk size, click on **OK**, and finish increasing the size.

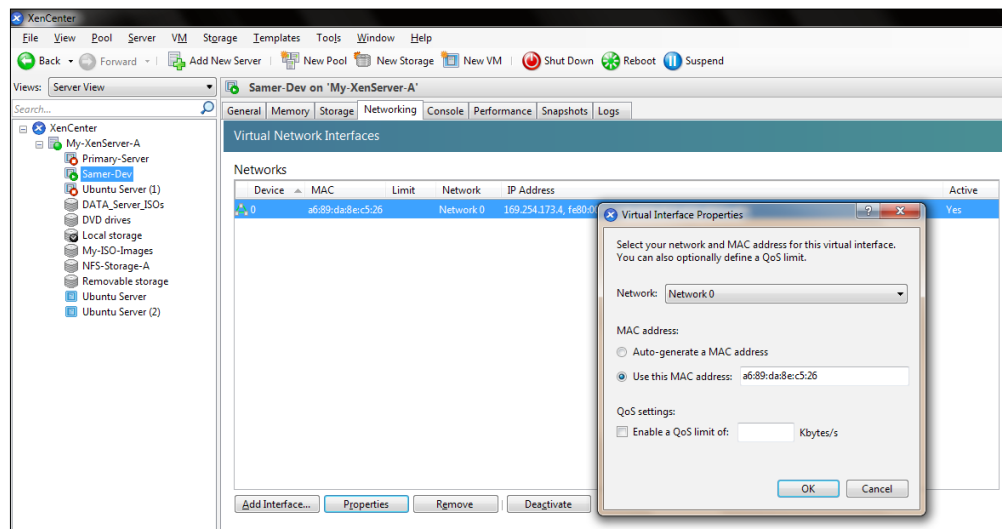
To add disk space, we need to click on **Add...** from the **Storage** tab instead of **Properties**; it is possible to have this new HDD configured in any other storage repository.



Adding another disk to a virtual machine

Managing network interfaces

We'll look at the networking in more detail and discuss advanced topics in the next chapter. Here we'll only look at the options available for us under the **Networking** tab for a virtual machine. Just like managing the storage of a VM, we need to select the VM of interest and click on the **Networking** tab.



Like storage, we can deactivate, remove, add, or change the properties of a configured NIC. There is a limit on the number of new interfaces we can add to a new VM, which we discussed in previous chapters.

All virtual machines connect to **Network 0** by default and (having the same network subnet) will be able to communicate with each other. We can change this by connecting our VM to some other network switch. We will definitely need to create a new network for this.

It is also possible to regenerate a new MAC address for the vNIC or use a predefined MAC on the interface.

One of the features of XenServer NIC management is that we can limit the bandwidth of a vNIC, so the network traffic of those machines is restricted to not go beyond that specified limit.

Importing and exporting virtual machines

Sometimes, we need to transport the virtual machines in or out of our XenServer to some other Hypervisor software, such as VMware or Microsoft virtualization. Technically, I'm talking about exporting the virtual machine so that it can be properly imported into the virtualization platform and be used effectively.

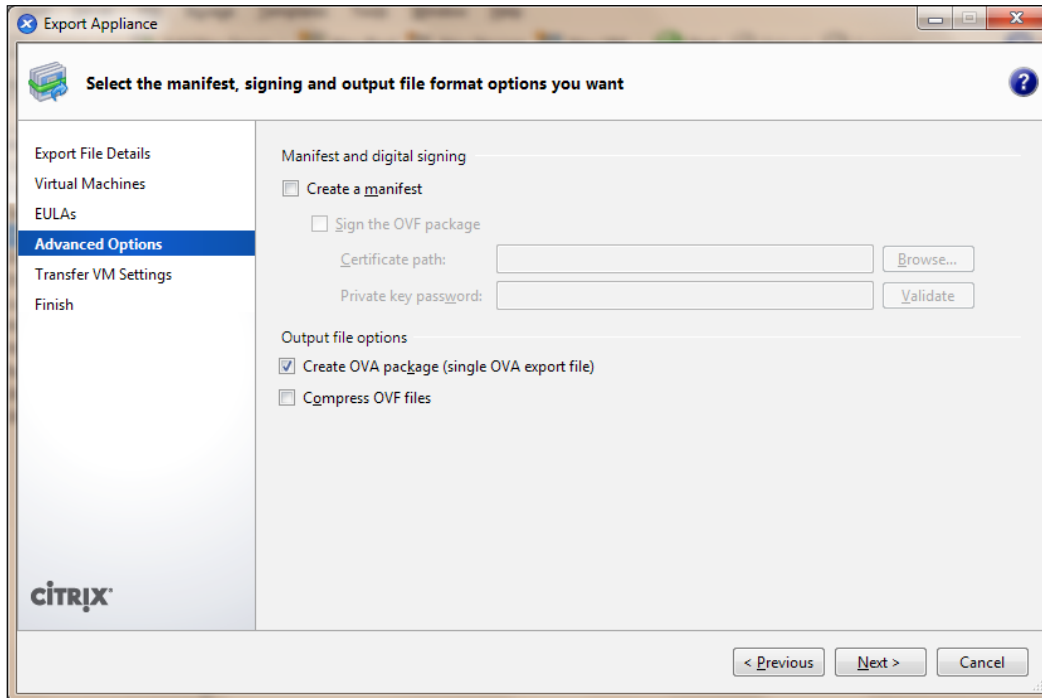
XenServer allows us to import VMs from, or export them to, a number of different formats, such as from disk images (VHD and VMDK), **Open Virtualization Format (OVF)**, and the XenServer XVA format.

Exporting a virtual machine

To begin with, let's discuss the exporting of a virtual machine. The virtual machine that is to be exported should be shut down.

1. Right-click on the virtual machine you want to export or click on the VM menu and select **Export...**
2. On the first page of the wizard, enter a meaningful name for the archive file. Specify a location where you want to save the file and select an OVF/OVA package or XVA file from the **Format** drop-down menu. Once you do this, click on **Next**.
3. Select the VMs you want to export and then click on **Next**.

- If you have selected to export the virtual machine as an OVF/OVA package, the EULAs page will be displayed. If you want to include some EULAs, this can be skipped.



Exporting a virtual machine

The **Advanced Options** page will be displayed as shown in the preceding screenshot if you select the virtual machine as an OVF/OVA package.

Here we can specify any manifest, signature, and/or output file options or just click on **Next** to continue. The manifest file is used during the importing process to verify that the files have not changed after the package was initially created.

To add a digital signature to the package, select the **Sign the OVF package** checkbox, browse to locate a security certificate, and enter the private key associated with the certificate in the **Private key password** field. We can choose to compress the OVF files and export the virtual machine as a single OVA file package.

If both the **Create OVA package** and **Compress OVF files** options are checked, the result is a compressed OVA file.

The **Transfer VM Settings** page appears for virtual machines to be exported as an OVF/OVA package. Configure networking options for the Transfer VM that XenServer will use to perform the export process and click on **Next** to continue.

Finally, the **Finish** tab will show us the summary of all the export settings. Click on **Finish** and wait for the export to finish. View the **Logs** tab to see the progress.

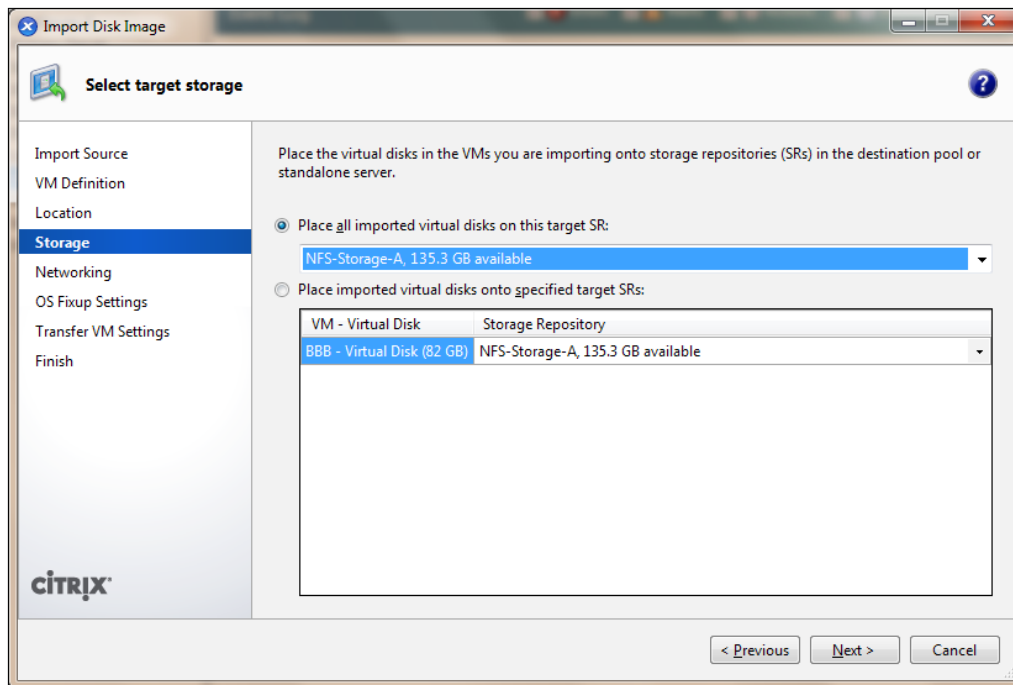
The preceding steps will export the virtual machine. We can use this exported VM on any other XenServer or platform.

Importing a virtual machine

This is almost the same process as that of exporting; only, in this case we're going to initiate an import wizard from the **File** menu. The wizard will pop up and ask us to browse for the file we wish to import. Since there are multiple formats – such as OVA and OVF, and for disks VHD and VMDK – depending upon what type of file we've selected, the wizard changes its process steps accordingly.

To import a VMDK, follow this process:

1. Open the **Import** wizard by selecting **Import** from the **File** menu.
2. Browse for the image or file you wish to import and click on **Next**.
3. Select the number of vCPUs and memory to be assigned to this new VM in the **VM Definition** step and continue to the next step.
4. In the **Location** tab, select which XenServer host to use to create this new VM from the imported file and then click on **Next**.
5. Select the storage repository where the virtual disk of the importing machine will reside.
6. Select the virtual network switch that should be used in this step. In my case, it's still **Network 0**.



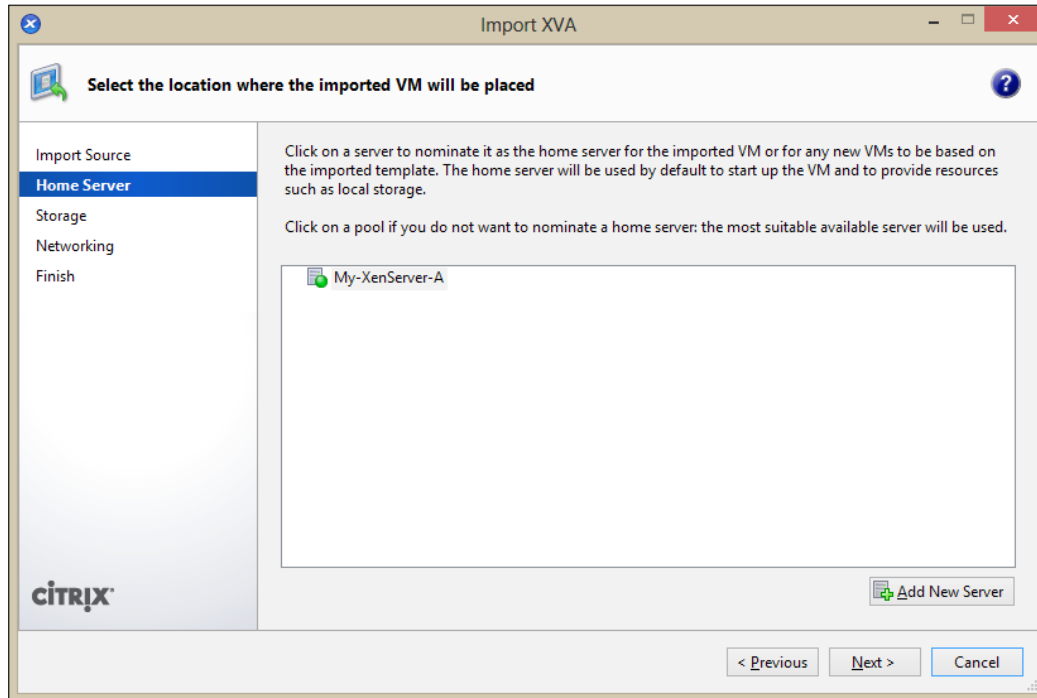
OS Fixup Settings is very interesting; it lets us fix the image that is to be imported into a XenServer-compatible format if they are exported from some other virtualization hypervisor. It mainly focuses on the boot process of the virtual machine to become compatible with the XenServer environment. This ensures that VMs from other hypervisors will work on XenServer as well.

For the **Transfer VM Settings** section, we should insert the network interface settings. Select a network from the list of network interfaces available in your resource pool or XenServer host.

By default, XenServer sets the Transfer VM to obtain network information through a **Dynamic Host Configuration Protocol (DHCP)** server. If you prefer to configure networking settings manually, click on **Use these network settings** and enter the IP address, subnet mask, and gateway. Click on **Next** to view the summary of what we're about to do.

Click on **Finish** and the **Import** process will start according to the configurations.

Like I said previously, the wizard changes its steps according to the file type, so let's see what happens when we import an XVA file.



So, as we can see, on the top bar it says **Import XVA**; likewise, if our OVF format file has some EULA or manifest associated with it, the steps for importing will change.

Virtual machine snapshots

Similar to the snapshots we take, where we save our images and look back at them and remember those moments, we can take snapshots of virtual machines; only, we can go back to that state and do something with the virtual machine. We can keep a daily snapshot of some critical development server, and if we face some issue while making some changes, we can always revert to the previous day's snapshot, when the server was known to be working fine.

Snapshots provide a fast way of creating templates that can be exported for backup purposes and can be used to quickly create new virtual machines.

When we create a snapshot, the result is a virtual machine very similar to a template. The VM snapshot contains the entire VM configuration, allowing you to export the snapshot and restore it if need be. To create a snapshot of a virtual machine, revisit *Chapter 5, Playing with VMs*.

Types of snapshots

What we did not talk about in the previous chapters were the different types of snapshots, which can be very useful to us.

Disk-only

This type stores a VM's configuration information (metadata) and disks (storage), allowing them to be exported and restored for backup purposes. A disk-only snapshot can be created on all virtual machines, Windows- and Linux-based.

Quiesced

This type takes advantage of the Windows **Volume Shadow Copy Service (VSS)** to generate an application's consistent point-in-time snapshots. The Microsoft Software Shadow Copy Provider service must be enabled and running before taking a quiesced snapshot.

Disk and memory

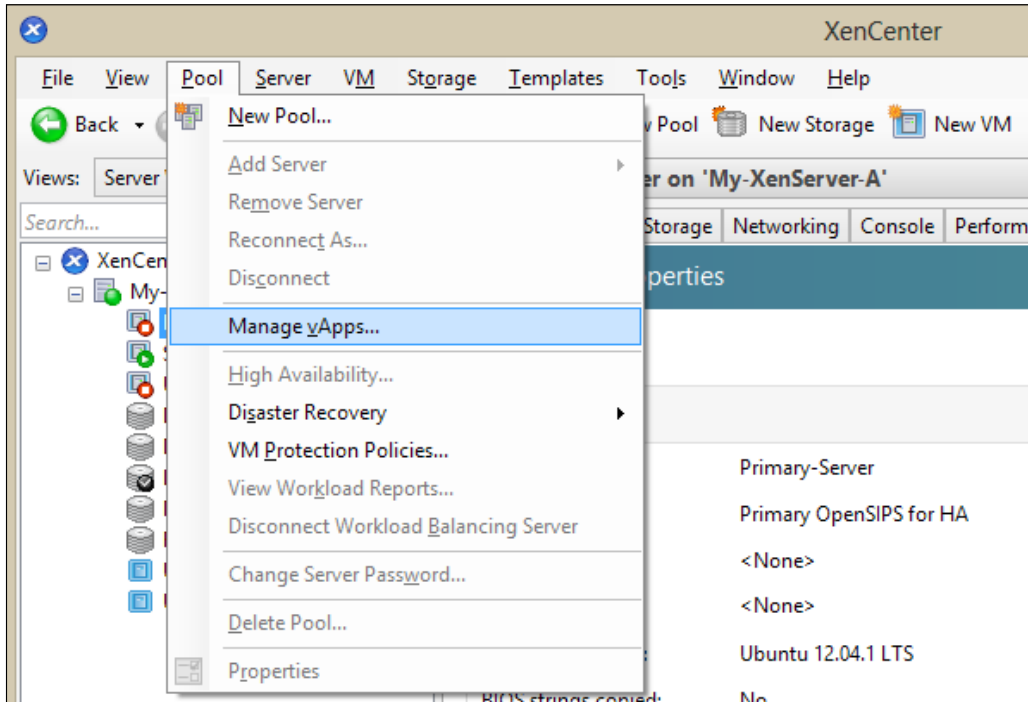
In addition to saving the disk state, this type of snapshot also saves the memory state (RAM) of a virtual machine. This can be useful if you are upgrading or patching software or want to test a new application and also want to have the option to revert to the previous state of the VM. Note that you can save the memory status only if the VM is running or is suspended. Also, during the snapshot-creation process, the VM is paused for a brief period of time and cannot be used in that period. This feature is only available in XenServer Enterprise Edition or later.

Introduction to vApps

XenServer 6 has introduced a new feature, named vApp. A vApp is a logical group of one or more VMs that can be started up as a single entity. When a vApp is started, the VMs contained within the vApp will start in a user-predefined order to allow VMs that depend upon one another to be automatically started.

This comes in handy in a situation in which some monitoring, router or firewall servers, database servers, and/or preloaded memory-cached data servers need to be in an up-and-running state before any other server turns on. So, logically, the applications using the database will continue working as soon as they become active.

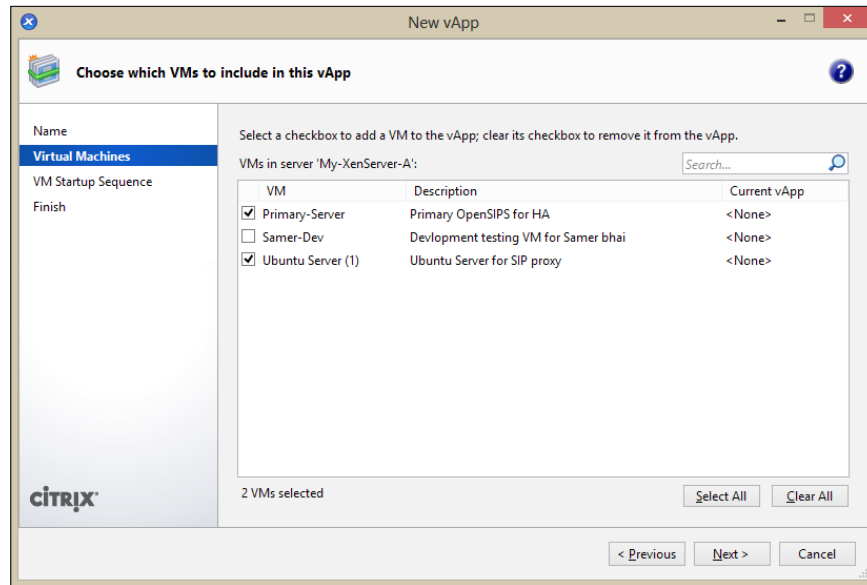
Let's create our first vApp.



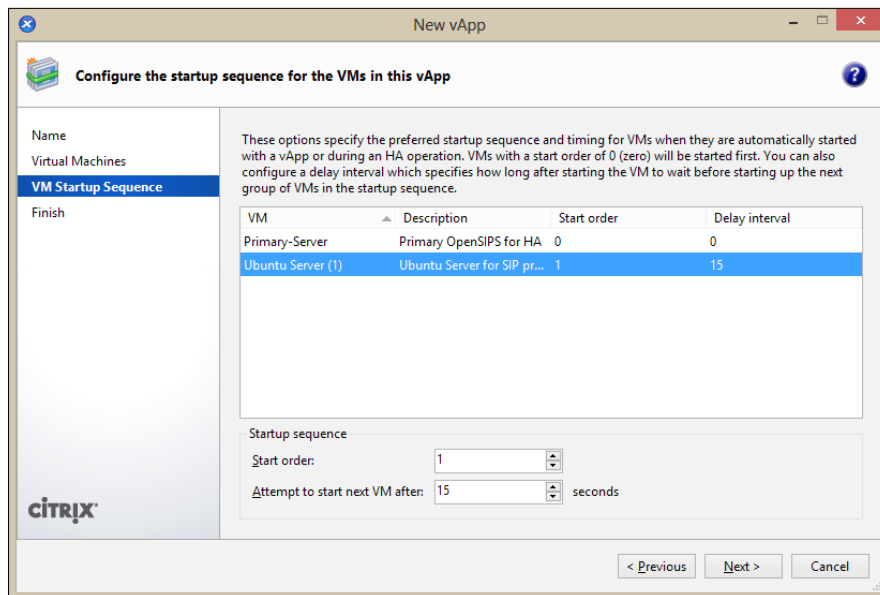
Creating a vApp

Clicking on **Manage vApps...** will pop up a new window where we can see any vApps previously created, manage them, change their properties, create a new vApp, or delete a vApp. In our case, it will be all empty, so I'll proceed to click on **New vApp...** to create a new vApp.

Enter the desired name and description and click on **Next** to select the virtual machines that should be a part of this vApp group. Note that if a VM is already a part of some vApp, it will appear next to it.



Click on **Next** to sort the selected virtual machines' power on priority and delay between the power on sequence, if needed.



Setting the startup sequence and time interval for the vApp

Click on **Next** and see the summary of the vApp that is to be created. Click on **Finish** and the vApp will appear in the vApp manager window. We can power the vApp on/off or export this vApp or import vApps from/to other XenServers.

Summary

In this chapter, we saw how to manage memory, CPU, and storage of our virtual machines, and how to change these properties using the XenCenter. Also, we looked at ways to import and export our virtual machines. We looked into the virtual machine snapshot as well as the vApp grouping of virtual machines.

The next chapter deals exclusively with the networking area of XenServer and virtual machines, and integration of virtual machines with networks.

7

Networking in XEN

In this chapter we'll be looking at the main concepts and features related to networking in XenServer, such as:

- Networking objects
- Virtual LANs
- Virtual switches and routers
- Connecting VMs to the LAN and Internet

The XenServer networking concept

Virtual networks are available in XenServer and give us the same networking options as real physical networks. Different types of networking objects enable communication between the virtual machines and the outer world outside the hypervisor. Physical **Network Interfacing Cards (NIC)** are a trivial requirement for XenServer, and using these physical network interfaces, our virtual NICs communicate with any other equipment available over the network.

Each physical network interface on our XenServer host is represented in hypervisor with an object known as PIF (physical interface). Similarly, the virtual NIC in each of the virtual machines is known as VIF (virtual interface). These PIFs and VIFs have names and descriptions and are identified by UUIDs (Unique Universal Identifier).

XenServer automatically generates a new MAC address for the NIC. When we create a new virtual interface for a virtual machine, we can also specify a MAC address manually, and in this case, the assigned MAC address is generated as a locally administered address.

Network types and objects

While installing XenServer, we configured the Primary Management interface, so it can be accessed over the network. That is the only NIC with an IP address on XenServer; the rest of the interfaces will be empty, and are used by virtual networks created inside XenServer.

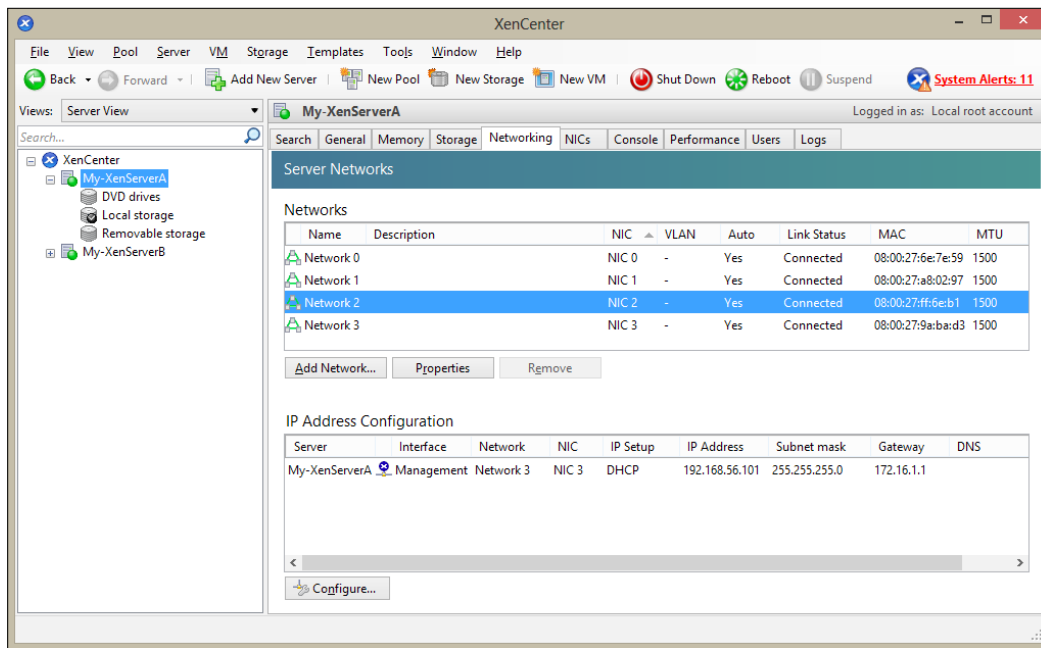
During installation, XenServer also creates a separate network for each NIC. It detects on the host and uses the additional NICs on the host for managing the virtual machines' traffic only. This means that if the host has, for example, three NICs, XenServer creates three networks, such as Network 0, Network 1, and Network 2.

The networking pool

Networking is a pool-level feature in XenServer. This means that XenServer automatically replicates the network settings on the pool master to the joining hosts or the newly created pools. Furthermore, when you change networking on the pool master, XenServer synchronizes all the hosts in a pool to use the same network settings.

Network types

In XenCenter, select the XenServer host, go to the **Networking** tab and we'll see the configured networks and their configurations.



Click on **Add Network** and we'll be presented with different types of networks to choose from, which are:

- **A single-server private network:** This type is not attached to a physical network interface and can be used to provide connectivity between the virtual machines that run on a specific host – network traffic remains isolated on that host and cannot reach other hosts. Just to clarify, think of a bubble (your host) where nothing can go out (your network traffic), but can only go in. Usually, we also refer to this type of network as `Internal`.

In the previous versions of XenServer, we could create only single-server private networks that allowed VMs running only on the same XenServer host to communicate with each other.

- **A cross-server private network:** This type is similar to the single-server private network type but differs from it because in this case, network traffic can also reach virtual machines running on different hosts in the resource pool and not only on a single, given host.

This type of network provides functionalities similar to VLANs, but unlike them, cross-server private networks provide isolation without requiring the typical configuration that a network administrator does on a physical network switch to implement VLANs.

Remember that the following conditions must be met when you want to create a cross-server private network:

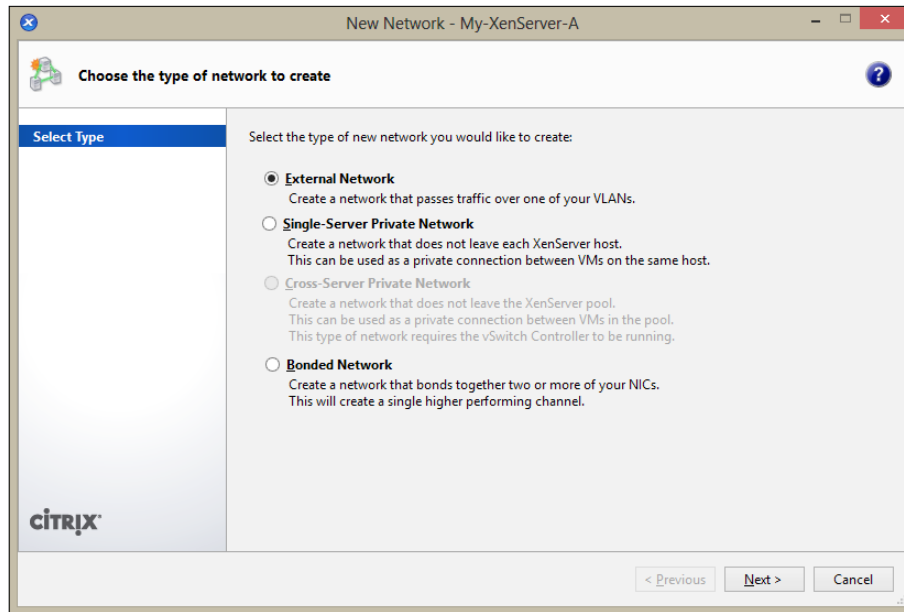
- All of the hosts in the pool must be using XenServer 6.0 or higher
- All of the hosts in the pool must be using vSwitch for the networking stack
- The vSwitch Controller must be running and you must have added the pool to it
- The cross-server private network must be created on a NIC configured as a management interface

A cross-server private network requires a Citrix XenServer Advanced Edition or higher.

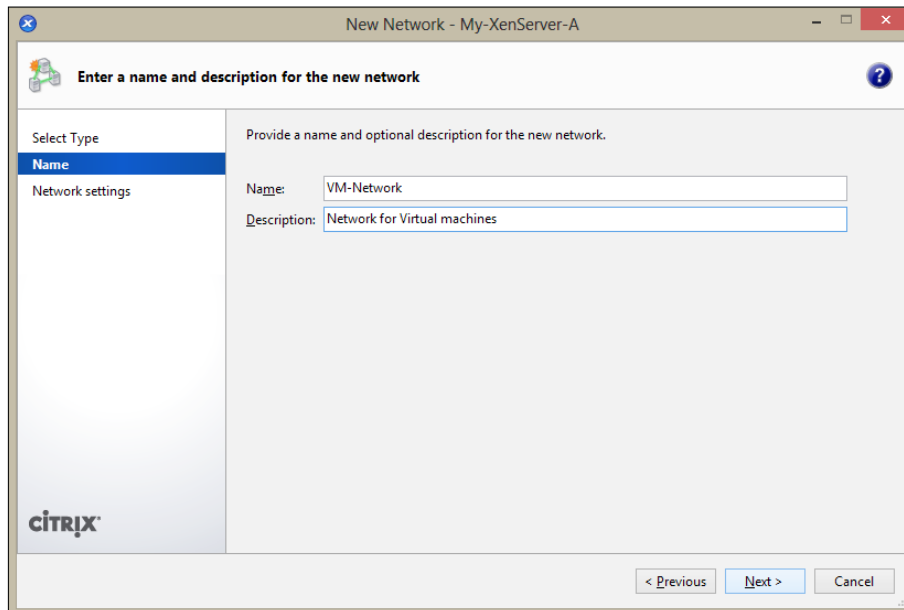
- An **external network**: This type is attached to a physical network interface and enables a virtual machine to connect to resources available through the physical network interface card installed on the host.
- A **bonded network**: In this type, a network is attached to a "bond," created by combining two network interface cards in order to have a single, high-performing, and redundant channel between your virtual machines and the network. This concept is the same as "Teaming".

Using virtual networks

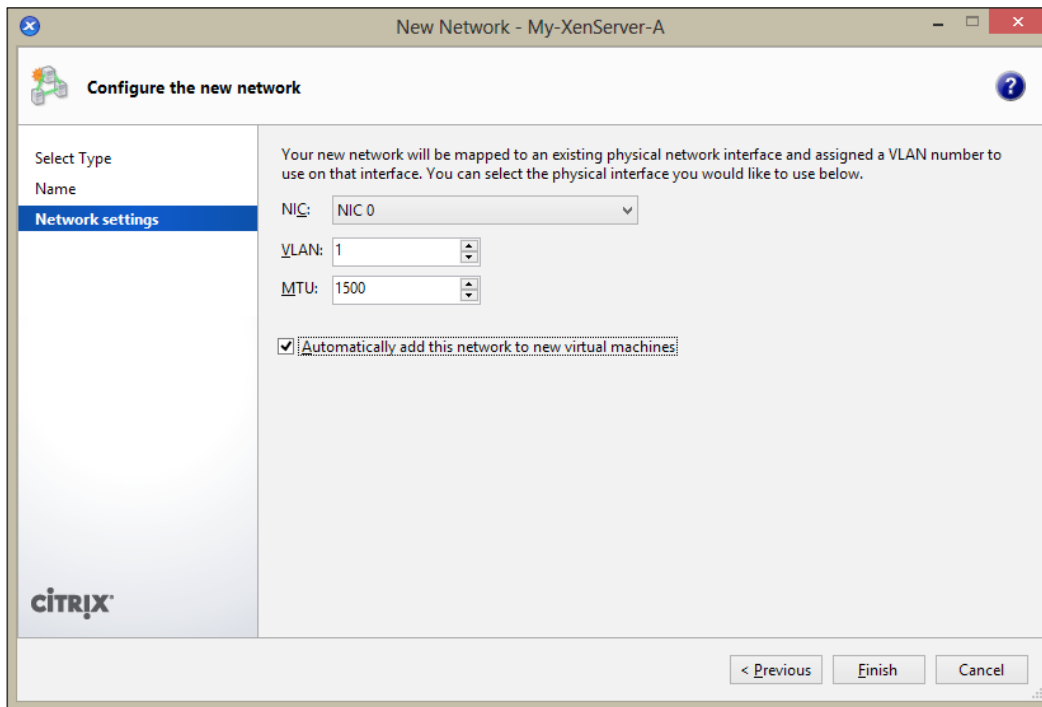
Let's start creating our new virtual network. On the **Networking** tab of the XenServer host, click on **Add Network...** and we'll be given four choices.



Select **External Network** so that traffic from virtual machines can reach outside the XenServer host. Click on **Next**.



Write the **Name** and **Description** of the new network and click on **Next**. Select the physical NIC on which this new network has to be attached, and if needed, select the option to automatically add this network in the new virtual machines created from then on. We usually don't need to modify the default **Maximum Transmission Unit (MTU)** of this network.



Click on **Finish** and that'll add a new network in our virtual environment.

The screenshot shows the XenServer Network Configuration interface. At the top, there are tabs for Search, General, Memory, Storage, Networking (selected), NICs, Console, Performance, Users, and Logs. Below the tabs is a header for 'Server Networks'. Underneath, there is a section titled 'Networks' containing a table with the following data:

Name	Description	NIC	VLAN	Auto	Link Status	MAC	MTU
Network 0		NIC 0	-	Yes	Connected	00:1c:c0:73:b1:91	1500
New Network		NIC 0	1	Yes	Connected	-	1500

Below the table are three buttons: 'Add Network...', 'Properties', and 'Remove'. Underneath these buttons is a section titled 'IP Address Configuration' containing another table:

Server	Interface	Network	NIC	IP Setup	IP Address	Subnet mask	Gateway	DNS
My-XenServerB	Management	Network 0	NIC 0	Static	172.16.37.150	255.255.0.0	172.16.2.50	172.16.2.11,172.16.2.12

VLANs with virtual machines

VLANs are used everywhere to separate traffic on the physical LAN, for different reasons, for example, scalability, security, and network management. XenServer provides us with the option to implement VLANs to separate different types of traffic; for example, the database's virtual machines are connected to one VLAN while management interfaces are connected to another. The IEEE 802.1Q protocol is used mostly in VLANs, in which each network frame contains a tag identifier, which defines the VLAN it belongs to. The network packets that don't belong to any VLAN don't have these tags and hence are referred to as "untagged traffic".

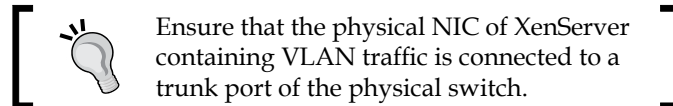
The VLAN configuration is usually made on the network switches. The network administrator can configure a switch's port as an access port or as a trunk port. The access port on a switch has a specific VLAN tag/ID, and manages the traffic with that VLAN tag only.

A trunk port on a switch has no specific VLAN tag, and hence can carry traffic related to all VLANs.

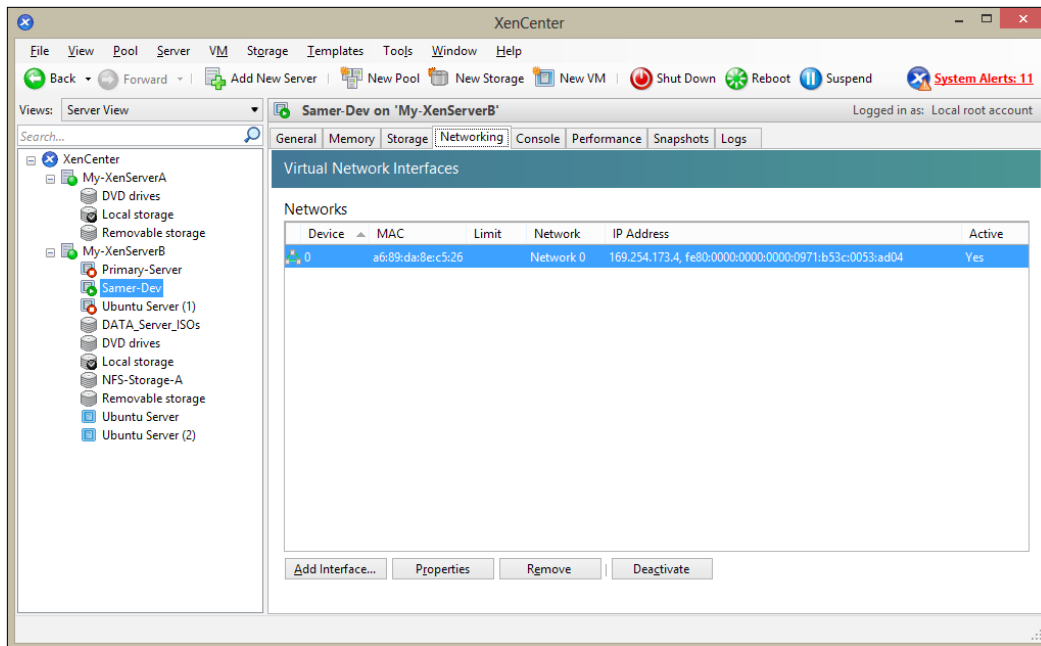
So, practically, access ports are usually used for connecting with hosts, and trunk ports on a switch are used to interconnect with another switch's trunk port, so that traffic with different VLAN tags can flow between the hosts, distributed on the connecting switches, via the trunk port.

Switch ports configured in "trunk mode" are typically used for connecting guest virtual networks. In this case, XenServer will tag and untag the network traffic with the VLAN identifier.

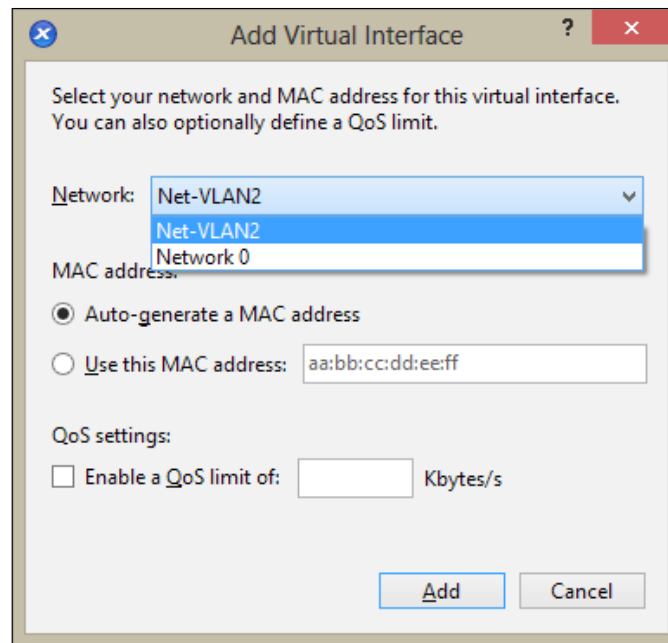
Configuring a specific VLAN tag on a network is very easy. Refer back to the previous pages where we created a new network. In network settings, we can set a different VLAN identifier.



To have the virtual machines route traffic out using that VLAN, the virtual machine must be connected to the particular tagged network switch. For this, click on a virtual machine from the **Resource** pane and select the **Networking** tab.



Click on **Add Interface...** and select the network with a new VLAN.



That will add the new interface on the virtual machine with that VLAN tag. Virtual machines with this VLAN tag will be able to communicate with each other.

NIC bonds

Using NIC bonding, an administrator configures two physical network interfaces together so that they logically function as one network card. If one NIC in the bond fails, the host's network traffic is automatically redirected through the second NIC. The behavior of bonded interfaces differs based on the functionality the bonded interfaces are used for; the first of the bonded NICs is used for management interfaces while the second NIC provides only failover for the management traffic. When we create a bond dedicated to a virtual machine's traffic, bonding provides not only failover but also load balancing.

The best practice is to consider making bonded interfaces more reliable by using two separate physical NICs, instead of one dual NIC card. Another best practice is connecting the network cables to separate switches – if you connect one of the links to a second redundant switch, and a NIC or switch fails, traffic is directed to the other link.

There are two different types of bonds: **Active/Active** and **Active/Passive**.

Active/Active is the default bonding mode and both NICs are used in order to balance the virtual machine's traffic. The bond used for the management interface will behave as explained earlier, and only one interface will remain active and the other will be used in case of failure.

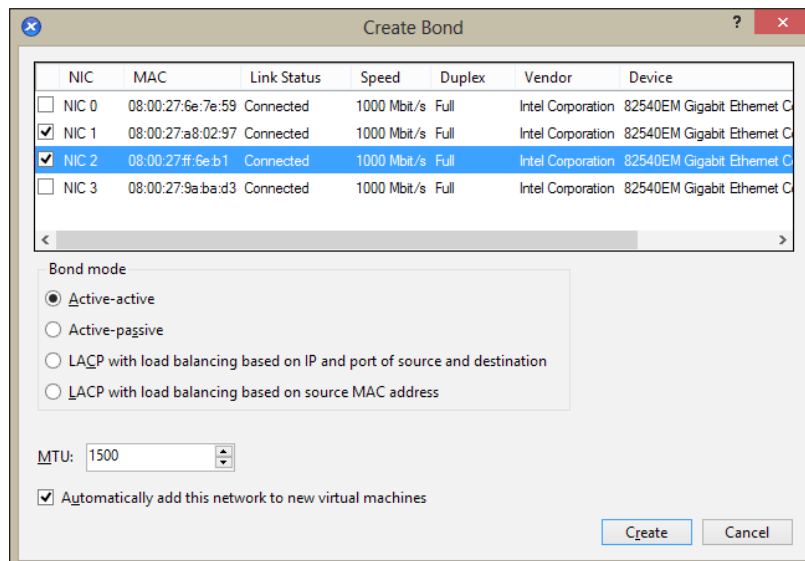
As the name Active/Passive suggests, only one NIC remains active and the other one will become active in case the active one fails.

There is another option available to us, which is **Link Aggregation Control Protocol (LACP)** bonding. This mode provides active-active bonding, where traffic is balanced between the bonded NICs. Unlike the active-active bond in a Linux bridge environment, LACP can load balance all traffic types. We can have this based on either the IP port of the source and destination, or on the MAC address of the traffic source.

Use LACP with load balancing based on source MAC option to balance traffic in an environment where you have several VMs on the same host. This option is not suitable if there are fewer virtual interfaces (VIFs) than NICs, as load balancing is not optimal because the traffic cannot be split across NICs.

Creating an NIC bond

In the **Resource** pane, click on **XenServer host** and select the **NICs** tab located on the right side. We'll be seeing all the physical interfaces connected to the XenServer host. Select any one of the NICs and click on **Create bond....**



Select the two interfaces we want to be bonded together, and select the type of bonding. **Active-active** is the default bonding, so let's proceed with it. Since this will become a new virtual network in our environment, it's always a good choice to have this new network interface connected with the new virtual machines.

Search	General	Memory	Storage	Networking	NICs	Console	Performance	Users	Logs
Network Interface Cards									
Interfaces									
NIC	MAC	Link Status	Speed	Duplex	Vendor	Device	PCI Bu		
Bond 1+2	08:00:27:a8:02:97	Connected	2000 Mbit/s	Full					
NIC 0	08:00:27:6e:7e:59	Connected	1000 Mbit/s	Full	Intel Corporation	82540EM Gigabit Ethernet Controller	0000:00		
NIC 1	08:00:27:a8:02:97	Connected	1000 Mbit/s	Full	Intel Corporation	82540EM Gigabit Ethernet Controller	0000:00		
NIC 2	08:00:27:ff:6e:b1	Connected	1000 Mbit/s	Full	Intel Corporation	82540EM Gigabit Ethernet Controller	0000:00		
NIC 3	08:00:27:9a:ba:d3	Connected	1000 Mbit/s	Full	Intel Corporation	82540EM Gigabit Ethernet Controller	0000:00		

Bonding is combining two NICs, hence in XenServer we see a new NIC created, which is known as bond master. The interfaces which were bonded to create this bond master are called NIC slaves. The NIC bond can be connected to XenServer network and used like any other interface. We can create NIC bonds in XenCenter from the **NICs** tab or from the server's **Networking** tab (**network type = Bonded Network**).

Deleting an NIC bond

If reverting a server to a non-bonded configuration, be aware of the following requirements:


- When creating a bond, all virtual machines with virtual network interfaces that use the bond must be shut down prior to destroying the bond. After reverting to a non-bonded configuration, reconnect the virtual network interfaces to an appropriate network.
- Move the management interface to another NIC using the **Management interfaces** dialog box before you delete the bond, otherwise connections to the server (including XenCenter) will be dropped.

To delete a bond, perform the following steps:

1. Select the server in the **Resource** pane, then click on the **NICs** tab.
2. Click on **Delete Bond**.

Virtual switches and routers


XenServer 6.0 supports Open vSwitch—it is a production quality, multilayer, virtual switch, licensed under the open source Apache 2.0 license. It is designed to enable massive network automation through programmatic extension, and supports standard management interfaces and protocols such as NetFlow and IEEE 802.1ag.

 You can find more information on Open vSwitch at <http://openvswitch.org/>.

Open vSwitch is enabled by default in XenServer 6.0. In order to provide a centralized administration of the virtualized networking configuration based on vSwitch and other features, such as Quality of Service (QoS) policies or cross-server private networks, Citrix has developed a virtual appliance known as the **Distributed vSwitch (DVS)** Controller that you can install and configure in your XenServer environment.

The Distributed vSwitch Controller is provided by Citrix as a .xva package and needs to be imported in our XenServer.

We can also look at any available software routers. Vyatta is one such software-based router that can be installed on any VM just like any ordinary Linux OS. It can be used to implement all the standard routing protocols such as RIP, OSPF, and BGP. It can also be used to function as a firewall. We can also use it to create VPNs of different types. Using such router tools in our XenServer can give us much more power over our virtual network.

 Visit <http://www.vyatta.org> to learn more about it.

Summary

In this chapter we learned about the networking objects and their concepts. We also touched on how to use VLANs and creating NIC Bonds. We briefly looked at the virtual switches and routing tools.

8

Advanced Concepts

In the previous chapter we learned about networking concepts in XenServer. In this chapter we'll look at some advanced topics, including the following:

- XenConvert
- High Availability
- User roles and permissions
- Distributed Virtual Switch Controller

Introduction to XenConvert

Citrix XenConvert is a tool to convert physical machines to virtual machines; that is, a **Physical-to-Virtual (P2V)** conversion tool. It converts a Windows-based workload from a server or desktop machine to a virtual machine in a XenServer, virtual appliance, virtual disk, or virtual disk connected to Provisioning Services.

A conversion involves copying the contents of the workload and configuring its operating systems for XenServer. So, the workload can be applications and data besides the base Windows operating system.

XenConvert can convert from a single server or desktop (physical machine) running Windows to a provisioned virtual disk or VHD that Provisioning Services can import.

A good scenario would be converting a physical Windows server with all the applications hosted in it to a virtual machine for pre-staging, so that the hosted applications' behavior can be tested after adjustments. This can be the operating system upgrade or application updates/upgrades.

System requirements

Since XenConvert can only convert Windows-based hosts and installs on the same host, the requirements are pretty much the same, as follows:

- Operating system: Windows XP, Windows Vista, Windows 7, Windows Server 2003 (SP1 or later), Windows Server 2008 (R2)
- .Net Framework 4.0
- Disk Space: 40 MB free disk space
- XenServer version 6.0 or 6.1

Converting a physical machine to a virtual machine

Let's take a quick look at how to convert a physical machine to a virtual machine. First we need to install XenConvert on the source physical machine.

 We can download XenConvert from this link:
<http://www.citrix.com/downloads/xenserver/tools/conversion.html>.

Once the standard Windows installation process is complete, launch the XenConvert tool; but before that we need to prepare the host machine for the conversion.

 To know more about XenConvert, refer to the XenConvert guide at <http://support.citrix.com/article/CTX135017>.

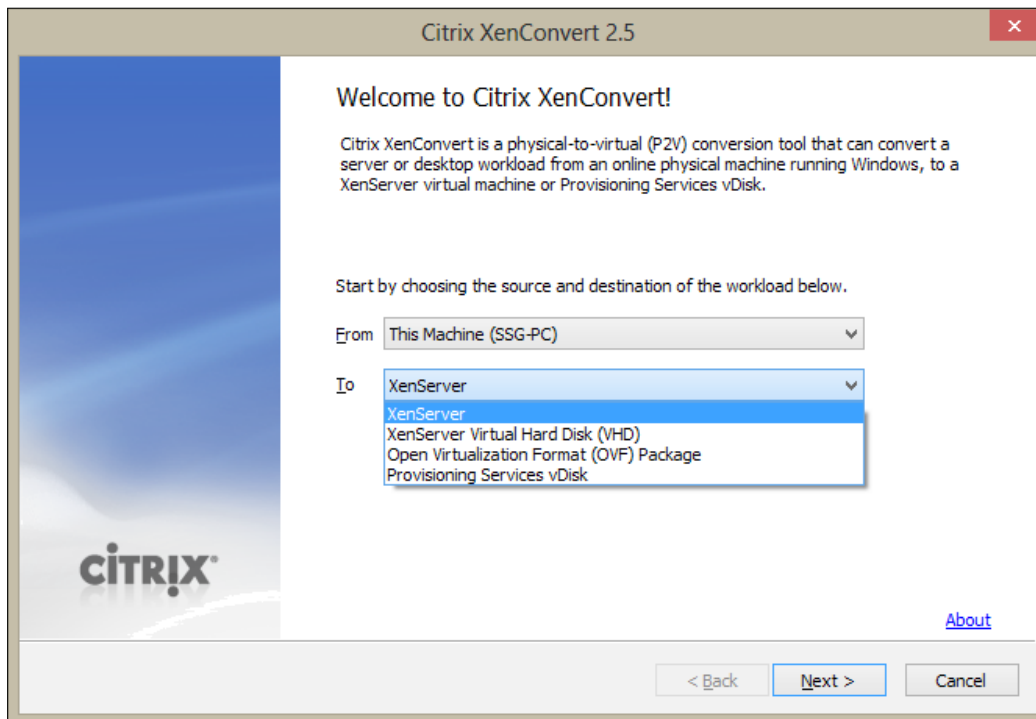
Preparing the host machine

For best results, prepare the host machine as follows:

- Enable Windows Automount on Windows Server operating systems.
- Disable Windows Autoplay.
- Remove any virtualization software before performing a conversion.

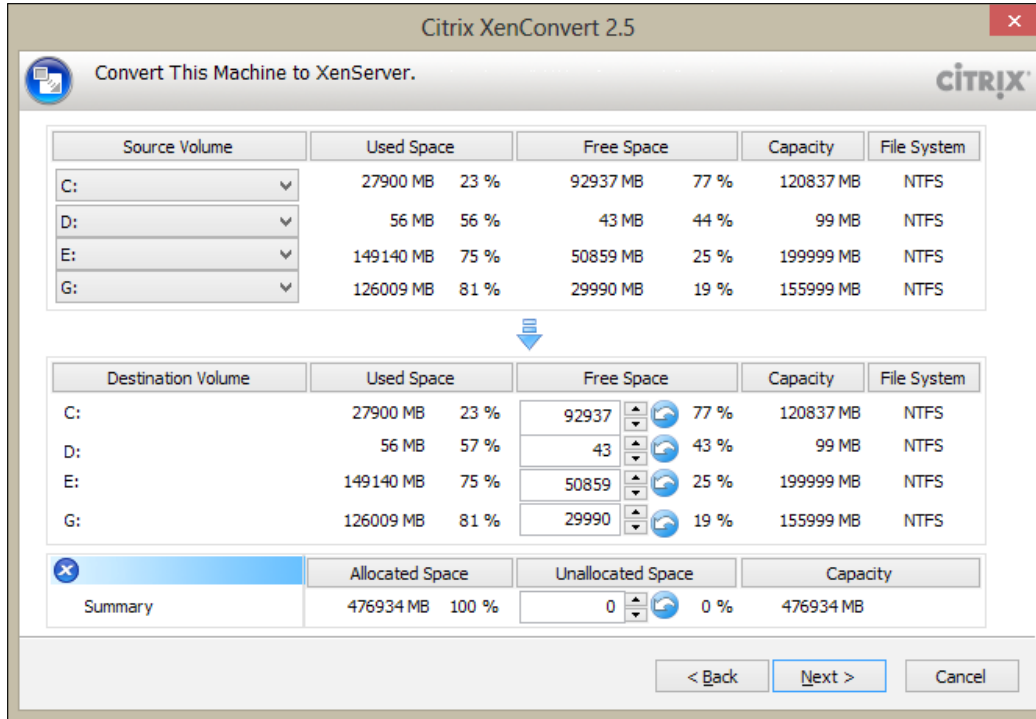
- Ensure that adequate free space exists at the destination, which is approximately 101 percent of used space of all source volumes.
- Remove any network interface teams; they are not applicable to a virtual machine.

We need to run the XenConvert tool on the host machine to start the physical-to-virtual conversion.



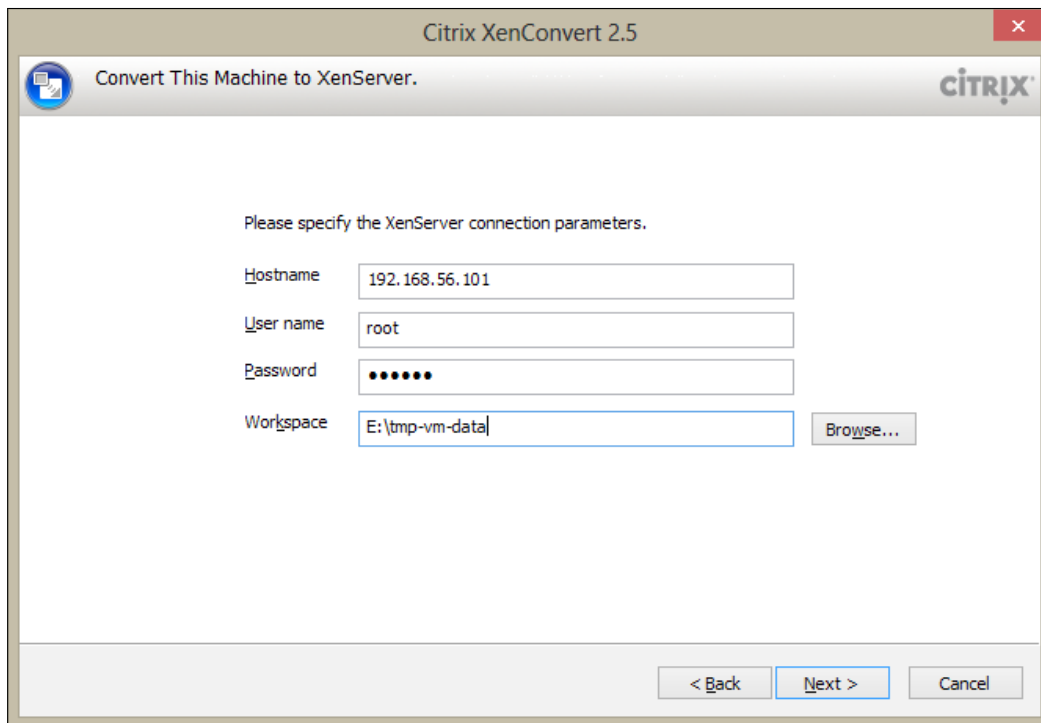
We can convert the physical machine directly to our XenServer if this host machine is accessible. The other options are to convert to VHD, OVF, or vDisk, which can be imported later on to XenServer using the methods we learned in previous chapters. These options are more useful if we don't have enough disk space or connectivity with XenServer.

I chose XenServer and clicked on **Next**.



We can select multiple partitions to be included in the conversion, or select **none** from the drop-down menu in **Source Volume** and those disks won't be included in the conversion. We can also increase or decrease the size of the new virtual partition to be allocated for this virtual machine. Click on **Next**.

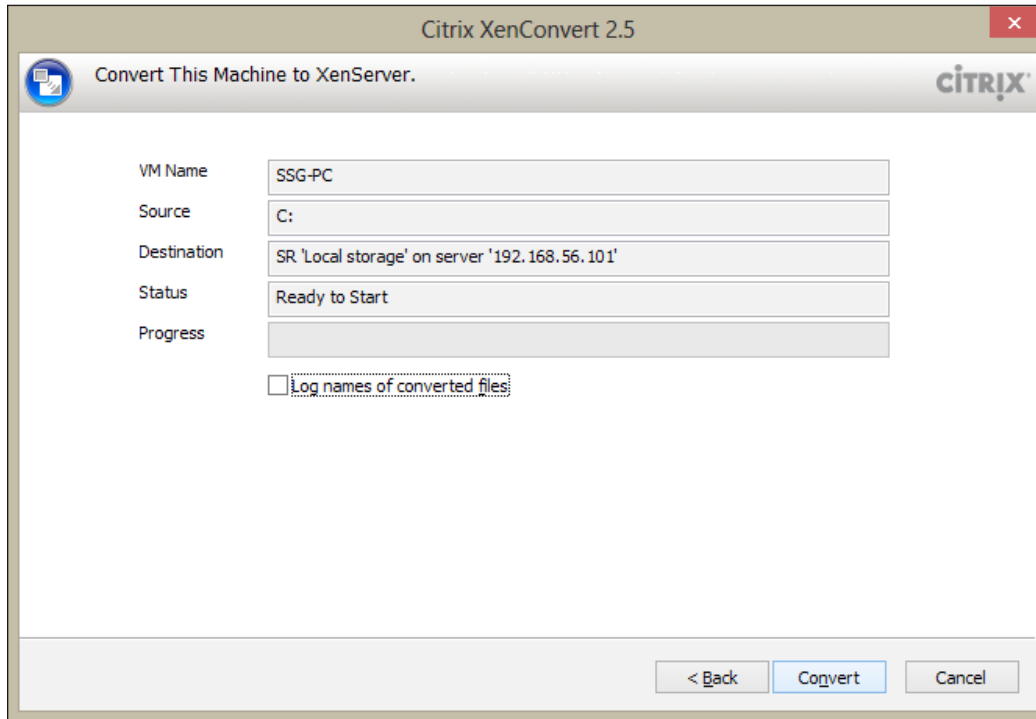
We'll be asked to provide the details of the XenServer host.



The screenshot shows a dialog box titled "Citrix XenConvert 2.5" with a close button (X) in the top right corner. The main title bar reads "Convert This Machine to XenServer." and the Citrix logo is in the top right. The dialog contains the instruction "Please specify the XenServer connection parameters." and four input fields: "Hostname" with the value "192.168.56.101", "User name" with the value "root", "Password" with masked characters "•••••", and "Workspace" with the value "E:\tmp-vm-data". A "Browse..." button is located to the right of the Workspace field. At the bottom, there are three buttons: "< Back", "Next >" (highlighted with a blue border), and "Cancel".

The hostname needs either an IP address or a FQDN of the XenServer; a username and password are standard login requirements. In the **Workspace** field, enter the path to the folder to store the intermediate OVF package that XenConvert will use during the conversion process. XenConvert will store the OVF package in the path we give.

Click on **Next** to select the storage repositories found with XenServer and continue to the last step, in which we'll be provided with the summary of the conversion.



Soon after the conversion is completed, we'll be able to have this new machine in our XenCenter. We'll need to have XenServer Tools installed on this new virtual machine.

High Availability

High Availability (HA), as the name suggests, is the mechanism of ensuring continuity of services even after the primary service-providing infrastructure has failed. In terms of XenServer, HA ensures that mission and business-critical virtual machines (VMs) are continuously operating within a resource pool.

XenServer continually monitors the health of the hosts in a pool. In case of current VM host failure, the HA mechanism automatically moves protected VMs to a healthy host. In case the master host fails, HA selects another host to take over the master role automatically so that we can continue to manage the XenServer pool.

Traditionally, only one mechanism of monitoring the health of hosts is used, and that's network interface. Redundancy has been implemented in XenServer for the heartbeat mechanism, and besides the network-based heartbeat mechanism a storage-based heartbeat mechanism is used, known as **Heartbeat Storage Repository**.

XenServer automatically creates small virtual disks in the Heartbeat SR. Each host in the pool writes to this small virtual disk to indicate to each other that they are alive.

To successfully enable HA on our pool, this shared storage repository must be based on iSCSI, Fibre Channel, or NFS.

It is vital to ensure that network adapters are bonded for resilience and storage interfaces using dynamic multipathing are supported. This ensures that any single adapter or wiring failure does not result in any availability issues.

Failure planning

Another important concept to understand is **number of host failures to tolerate (nhtol)**. This is the maximum number of hosts that can fail before the pool can no longer guarantee that all protected VMs in the pool will continue to run. This nhtol value calculation is called failure planning; XenServer dynamically calculates the amount of resources that need to be available in order to ensure that the virtual machines are restartable. If there is a new virtual machine created on XenServer due to which a new plan is not feasible, for example, lack of available memory or disk space across the pool, XenServer stops the user from creating this new virtual machine and returns an *Over-Commitment* error message.

The hosts to be configured with High Availability must be in the same resource pool and must have Advanced Edition or a higher license installed on them. Also, it's very logical to have all the hosts in HA mode configured with static IP addresses.

Similarly, virtual machines running on hosts with HA should not have any binding to local CD/DVD drives and should have the network interfaces connected to a pool-wide network.

Distributed vSwitch Controller

We briefly looked at the vSwitch and DVS topic in the previous chapter; it's now time to get some more insights about the Distributed Virtual Switch. DVS sits on top of multiple vSwitches and gives an overall effect of one single vSwitch. Therefore all the VIFs and PIFs inside each vSwitch get included in the DVS Controller. Imagine one central vSwitch for all the hosts.

By definition, the DVS Controller is a XenServer virtual appliance that controls multiple vSwitches. The vSwitch Controller supports fine-grained security policies to control the flow of traffic sent to and from a VM. It provides detailed visibility into the behavior and performance of all traffic sent in the virtual network environment. A distributed virtual switch greatly simplifies IT administration within virtualized networking environments, as all VM configuration and statistics remain bound to the VM even if it migrates from one physical host in the resource pool to another.

The Distributed Virtual Switch Controller is available in Citrix XenServer Advanced Edition or higher. The standard DVS Controller configuration supports deployments of up to 16 XenServers and 256 virtual interfaces connected to it. The maximum supported limit is 64 XenServers and 1024 virtual interfaces.

The DVS virtual machine default hardware configuration is as follows:

- 2 vCPUs
- 2 GB RAM
- 16 GB disk space

If the DVS Controller connects to 64 XenServers and 1024 interfaces, the CPUs and the memory allocated should be doubled.



To learn further about Distributed Virtual Switch, read the article at <http://support.citrix.com/article/CTX130423>.

Workload Balancing

Workload Balancing (WLB) is a XenServer component, distributed as an .xva package by Citrix, that offers us reporting features about our virtual environment's performance and usage. Also, Workload Balancing helps in placing our virtual machines on the best possible hosts in the pool, according to resource availability.

Since WLB captures performance data, we use this component to generate reports, known as Workload Balancing Reports, about our virtualized environment.

These reports provide us with information about the pool or host's health, auditing, optimizations, and placement history. Also, we can monitor the virtual-machine resource-usage history so that this can help us in compiling statistics and performing capacity planning.

When deployed to manage virtual machine workloads, Workload Balancing can do the following:

- Balance virtual-machine workloads across hosts in a XenServer resource pool
- Determine the best host on which to start a virtual machine
- Determine the best host on which to resume a virtual machine that we powered off
- Determine the best host to move a virtual machine to when a host fails
- Determine the optimal server for each of the host's virtual machines when we put a host into or take a host out of Maintenance Mode.

As you can imagine, running virtual machines consumes computing resources, such as CPU and memory, on the physical host. This reduces the host's available resources.

When Workload Balancing evaluates that the resource utilization of virtual machines on a host is exceeding the performance threshold, it suggests that we move the VM to a host in the pool with available resources in order to balance workloads. This action is known as a **recommendation**.

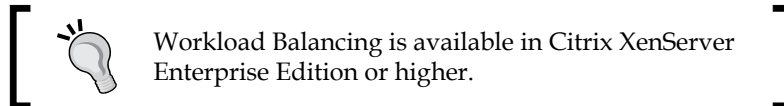
Depending on our preference, Workload Balancing can accomplish these tasks automatically or prompt us to accept its rebalancing and placement recommendations.

Furthermore, if our hosts are configured with Wake-on-LAN-enabled network cards, we can also configure Workload Balancing to power off hosts automatically at specific times of the day, for example, because we want to reduce power consumption at night.

We can configure Workload Balancing behavior in order to maximize our virtual machine's performance or the number of virtual machines running on a host. In the latter case, we set Workload Balancing in order to achieve maximum density on a host.

When we choose to maximize performance, Workload Balancing recommends placing virtual machines on hosts to ensure that maximum amount of resources are available for each running virtual machine.

When we choose to maximize density, Workload Balancing's goal is to reduce the number of hosts powered on in the pool. To do this, it recommends placing virtual machines on as few hosts as possible.



Role-based administration control

So far, `root` is the only user we've dealt with for logging in to the XenServer host. We have used the same user to interconnect XenCenter with the XenServer host. `root` is known as the **Local Super User (LSU)** and has all the rights and permissions. In XenServer, the local super user is the default account at installation. The LSU is authenticated by XenServer and not by an external authentication service. This means that if the external authentication service fails, the LSU can still log in and manage the system. The LSU can always access the XenServer physical server through SSH.

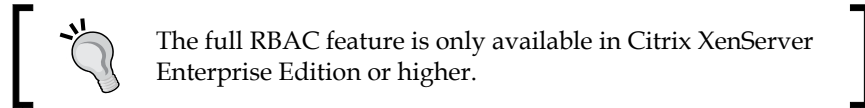
But in a work scenario where we have multiple administrators with different skills and permissions, we definitely want to have a system where we can limit users to do some specific set of tasks or even just take a look at logs. For example, the NMS team accesses XenServer to look at graphs and notifies the responsible teams to perform measures if an abnormality is found.

Role Based Access Control (RBAC) allows us to assign roles and permissions to users in order to control who has access to our XenServer pool and what actions they can perform. Using RBAC, we can map a user or a group of users that can access your XenServer pool to defined roles. Each role has some associated permissions that give to the user or the group of users the ability to perform certain operations.

RBAC allows us to easily restrict which operations different groups of users can perform, thus reducing the probability of an accident by an inexperienced user. Note that we must enable Active Directory authentication in order to take advantage of RBAC. So you must join the pool to the domain and add the Active Directory accounts before you can assign roles.

All editions of XenServer can add user accounts from Active Directory. However, only the XenServer Enterprise and Platinum editions let you assign these Active Directory accounts' different levels of permissions (through the RBAC feature).

If we are not using Active Directory with our XenServer, we're limited to using LSU only.



Roles

There are six predefined roles in XenServer, as follows:

- **Pool Administrator:** This role is equivalent to root user and can perform all operations.
- **Pool Operator:** This role can do everything apart from adding/removing users and modifying their roles. This role is focused mainly on host and pool management (that is, creating storage, making pools, managing the hosts, and so on).
- **Virtual Machine Power Administrator:** This role creates and manages virtual machines. This role is focused on provisioning VMs for use by a VM operator.
- **Virtual Machine Administrator:** Similar to the previous role, but cannot migrate VMs or perform snapshots.
- **Virtual Machine Operator:** Same as VM Admin, but cannot create/destroy VMs. They can, however, perform start/stop lifecycle operations.
- **Read-only:** This role lets you view only resource-pool and performance data.

To get in-depth details of various roles, their definitions, and definition of their permissions, go through the Citrix documentation at http://docs.vmd.citrix.com/XenServer/5.6.0/1.0/en_gb/reference.html#rbac.

Summary

In this chapter we've covered some advanced topics, such as converting a physical Windows server to a virtual machine, High Availability, and failure planning. We've also looked at Workload Balancing for more detailed reporting. Finally, we've covered the advanced role-based user permissions and the administration-control feature.

Index

A

Active/Active bonds 98
Active/Passive bond 98
Advanced Options page
importing 81

B

BIOS
virtualization, enabling 14
bonded network type 92

C

Challenge-Handshake Authentication
Protocol. *See* CHAP
CHAP 39
CIFS 49
Citrix XenServer
benefits 9
Compatible Time Sharing System.
See CTSS
control domain (dom0) 37
Convert to Template 63
cross-server private network type 91, 92
CTSS 8

D

DAS 38
DDK 16
DHCP 83
Direct attached storage. *See* DAS

Distributed Virtual Switch. *See* DVS
Distributed vSwitch. *See* DVS
Driver Development Kit. *See* DDK
DVS 100, 108
Dynamic Host Configuration Protocol.
See DHCP
dynamic memory control 76

E

End User License Agreement. *See* EULA
EULA 16
Extended Migration 35
external network type 92

F

fabric 40
fast cloning 43
FC
about 40
arbitrated loop 40
Point-to-Point 40
switched fabric 40
FCoE
creating 47
Fibre Channel. *See* FC
Fibre Channel over Ethernet. *See* FCoE
FlexMigration 35
FQDN 18
free license
activating 26-28
fully qualified domain name. *See* FQDN

G

guests 6

H

hardware HBA storage repository

creating 47

Heartbeat Storage Repository 107

heterogeneous pool 34, 35

high availability

about 106, 107

failure planning 107

host 6

host XenServer

selecting, for VM 56

hypervisor

about 7

Type-1 hypervisor 7

Type-2 hypervisor 8, 9

I

initiator 39

installing

virtualization, enabling in BIOS 14

XenCenter 22

XenServer 13-21

Internet Small Computer System Interface.

See iSCSI

IQNs 45

iSCSI 39

iSCSI Qualified Names. *See* IQNs

ISO library

about 49

connecting, options 49

creating 49-51

storage repositories, managing 51

L

LACP 98

License Manager window 28

Link Aggregation Control Protocol.

See LACP

Local Super User. *See* LSU

Logical Volume Manager. *See* LVM

LSU 110

LVM 41

M

Manage vApps 86

Maximum Transmission Unit. *See* MTU

memory

managing 74, 75

MTU 94

N

NAS 38

Network-0 option 68

network connectivity 58, 59

Network file system. *See* NFS

networking pool 90

network interfaces

about 58

managing 79, 80

Network Interfacing Cards. *See* NIC

network types

about 90

NIC bond 97

networking pool 90

routers 100

virtual networks, using 92-94

virtual switches 100

VLANs, with virtual machines 95-97

New Storage Repository page 43

new templates

creating 62

creating, from snapshots 65, 66

existing VM, converting to 62-65

NFS 39, 49

NFS VHD storage repository

creating 44

nhtol 107

NIC 89

NIC bond

about 97, 98

creating 98, 99

deleting 99

non-free license
activating 28, 29
number of host failures to tolerate.
See *nhtol*

O

Open Virtualization Format. See *OVF*
operating-system-bootable ISO
selecting 55
Organizationally Unique Identifier.
See *OUI*
OUI 40
OVF 80

P

P2V 101
PBDs 41
Performance tab 33
Physical Block Devices. See *PBDs*
Physical-to-Virtual. See *P2V*
Pool Administrator, roles 111
Pool Operator, roles 111

Q

QoS 42
Quality of Service. See *QoS*
Quick Create 64

R

RBAC 110
Read-only, roles 111
recommendation 109
resizable attribute 43
resource pool
about 34
heterogeneous 34
homogeneous 34
Resources pane 51
Role Based Access Control. See *RBAC*
roles, XenServer
Pool Administrator 111
Pool Operator 111
Read-only 111

Virtual Machine Administrator 111
Virtual Machine Operator 111
Virtual Machine Power Administrator 111

S

SAN 38
SAS 38
Serially attached SCSI. See *SAS*
single-server private network type 91
snapshots, VM
about 84
types, disk and memory 85
types, disk-only 85
types, quiesced 85
software iSCSI Storage Repository
creating 45, 46
sparse allocation attribute 43
storage area network. See *SAN*
storage attributes
about 42
fast cloning 43
resizable 43
sparse allocation 43
StorageLink repository
creating 48
storage objects, XenServer
about 40
PBDs 41
Storage repository (SR) 40
VBDs 42
Virtual Disk Images (VDIs) 41
storage protocols
about 39
FC 40
iSCSI 39
NFS 39
storage repository
about 40
creating 43
hardware HBA , storage repository
creating 47
NFS VHD storage repository,
creating 44, 45
software iSCSI storage repository,
creating 45, 46

StorageLink repository, creating 48, 49
storage space
selecting, for vHDD creation 57, 58
sysprep 70

T

target 39
templates
importing 66-68
Type-1 hypervisor 7
Type-2 hypervisor 8

V

vApps 85-88
VBDs 42
VHD (Virtual Hard Disk) 41
Virtual Block Devices. *See* VBDs
virtualization
about 6
importance 5
types 6, 7, 8
types, guests 6
types, host 6
Virtual Machine. *See* VM
Virtual Machine Administrator, roles 111
virtual machine manager 7
Virtual Machine Operator, roles 111
Virtual Machine Power Administrator, roles 111
virtual machine storage
managing 77, 78
virtual networks
using 92
VM
about 53
creating, methods 53
exporting 80, 81
host XenServer, selecting for 56
importing 80-84
network connectivity 58, 59
network interface 58, 59
operating-system-bootable ISO,
selecting 55

snapshots 84
storage space, selecting for vHDD
creation 57, 58
template, selecting for 54
VM cloning
about 68
options 69
Windows VMs cloning, VSS used 70
vm-copy CLI command 69
VM CPU
managing 74, 75
VMDK
importing 82
Volume Shadow Copy Service. *See* VSS
VSS 85

W

Windows VMs
cloning, VSS used 70
WLB
about 108
activities 109
Workload Balancing. *See* WLB
Workspace field 105
World Wide Name. *See* WWN
WWN 40

X

Xen
history 9
XenCenter
about 10
connecting, with XenServer 23
free license, activating 26-28
installing 22
installing, way 22
non-free license, activating 28, 29
system requirements 11
XenConvert
about 101
host machine, preparing 102-106
physical machine, converting to
virtual machine 102
system requirements 102

XenServer

- about 13
- connecting, with XenCenter 23
- downloading 13
- editions 25
- editions, features 25
- Enterprise and Platinum editions, features 26
- free license, activating 26, 27, 28
- history 9
- installing 13-21
- licensing 25, 26
- networking concept 89
- non-free editions, features 26
- non-free license, activating 28, 29
- requirements 10
- roles 111
- storage, managing 73
- storage objects 40
- system requirements 10
- tools 60
- working 9

XenServer host

- managing 30-33

XenServer host management

- about 30
- Console tab 31-33
- General tab 30
- NICs tab 31
- Performance tab 33
- Storage tab 30
- Users tab 34

XenServer Storage

- architectures 38
- DAS 38
- NAS 38
- SAN 38

XenServer tools 60, 61

- xsconsole 21**



Thank you for buying Implementing Citrix XenServer Quickstarter

About Packt Publishing

Packt, pronounced 'packed', published its first book "Mastering phpMyAdmin for Effective MySQL Management" in April 2004 and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern, yet unique publishing company, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike. For more information, please visit our website: www.packtpub.com.

About Packt Enterprise

In 2010, Packt launched two new brands, Packt Enterprise and Packt Open Source, in order to continue its focus on specialization. This book is part of the Packt Enterprise brand, home to books published on enterprise software – software created by major vendors, including (but not limited to) IBM, Microsoft and Oracle, often for use in other corporations. Its titles will offer information relevant to a range of users of this software, including administrators, developers, architects, and end users.

Writing for Packt

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to author@packtpub.com. If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.

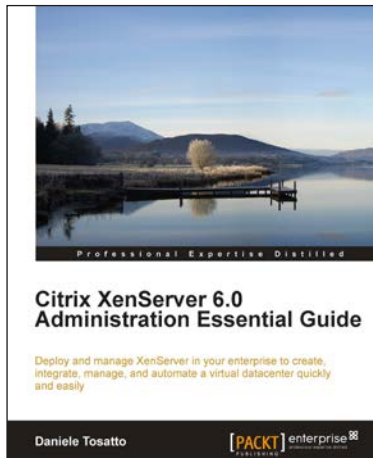


Getting Started with Citrix XenApp 6.5

ISBN: 978-1-849686-66-2 Paperback: 478 pages

Design and implement Citrix farms based on XenApp 6.5

1. Use Citrix management tools to publish applications and resources on client devices with this book and eBook
2. Deploy and optimize XenApp 6.5 on Citrix XenServer, VMware ESX, and Microsoft Hyper-V virtual machines and physical servers
3. Clear, easy-to-follow steps and screenshots to carry out each task



Citrix XenServer 6.0 Administration Essential Guide

ISBN: 978-1-847195-60-9 Paperback: 364 pages

Deploy and manage XenServer in your enterprise to create, integrate, manage, and automate a virtual datacenter quickly and easily

1. This book and eBook will take you through deploying XenServer in your enterprise, and teach you how to create and maintain your datacenter
2. Manage XenServer and virtual machines using Citrix management tools and the command line
3. Organize secure access to your infrastructure using role-based access control

Please check www.PacktPub.com for information on our titles



Citrix XenDesktop 5.6 Cookbook

ISBN: 978-1-849685-04-7 Paperback: 354 pages

Implement a fully featured XenDesktop 5.6 architecture in

a rich and powerful VDI experience configuration

1. Real-world methodologies and functioning explanations about the XenDesktop 5.6 architecture and its satellite components used to perform a service-oriented architecture
2. Learn how to publish desktops and applications to end user devices, optimizing their performance and increasing the general security
3. Step-by-step guide on how to install and configure the XenDesktop 5.6 architecture to access and use the published virtual resources



Instant Citrix XenDesktop 5 Starter

ISBN: 978-1-782170-02-0 Paperback: 66 pages

Your step-by-step guide to building a full-fledged XenDesktop infrastructure from scratch

1. Learn something new in an Instant! A short, fast, focused guide delivering immediate results.
2. Know how to install XenDesktop, integrate it with PVS and build streamed and pooled desktops
3. Learn how to build provisioning servers, capture VHD files, and configure streaming

Please check www.PacktPub.com for information on our titles