Book Collection

Learning Path

The Complete VMware vSphere Guide

Design a virtualized data center with VMware vSphere 6.7

Mike Brown, Hersey Cartwright,
Martin Gavanda, Andrea Mauro,
Karel Novak, and Paolo Valsecchi

Packt
www.packt.com

The Complete VMware vSphere Guide

Design a virtualized data center with VMware vSphere 6.7

Mike Brown
Hersey Cartwright
Martin Gavanda
Andrea Mauro
Karel Novak
Paolo Valsecchi

Packt>

BIRMINGHAM - MUMBAI

The Complete VMware vSphere Guide

First published: November 2019

Production reference: 1281119

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

ISBN 978-1-83898-575-2

www.packtpub.com

mapt.io

Mapt is an online digital library that gives you full access to over 5,000 books and videos, as well as industry leading tools to help you plan your personal development and advance your career. For more information, please visit our website.

Why subscribe?

- Spend less time learning and more time coding with practical eBooks and Videos from over 4,000 industry professionals
- Improve your learning with Skill Plans built especially for you
- Get a free eBook or video every month
- Mapt is fully searchable
- Copy and paste, print, and bookmark content

Packt.com

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.packt.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at customercare@packtpub.com for more details.

At www.packt.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.

Contributors

About the authors

Mike Brown is an army veteran and full-stack data center engineer with over 10 years in IT, with the cage nut scars to prove it. Mike has held many positions in IT, from helpdesk to systems administrator to engineer and consultant. His technical achievements include VCIX6-DCV and other VMware, Cisco, NetApp, and Microsoft certifications.

Hersey Cartwright has worked in the technology industry since 1996 in many roles, from help desk support to IT management. He first started working with VMware technologies in 2006. He is currently a Solutions Engineer for VMware, where he designs, sells, and supports VMware software-defined datacenter products in enterprise environments within the healthcare industry. He has experience working with a wide variety of server, storage, and network platforms.

Martin Gavanda has more than 10 years of experience, mainly for service providers offering IaaS solutions based on VMware vSphere products, responsible for the design and implementation of the IaaS solution in the CE region. Currently, he is working as an independent cloud architect, focusing on large infrastructure projects and practicing as a VMware instructor. He has created several virtual classes focusing on the VMware vSphere platform, with thousands of students subscribed, and he runs his own blog about virtualization and the cloud.

Andrea Mauro has more than 20 years of industrial and academic experience in IT. He works as a solutions architect and is responsible for infrastructure implementation, architecture design, upgrades, and migration processes. He is a virtualization and storage architect, specializing in VMware, Microsoft, Citrix, and Linux solutions. His first virtualized solution in production was built around ESX 2.x, several years ago. His professional certifications include not only several VMware certifications but also other vendor-related certifications. He is also a VMware vExpert, Nutanix NTC, Veeam Vanguard, and was a Microsoft MVP.

Karel Novak has 18 years of experience in the IT world. He currently works as a senior virtual infrastructure engineer at Arrow ECS Czechia and is responsible for implementation, design, and complete consultation when it comes to VMware and Veeam. As an instructor of advanced VMware and Veeam, he has delivered many courses. He specializes in VMware DCV, NSX, and, of course, Veeam. He is a VMware vExpert, VMware vExpert NSX, and a Veeam Vanguard. His highest certifications are VCI-Level 2, VCIX6-NV, VCIX6-DCV, VMCT-Mentor, and VMCA. He is also a VMware certification subject matter expert.

Paolo Valsecchi has worked in the IT industry for more than 20 years and currently works as a system engineer, mainly focused on VMware vSphere, Microsoft technologies, and backup/DR solutions. His current role involves covering all tasks related to ensuring IT infrastructure availability and data integrity (including implementation, upgrades, and administration). He holds the VMware VCP65-DCV and Veeam VMCE professional certifications and has been awarded the VMware vExpert title and the Veeam Vanguard title.

Packt is searching for authors like you

If you're interested in becoming an author for Packt, please visit `authors.packtpub.com` and apply today. We have worked with thousands of developers and tech professionals, just like you, to help them share their insight with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Table of Contents

Preface

vSphere 6.7 is the latest release of VMware's industry-leading virtual cloud platform. By understanding how to manage, secure, and scale apps with vSphere 6.7, you can easily run even the most demanding of workloads.

This Learning Path begins with an overview of the features of the vSphere 6.7 suite. You'll learn how to plan and design a virtual infrastructure. You'll also gain insights into best practices to efficiently configure, manage, and secure apps. Next, you'll pick up on how to enhance your infrastructure with high-performance storage access, such as remote direct memory access (RDMA) and persistent memory. The book will even guide you in securing your network with security features, such as encrypted vMotion and VM-level encryption. Finally, by learning how to apply Proactive High Availability and Predictive Distributed Resource Scheduler (DRS), you'll be able to achieve enhanced computing, storage, network, and management capabilities for your virtual data center.
By the end of this Learning Path, you'll be able to build your own VMware vSphere lab that can run high workloads.
This Learning Path includes content from the following Packt products:

- VMware vSphere 6.7 Data Center Design Cookbook - Third Edition by Mike Brown and Hersey Cartwright
- Mastering VMware vSphere 6.7 - Second Edition by Martin Gavanda, Andrea Mauro, Karel Novak, and Paolo Valsecchi

Who this book is for

This Learning Path is for administrators, infrastructure engineers, consultants, and architects who want to design virtualized data center environments using VMware vSphere 6.x (or previous versions of vSphere and the supporting components). Basic knowledge of VMware vSphere is required to get the most out of this Learning Path.

What this book covers

Chapter 1, *The Virtual Data Center*, provides an introduction to the benefits of the virtual data center, VMware vSphere products, and the basic virtualization concepts. This chapter identifies the differences between a data center administrator and a data center architect. An overview of the VMware Certified Advanced Professional Datacenter Design (VCAP-DCD) and VMware Certified Design Architect (VCDX) certifications are also covered.

Chapter 2, *The Discovery Process*, explains how to identify stakeholders, conduct stakeholder interviews, and perform technical assessments to discover the business and technical goals of a virtualization project. This chapter covers how to use the following tools—VMware Capacity Planner, Windows Performance Monitor, and vRealize Operations Manager—to collect resource information during the discovery process.

Chapter 3, *The Design Factors*, explains how to identify and document the design requirements, constraints, assumptions, and risks. This chapter details how to use the design factors to create a conceptual design.

Chapter 4, vSphere Management Design, describes the vCenter Server components and their dependencies. Recipes for determining which vCenter Server deployment options to use, the Windows server or virtual appliance to be used, and for determining the type of database to use based on the deployment size, are included.

Chapter 5, *vSphere Storage Design*, covers logical storage design. Recipes are included for calculating the storage capacity and performance requirements for the logical storage design. This chapter covers the details of selecting the correct RAID level and storage connectivity to support design. Recipes for VSAN and VVOLs are provided in this chapter.

Chapter 6, *vSphere Network Design*, provides details on the logical network design. This chapter explains how to calculate bandwidth requirements to support a vSphere design. Details on selecting a virtual switch topology, designing for network availability, and the network requirements to support vMotion and IP connected storage, are also covered.

Chapter 7, vSphere Compute Design, provides recipes for calculating the CPU and memory requirements to create a logical compute design. The chapter also covers cluster design considerations for **High Availability** (**HA**) and the **Distributed Resource Scheduler** (**DRS**).

Chapter 8, *vSphere Physical Design*, explains how to satisfy design factors by mapping the logical management, storage, network, and compute designs to hardware to create a physical vSphere design. The chapter also provides details on creating a custom installation ISO to install ESXi and the best practices for host BIOS configurations.

Chapter 9, *Virtual Machine Design*, looks at the design of virtual machines and application workloads running in the virtual data center. Recipes are provided for right-sizing virtual machine resources, enabling the ability to add virtual machine resources, and creating virtual machine templates. This chapter details the use of affinity and anti-affinity rules to improve application efficiency and availability. Converting or migrating physical servers to virtual machines is also covered in this chapter.

Chapter 10, Deployment Workflow and Component Installation, starts by explaining the components of vSphere and the roles and services they provide. We will walk through the main aspects to consider in terms of the preparation of a deployment plan for your environment, analyzing the criteria for hardware platform selection, storage, and network requirements.

Chapter 11, *Configuring and Managing vSphere 6.7*, describes the different ways to manage a vSphere 6.7 infrastructure, including the new HTML5 clients, and also contains an introduction to the scripting and automation tools. ESXi, vCenter, VMware cluster-related configuration, and management topics are covered

Chapter 12, *Life Cycle Management, Patching, and Upgrading*, looks at how, with vSphere 6.7, administrators will find significantly more powerful capabilities for patching, upgrading, and managing the configuration of the virtual environment using the Update Manager and Host Profile features. We also cover the upgrade path and considerations to make regarding upgrading or migrating your virtual environment.

Chapter 13, VM Deployment and Management, introduces the practices and procedures involved in deploying, configuring, and managing Virtual Machines (VMs) in a vSphere infrastructure. Different types of VM provisioning are considered, including the use of templates, the content library, and OVF.

Chapter 14, *VM Resource Management*, provides a comprehensive view of vSphere resources management, including reservations, limits, and shares, and how to balance and optimize them in your environment. Finally, we will discuss different migration techniques for moving your workload across different environments.

Chapter 15, *Availability and Disaster Recovery*, focuses on specific availability (and resiliency) solutions in vSphere, including the new vSphere High Availability (HA) features, proactive HA, vSphere Fault Tolerance (FT), and other solutions, such as guest clustering.

Chapter 16, *Securing and Protecting Your Environment*, looks at how security has become a critical part of any implementation, including virtual environments. In addition to the security and hardening aspects of vSphere, the new 6.7 version brings other important related features (though some were introduced with version 6.5), such as VM encryption, encrypted vMotion, secure boot support for VMs, and secure boot plus cryptographic hypervisor assurance for ESXi.

Chapter 17, *Analyzing and Optimizing Your Environment*, covers the native tools used to monitor your environment for performance analysis or for possible issues in order to improve the virtual environment and workloads. This chapter focuses on monitoring different critical resources, such as computing, storage, and networking resources, across ESXi hosts, resource pools, and clusters. Other tools, such as vRealize Operations and third-party tools, will also be described briefly.

Chapter 18, *Troubleshooting Your Environment*, covers the native tools used to troubleshoot performance issues and other issues in a vSphere environment. Also, the chapter provides some examples and methods for troubleshooting approaches.

Chapter 19, B*uilding Your Own VMware vSphere Lab*, goes into the basics of why you should build your own lab environment, looking at what the benefits of running such a lab are in comparison with using VMware Hands-On Labs (HOLs). Different approaches to how labs can be designed will be covered.

To get the most out of this book

This book assumes a basic level of VMware vSphere and virtualization knowledge, which you will need in order to understand all the concepts.

This book requires the following minimum software components: VMware vSphere 6.7, and VMware vCenter Server 6.7. There is also other optional software.

The best way to practice without the need for software licenses or hardware components is to try VMware HOLs (`https://labs.hol.vmware.com/`), which cover different products and technologies. The first ones that you should use if you are new to the features of vSphere 6.7 are listed here:

- HOL-1911-01-SDC – What's New in VMware vSphere 6.7
- HOL-1911-91-SDC – vSphere 6.7 Lightning Lab
- HOL-1904-02-CHG – vSphere 6.7 – Challenge Lab

If you would prefer your own lab, there are several suggestions for what type of hardware to use, whether a single big server with nested ESXi hypervisors or a cloud service such as Ravello (which can also host nested ESXi hosts). There are also suggestions on how to deploy all software components. One interesting way of doing so is using AutoLab (`http://www.labguides.com/autolab/`), or you can see the blogs of Alan Renouf and William Lam, where you can find some powerful scripts for building an entire vSphere 6.5 environment (also with vSAN and NSX!).

Download the example code files

You can download the example code files for this book from your account at `www.packt.com`. If you purchased this book elsewhere, you can visit `www.packt.com/support` and register to have the files emailed directly to you.

You can download the code files by following these steps:

1. Log in or register at `www.packt.com`.
2. Select the **SUPPORT** tab.
3. Click on **Code Downloads & Errata**.
4. Enter the name of the book in the **Search** box and follow the onscreen instructions.

Once the file is downloaded, please make sure that you unzip or extract the folder using the latest version of:

- WinRAR/7-Zip for Windows
- Zipeg/iZip/UnRarX for Mac
- 7-Zip/PeaZip for Linux

We also have code bundles from our rich catalog of books and videos available at `https://github.com/PacktPublishing/`. Check them out!

Conventions used

There are a number of text conventions used throughout this book.

`CodeInText`: Indicates code words in the text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles. Here is an example: "At the `runweasel` command line, type `ks=usb:/ks.cfg`."

A block of code is set as follows:

```
vmaccepteula
rootpw mypassword
install --firstdisk –overwritevmfs
keyboard English
network --bootproto=dhcp --device=vmnic0
reboot
```

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

```
esxcli system syslog config set –loghost tcp://SYSLOG_IP:514
esxcli system syslog reload
```

Any command-line input or output is written as follows:

```
cd /usr/lib/vmware-sso/
openssl x509 -inform PEM -in xyzCompanySmartCardSigningCA.cer >>
/usr/lib/vmware-sso/vmware-sts/conf/clienttrustCA.pem
```

Bold: Indicates a new term, an important word, or words that you see onscreen. For example, words in menus or dialog boxes appear in the text like this. Here is an example: "Under **Settings**, switch to **General** and click the **Edit...** button."

Warnings or important notes appear like this.

Tips and tricks appear like this.

Get in touch

Feedback from our readers is always welcome.

General feedback: If you have questions about any aspect of this book, mention the book title in the subject of your message and email us at `customercare@packtpub.com`.

Errata: Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you have found a mistake in this book, we would be grateful if you would report this to us. Please visit `www.packt.com/submit-errata`, selecting your book, clicking on the Errata Submission Form link, and entering the details.

Piracy: If you come across any illegal copies of our works in any form on the Internet, we would be grateful if you would provide us with the location address or website name. Please contact us at `copyright@packt.com` with a link to the material.

If you are interested in becoming an author: If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, please visit `authors.packtpub.com`.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions, we at Packt can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about Packt, please visit `packt.com`.

1
The Virtual Data Center

This chapter focuses on many of the basic concepts and benefits of virtualization. It provides a quick overview of VMware virtualization, introduces the virtual data center architect, and lays some of the groundwork necessary for creating and implementing a successful virtual data center design using VMware vSphere 6.7.

We will also explore the **VMware Certified Advanced Professional 6-Data Center Virtualization Design** (**VCAP6-DCV Design**) exam and the new **VMware Certified Design Expert** (**VCDX**) certification, including a few tips that should help you prepare to successfully complete the exam and certification. Then, we will look over some of the new features of vSphere 6.7. This section will include where to find the current release notes and the latest vSphere product documentation. Finally, we will take a high-level look at the process for planning an upgrade to an existing vSphere deployment to vSphere 6.7.

In this chapter, we will cover the following recipes:

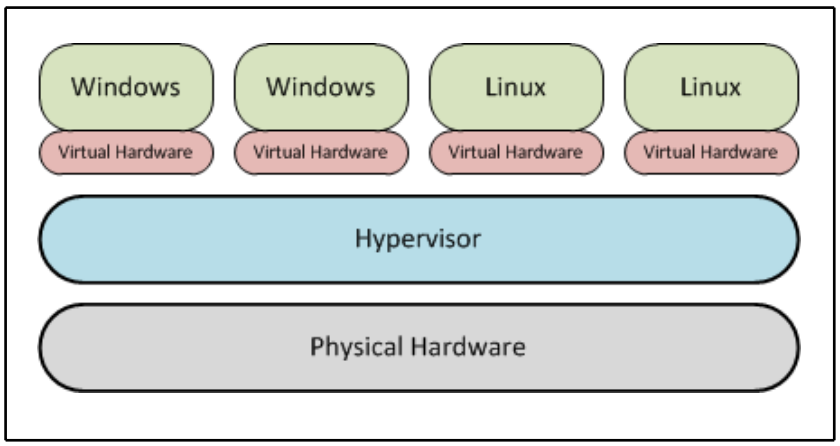
- Becoming a virtual data center architect
- Using a holistic approach to data center design
- Passing the VMware VCAP6-DCV Design exam
- Becoming a VMware Certified Design Expert
- Identifying what's new in vSphere 6.7
- Planning a vSphere 6.7 upgrade

Benefits and technologies of virtualization

If you are already familiar with virtualization, this chapter will provide a review of many of the benefits and technologies of virtualization.

Since the focus of this book is on design, we will not go into great detail discussing the specifics of how to configure resources in a virtual data center. Most of you probably already have a good understanding of VMware's virtualization architecture, so this chapter will just provide a basic overview of the key VMware components that are the building blocks to the virtual data center.

Virtualization creates a layer of abstraction between the physical hardware and the virtual machines that run on it. Virtual hardware is presented to the virtual machine granting access to the underlying physical hardware, which is scheduled by the hypervisor's kernel. The hypervisor separates the physical hardware from the virtual machine, as shown in the following diagram:



Logical representation of hypervisor layer

The hypervisor separates the physical hardware from the virtual machines. The new release of vSphere 6.7 does not change the design process or the design methodologies. The new functions and features of the release provide an architect with more tools to satisfy design requirements.

The hypervisor

At the core of any virtualization platform is the hypervisor. The VMware hypervisor is named **vSphere ESXi**, simply referred to as **ESXi**. ESXi is a Type 1 or bare-metal hypervisor. This means that it runs directly on the host's hardware to present virtual hardware to the virtual machines. In turn, the hypervisor schedules access to the physical hardware of the hosts.

ESXi allows multiple virtual machines with a variety of operating systems to run simultaneously, sharing the resources of the underlying physical hardware. Access to physical resources, such as memory, CPU, storage, and network, used by the virtual machines is managed by the scheduler, or **Virtual Machine Monitor** (**VMM**), provided by ESXi. The resources presented to the virtual machines can be over committed; this means more resources that are physically available can be allocated to the virtual machines on the physical hardware. Advanced memory sharing and reclamation techniques, such as **Transparent Page Sharing** (**TPS**) and ballooning, along with CPU scheduling, allow for over commitment of these resources to be possible, resulting in greater virtual-to-physical consolidation ratios.

ESXi 6.7 is a 64-bit hypervisor that must be run on a 64-bit hardware. An ESXi 6.7 installation requires at least 1 GB of disk space for installation. It can be installed on a hard disk locally, a USB device, a **Logical Unit Number** (**LUN**) on a **Storage Area Network** (**SAN**), or deployed stateless on hosts with no storage using Auto Deploy. The small footprint of an ESXi installation provides a reduction in the management overhead associated with patching and security hardening.

With the release of vSphere 5.0, VMware retired the ESX hypervisor. ESX had a separate, Linux-based service console for the management interface of the hypervisor. Management functions were provided by agents running in the service console. The service console has since been removed from ESXi, and agents now run directly on ESXi's VMkernel.

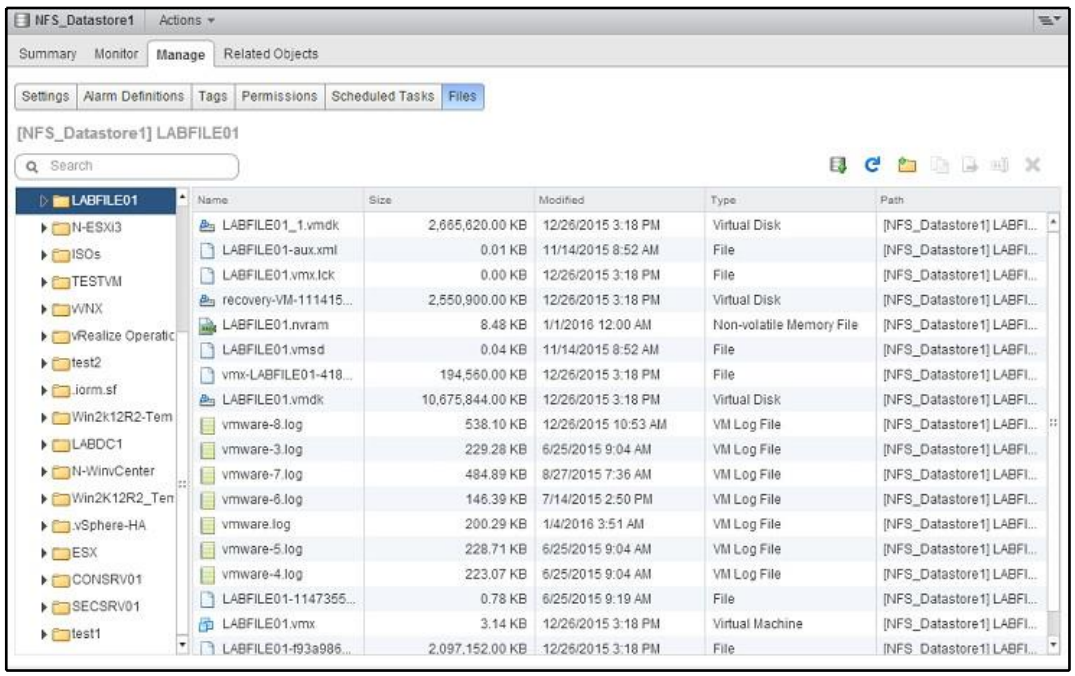
To manage a standalone host running ESXi, a **Direct Console User Interface** (**DCUI**) is provided for basic configuration and troubleshooting. A shell is available that can either be accessed locally from the console or remotely using **Secure Shell** (**SSH**). The `esxcli` command-line tools and others can be used in the shell to provide advanced configuration options. An ESXi host can also be accessed directly using the vSphere Client. The ESXi DCUI is shown in the following screenshot:

Screenshot of ESXi's DCUI

The DCUI can be accessed remotely using SSH by typing the `dcui` command in the prompt. Press *Ctrl* + *C* to exit the remote DCUI session.

Virtual machines

A virtual machine is a software computer that runs a guest operating system. Virtual machines are comprised of a set of configuration files and data files stored on local or remote storage. These configuration files contain information about the virtual hardware presented to the virtual machine. This virtual hardware includes the CPU, RAM, disk controllers, removable devices, and so on, and emulates the same functionality as the physical hardware. The following screenshot depicts the virtual machine files that are stored on a shared **Network File System** (**NFS**) datastore:



Virtual machine files stored on a shared NFS datastore displayed using the vSphere Web Client

The files that make up a virtual machine are typically stored in a directory set aside for the particular virtual machine they represent. These files include the configuration file, virtual disk files, NVRAM file, and virtual machine log files.

The following table lists the common virtual machine file extensions along with a description of each:

File extension	Description
`.vmx`	This is a virtual machine configuration file. It contains the configurations of the virtual hardware that is presented to the virtual machine.
`.vmdk`	This is a virtual disk descriptor file. It contains a header and other information pertaining to the virtual disk.
`-flat.vmdk`	This is a preallocated virtual disk. It contains the content or data on the disk used by the virtual machine.
`.nvram`	This is a file that stores the state of a virtual machine's **Basic Input Output System** (**BIOS**) or **Extensible Firmware Interface** (**EFI**) configurations.
`.vswp`	This is a virtual machine swap file. It gets created when a virtual machine is powered on. The size of this file is equal to the amount of memory allocated minus any memory reservations.
`.log`	This is a virtual machine log file.
`.vmsd`	This is a virtual machine file used with snapshots to store data about each snapshot active on a virtual machine.
`.vmsn`	This is a virtual machine snapshot data file.

Virtual machines can be deployed using a variety of methods, as follows:

- Using the New Virtual Machine Wizard in the vSphere Client or vSphere Web Client
- By getting converted from a physical machine using the VMware Converter
- By getting imported from an **Open Virtualization Format** (**OVF**) or **Open Virtualization Archive** (**OVA**)
- By getting cloned from an existing virtual machine
- By getting deployed from a virtual machine template

When a new virtual machine is created, a guest operating system can be installed on the virtual machine. VMware vSphere 6.7 supports more than 120 different guest operating systems. These include many versions of the Windows server and desktop operating systems, many distributions and versions of Linux and Unix operating systems, and Apple macOS operating systems.

Virtual appliances are preconfigured virtual machines that can be imported to the virtual environment. A virtual appliance can be comprised of a single virtual machine or a group of virtual machines with all the components required to support an application. The virtual machines in a virtual appliance are preloaded with guest operating systems, and the applications they run are normally preconfigured and optimized to run in a virtual environment.

Since virtual machines are just a collection of files on a disk, they become portable. Virtual machines can be easily moved from one location to another by simply moving or copying the associated files. Using VMware vSphere features, such as vMotion, Enhanced vMotion, or Storage vMotion, virtual machines can be migrated from host to host or datastore to datastore while a virtual machine is running. Virtual machines can also be exported to an OVF or OVA to be imported into another VMware vSphere environment.

Virtual infrastructure management

VMware vCenter Server provides a centralized management interface to manage and configure groups of ESXi hosts in the virtualized data center. The vCenter Server is required to configure and control many advanced features, such as the **Distributed Resource Scheduler** (**DRS**), Storage DRS, and VMware **High Availability** (**HA**). The vCenter Server management **Graphical User Interface** (**GUI**) is accessed using the browser-based vSphere Client. Many vendors provide plugins that can be installed to allow third-party storage, network, and compute resources to be managed using the vSphere Client.

vCenter access using the C#, or Windows vSphere Client, is only available in versions prior to 6.5. Since the release of vSphere 5.5, however, access to, and the configuration of, new features is only available using the vSphere Web Client. The vSphere Web Client can be accessed at `https://FQDN_or_IP_of_vCenter_Server:9443/`.

vCenter Server 6.7 must use a 64-bit architecture if installed on a Windows Server. It can be run on dedicated physical hardware or as a virtual machine. When the vCenter Server is deployed on Windows, it requires either the embedded PostgreSQL database, a Microsoft SQL database, or an Oracle database to store configuration and performance information. IBM DB2 databases are supported with vSphere 5.1, but this support was removed in vSphere 5.5.

With the release of vCenter 6.0, the Microsoft SQL Express database is no longer used as the embedded database. Embedded PostgreSQL is now used as the embedded database for small deployments. The PostgreSQL database on a Windows Server can be used to support environments of less than 20 hosts and 200 virtual machines. When upgrading to vCenter 6.7, if the previous version was using the Microsoft SQL Express database, the database will be converted to the embedded PostgreSQL as part of the upgrade. The embedded PostgreSQL database is suitable for almost all deployments, but using an external database is still supported.

Another option for deploying the vCenter Server is the **vCenter Server Appliance** (**VCSA**). The VCSA is a preconfigured, Linux-based virtual machine preinstalled with the vCenter Server components. The appliance includes an embedded PostgreSQL database that supports the configuration maximums of 2,000 hosts and 25,000 powered-on virtual machines.

Several other management and automation tools are available to aid the day-to-day administration of a vSphere environment: the **vSphere Command-Line Interface** (**vCLI**); vSphere PowerCLI provides a Windows PowerShell interface; vRealize Orchestrator can be used to automate tasks; and the **vSphere Management Assistant** (**vMA**) is a Linux-based virtual appliance that is used to run management and automation scripts against hosts. vMA was deprecated, and its final release only supports vSphere 6.5. These tools allow an administrator to use command-line utilities to manage hosts from remote workstations.

VMware provides a suite of other products that benefit the virtualized data center. These data center products, such as **VMware vRealize Operations** (**vROps**), **VMware Site Recovery Manager** (**SRM**), and **VMware vRealize Automation** (**vRA**), can each be leveraged in the virtual data center to meet specific requirements related to management, disaster recovery, and cloud services. At the core of these products is the vSphere suite, which includes ESXi, the vCenter Server, and the core supporting components.

Understanding the benefits of virtualization

The following table provides a matrix of some of the core VMware technologies and the benefits that can be realized by using them:

VMware technology	Primary benefits	Description
vSphere ESXi	Server consolidation Resource efficiency	ESXi is VMware's bare-metal hypervisor that hosts virtual machines, also known as guests, and schedules virtual hardware access to physical resources.
vSphere HA	Increased availability	HA restarts virtual machines in the event of a host failure. It also monitors and restarts the virtual machines in the event of a guest operating system failure.

vMotion and vSphere DRS	Resource efficiency Increased availability	vMotion allows virtual machines to be live-migrated between hosts in a virtual data center. DRS determines the initial placement of the virtual machine on the host resources within a cluster and makes recommendations, or automatically migrates the virtual machines to balance resources across all hosts in a cluster.
Resource pools	Resource efficiency	These are used to guarantee, reserve, or limit the virtual machine's CPU, memory, and disk resources.
VMware **Fault Tolerance** (**FT**)	Increased availability	FT provides 100 percent uptime for a virtual machine in the event of a host hardware failure. It creates a secondary virtual machine that mirrors all the operations of the primary. In the event of a hardware failure, the secondary virtual machine becomes the primary and a new secondary is created.
Thin provisioning	Resource efficiency	This allows for storage to be over provisioned by presenting the configured space to a virtual machine, but only consuming the space on the disk that the guest actually requires.
Hot add CPU and memory	Resource efficiency scalability	This allows for the addition of CPU and memory resources to a virtual machine while the virtual machine is running.
Storage vMotion	Resource efficiency	This moves virtual machine configuration files and disks between storage locations that have been presented to a host.
vSphere Storage **Application Programming Interface** (**APIs**); data protection	VM backups and disaster recovery	Allows third parties to build agentless backup and disaster recovery solutions that integrate with the vSphere platform
vSphere replication	Disaster recovery	This features provides the ability to replicate virtual machines between sites.
vCenter server	Simplified management	This provides a single management interface to configure and monitor the resources available to virtual data centers.
vCenter server linked mode	Simplified management	This links multiple vCenter Servers together to allow them to be managed from a single client.
Host profiles	Simplified management	This maintains consistent configuration and configuration compliance across all the hosts in the environment.

This is not meant to be an exhaustive list of all VMware technologies and features, but it does provide an insight into many of the technologies commonly deployed in the enterprise virtual data center.

There are many others, and each technology or feature may also have its own set of requirements that must be met in order to be implemented. The purpose here is to show how features or technologies can be mapped to benefits that can then be mapped to requirements and ultimately mapped into a design. This is helpful in ensuring that the benefits and technologies that virtualization provides satisfy design requirements.

Identifying when not to virtualize

Not all applications or server workloads are good candidates for virtualization. It is important that these workloads are identified early on in the design process.

There are a number of reasons why a server or application may not be suitable for virtualization. Some of these include the following:

- Vendor support
- Licensing issues
- Specialized hardware dependencies
- High resource demand
- Lack of knowledge or skillsets

A common reason to not virtualize an application or workload is the reluctance of a vendor to support their application in a virtual environment. As virtualization has become more common in the enterprise data center, this has become uncommon; but, there are still application vendors that will not support their products once virtualized.

Software and operating system licensing in a virtual environment can also be a challenge, especially when it comes to physical server to virtual machine conversions. Many physical servers are purchased with **Original Equipment Manufacturer** (**OEM**) licenses, and these licenses, in most cases, cannot be transferred to a virtual environment. Also, many licenses are tied to hardware-specific information, such as interface MAC addresses or drive signatures. Licensing issues can usually be overcome. Many times, the primary risk becomes the cost to upgrade or acquire new licensing. As with other potential design risks, it is important that any issues and potential impacts licensing may have on the design be identified early on in the design process.

Some applications may require the use of specialized hardware. Fax boards, serial ports, and security dongles are common examples. There are ways to provide solutions for many of these, but often, given the risks associated with the ability to support the application, or the loss of one or more of the potential benefits of virtualizing the application, the better solution may be to leave the application on dedicated physical hardware. Again, it is important that these types of applications be identified very early on in the design process.

Physical servers configured with a large amount of CPU and memory resources where applications are consuming a large amount of these resources may not be good candidates for virtualization. This also holds true for applications with high network utilization and large storage I/O requirements. vSphere 6.7 supports virtual machines configured with up to 128 **virtual CPUs** (**vCPUs**) and 6 TB of memory, but the high utilization of these configured resources can have a negative impact on other workloads in the virtual environment. These high-utilization workloads will also require more resources to be reserved for failover. The benefits of virtualizing resource-intensive applications must be weighed against the impact placed on the virtual environment. In some cases, it may be better to leave these applications on dedicated physical hardware.

Many administrators may lack knowledge of the benefits or skills to manage a virtualized data center. The administrator of a virtual environment must be well-versed with storage, networking, and virtualization in order to successfully configure, maintain, and monitor a virtual environment. Though this may not necessarily be a reason not to leverage the benefits of a virtualized environment, it can be a substantial risk to the acceptance of a design and the implementation. This is especially true with smaller IT departments, where the roles of the server, application, storage, and network administrators are combined.

Becoming a virtual data center architect

The virtual data center architect, or simply the architect, is someone who identifies requirements, designs a virtualization solution to meet those requirements, and then oversees the implementation of the solution. Sounds easy enough, right?

How it works…

The primary role of the architect is to provide solutions that meet customer requirements. At times, this can be difficult, since the architect may not always be part of the complete sales process. Often, customers may purchase hardware from other vendors and look to us to help them make it all work. In such situations, the purchased hardware becomes a constraint on the design. Identifying and dealing with constraints and other design factors will be discussed in more detail in *Chapter 3, The Design Factors*.

The architect must also be able to identify requirements, both business and technical, by conducting stakeholder interviews and analyzing current configurations. Once the requirements have been identified, the architect must then map the requirements into a solution by creating a design. This design is then presented to the stakeholders, and if, approved, it is implemented. During the implementation phase, the architect ensures that configurations are done to meet the design requirements and that the work done stays within the scope of the design.

The architect must also understand best practices. Not just best practice for configuring the hypervisor, but for management, storage, security, and networking. Understanding the best practice is the key. The architect not only knows best practice but understands why it is considered best practice. It is also important to understand when to deviate from what is considered best practice.

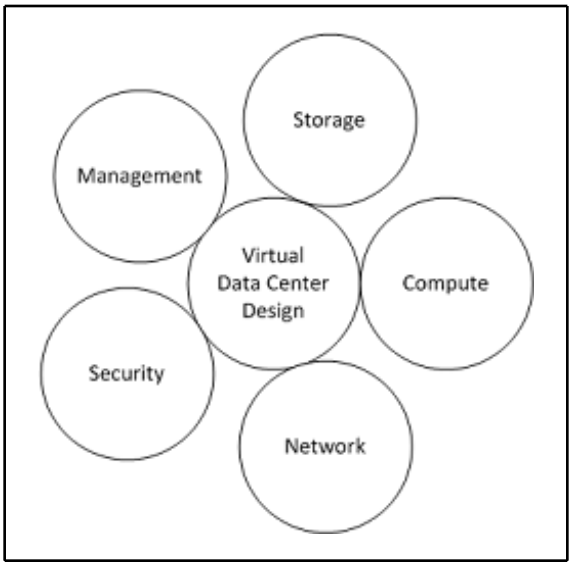
There's more…

The large part of an architect's work is facing customers. This includes conducting interviews with stakeholders to identify requirements and ultimately presenting the design to decision makers. Besides creating a solid solution to match the customer's requirements, it is important that the architect gains and maintains the trust of the project stakeholders. A professional appearance and, more importantly, a professional attitude, are both helpful in building this relationship.

Using a holistic approach to data center design

The virtual data center architect must be able to take a holistic approach to data center design. This means that for every decision made, the architect must understand how the environment as a whole will be impacted.

An architect is required to be, at the very least, familiar with all aspects of the data center. They must understand how the different components of a data center, such as storage, networking, computing, security, and management, are interconnected, as shown in the following diagram:



The holistic approach to data center design

It has become very important to understand how any decision or change will impact the rest of the design. Identifying dependencies becomes an important part of the design process. If a change is made to the network, how are computing, management, and storage resources affected? What other dependencies will this introduce in the design? Failing to take a holistic approach to design can result in unnecessary complications during the design process, and potentially costly fixes after the design is implemented.

How to do it…

The following scenario is built as an example which helps illustrate the concept of using a holistic design approach.

You have been engaged to design a virtualization solution for a financial organization. The solution you are proposing is to use 10 GB **Converged Network Adapters** (**CNA**) to provide connectivity to the organization's network in three 1U rack-mount servers. The organization needs to separate a **Virtual Local Area Network** (**VLAN**) that is currently configured to be delivered over the CNA onto a physically separate network to satisfy a new compliance requirement. A 1 GB network will provide sufficient bandwidth for this network, and the network should be highly available. Single points of failure should be minimized.

To support this compliance requirement, you, the architect, must take a holistic approach to the design and answer a number of questions about each design decision, for example:

1. Are there network ports available in the current rack-mount servers, or will a network card need to be added? If a card has to be added, are there **Peripheral Component Interconnect** (**PCI**) slots available?
2. Will a dual-port network card provide sufficient redundancy, or will the network need to be separated across physical cards? Are there onboard network ports available that can be used with a PCI network card to provide in-box redundancy?
3. Has the hardware for the physically separate switch been obtained? If not, how long before the equipment is received and deployed? Will this have an impact on the implementation schedule?
4. How will the virtual switch need to be configured to provide the connectivity and redundancy that is required?

How it works…

The impact can be fairly significant, depending on some of the answers. For example, let's say the 1U rack-mount server will not support the required network adapters needed to satisfy the requirement and a different 2U rack-mount server must be used. This then raises more questions, such as whether there is sufficient space in the rack to support the new server footprint.

What if the requirement had been that the applications connected to this network be virtualized on separate physical server hardware and storage? What parts of the design would have to change? The architect must be able to understand the dependencies of each part of the design and how a change in one place may affect other areas of the design.

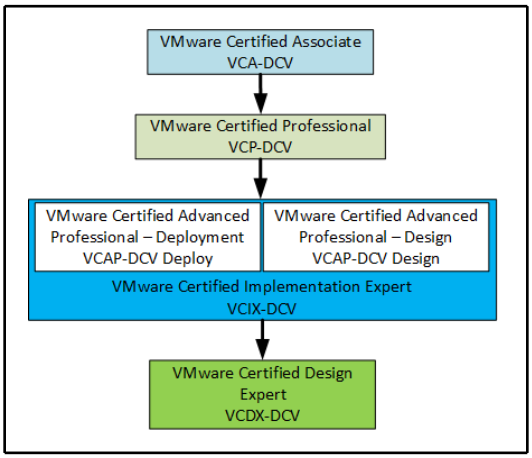
As you think through these questions, you should be able to see how a change to a requirement can have a deep impact on many other areas of the design. It becomes very important to identify requirements early on in the design process.

Passing the VMware VCAP6-DCV Design exam

VMware has **VMware Certified Advanced Professional** (**VCAP**) exams testing the ability of a person to deploy, administer, and design complex virtual environments. The exams for vSphere 6 come in two types: Design and Deployment. Passing both exams earns the designation **VMware Certified Implementation Expert** (**VCIX**). The VCIX is not a certification the same way that VCAPs are; rather it is a special designation that proves the earner has deep and wide expertise in designing and deploying complex vSphere 6 infrastructures.

VMware is constantly reviewing and updating their certification system. Recent changes to the vSphere 6.x advanced certifications included adding the VCAP6.5-**Data Center Virtualization** (**DCV**) Design exam and retiring the VCAP6-DCV Design exam. Overarching changes to the entire VMware certification program include the replacement of product versions in the certification title with the year in which the certification was earned. For example, the only current, advanced DCV Deployment exam is titled VCAP-DCV Deployment 2018. This change was made in an effort to show the timeliness of the certification simply by its name. Reading the certification page for this exam is the only way to understand which product version is tested within the exam.

The current, high-level VMware certification path is mapped out in the following flowchart:



VMware certification path for data center administrators and architects

The VCAP-DCV Design exam tests your ability to design enterprise virtualized environments. To be successful, you must have an in-depth understanding of VMware's core components and the relationship they share with other components of the data center, such as storage, networking, and application services, along with a mastery of VMware's data center design methodologies and principles. All the exam objectives, including study resources, can be found in the exam blueprint. VMware exam roadmaps and the VCAP exam blueprints can be found on the VMware Certification portal page at `https://mylearn.vmware.com/portals/certification/`.

Getting ready

Before you are eligible to take a VCAP6.5-DCV Design exam, you should have obtained the relevant **VMware Certified Professional-Data Center Virtualization** (**VCP6.5-DCV**) certification. Besides the training required for the VCP6.5-DCV certification, there is no other requisite training that must be completed in order to sit the VCAP6.5-DCV Design exam. When you are ready to schedule your VCAP6.5-DCV Design exam, you must submit an exam authorization request to VMware. When you submit the exam authorization request, VMware will verify that you have met the certification prerequisites and provide you with the access necessary to schedule the exam.

The VCAP6.5-DCV Design exam consists of 60 questions with a time limit of 135 minutes. The passing score is 300 out of 500. The exam questions are comprised of a mixture of multiple choice, matching, and drag and drop. VMware has removed the Visio-style design scenario formatted questions from this exam. Refer to the VMware Certification Portal for
details: `https://mylearn.vmware.com/portals/certification/`.

How to do it…

The VCAP-DCV Design exam for vSphere 6 was one of the most challenging exams I have ever taken. Here are a few tips to help you prepare for and successfully sit the VCAP6-DCV Design exam:

1. **Study the material on the exam blueprint**: The exam blueprint lists all the objectives of the exam, along with links to documentation related to each exam objective.
2. **Review the vSphere 6 release notes and product documentation**: The release notes and product documentation will provide an overview of the features available, the requirements that must be met to support implementation of the new features, and the best practices for implementing features to support design requirements.
3. **Schedule your exam**: Scheduling your exam sets a goal date for you to work toward. Setting the date can provide motivation to help you stay on track with your studying efforts.
4. **Watch the APAC vBrownBag DCD5 series**: The APAC vBrownBag did a series of podcasts focusing on the VCAP-DCD exam for vSphere 5 exam objectives. Even though these podcasts focus on version five of the exam, many of the design methodologies and concepts are similar. These podcasts are still relevant and provide a valuable study resource. The podcast can be found at `http://www.professionalvmware.com/brownbags`.
5. **Get familiar with the exam design interface**: On VMware's VCAP Certification page for the Design exam, there is a UI Demo that will help get you familiar with the design interface that is used on the exam.
6. **Practice time management**: It is very important that you are aware of the amount of time you are taking on a question, and how much time remains. If you get hung up on a multiple choice question, take your best guess and move on. Conserve time for the more complex drag and drop and design scenario questions.

7. **Answer every question**: A question left unanswered will be marked incorrect and will not benefit your score in any way. A guess has some chance of being correct.
8. **Study the material on the exam blueprint**: I know this has already been mentioned once, but it is worth mentioning again. The exam blueprint contains all the testable objectives. Study it!

There's more…

For up-to-date information on the VCAP-DCV Design certification, to download the exam blueprint, and to book the exam once it has been released, visit the VMware Certification Portal page at `https://mylearn.vmware.com/portals/certification/`.

The final stop on the VMware Certification path is **VMware Certified Design Expert-Data Center Virtualization** (**VCDX**). The VCDX certification requires creating a VMware vSphere design, submitting the design to VMware for review, and then defending the design before a panel of VMware design experts.

Becoming a VMware Certified Design Expert

The VCDX is the pinnacle of VMware's certifications. A VCDX certification validates an architect's ability to design, implement, test, document, present, and defend the design of complex, enterprise solutions based on VMware products. Earning the certification ultimately comes down to two things: creating a design, and defending your design in front of a panel of VCDX veterans.

Before attempting the VCDX certification, an architect usually has experience designing the same level of advanced, vSphere designs that the VCDX defense panel is looking for. Before designing such solutions, a VCDX candidate also usually has experience implementing and administering complex vSphere designs. While these experiences are not hard requirements, it is a natural progression that sets the candidate up for success and gives them the best chance of succeeding in the VCDX process. The only other prerequisites to attempt the VCDX6 is to hold either a **VMware Certified Professional 6-Data Center Virtualization** (**VCP6-DCV**) or **VCP6.5-DCV**, and earn the VCIX6-DCV or VCIX6.5-DCV badge.

This section discusses the VCDX6-DCV that is based on vSphere 6.x designs, but there are other current tracks that lead to VCDX and include the following:

- **VCDX6**: **Network Virtualization** (**VCDX6-NV**)—this certification is focused on both vSphere and NSX 6.x
- **VCDX7**: **Cloud Management and Automation** (**VCDX7-CMA**)—this certification is based on vRealize Automation 7.x
- **VCDX6**: **Desktop and Mobility** (**VCDX6-DTM**)—this certification is based on the Horizon Suite

No matter which track is chosen, understand the VCDX certification process well. VMware has published two documents for most tracks that cover this information: the blueprint and the handbook. The blueprint describes the rules of the VCDX process, including things such as what format the process uses, time limits, and the language in which the process is held. It also covers the objectives of the specific test format used and explains what the VCDX panelists are looking for in a VCDX candidate.

The handbook offers some details on how to choose a good design on which to base a VCDX defense, VMware's policy on teamwork in the VCDX process, and finally, what to expect during the live defense portion of the defense. Becoming familiar with the contents of each document will help focus a candidate's time and effort while progressing through the VCDX process.

How to do it…

After meeting the prerequisites, there are only two more steps to becoming a VCDX. The fees have changed over time, so be sure to check VMware's website for up-to-date costs. You must do the following:

- Submit a VCDX design application with an application fee of $995
- If successful, defend your design, live, in front of a panel of current VCDXs, and pay a defense fee of $3,000

Getting your VCDX application accepted, however, is a lot of work and a big hurdle to overcome. If your application is accepted, the VCDX program is telling you that the documentation, by itself, is of expert quality, and the only thing left to do is prove to them during the live defense panel that you are, indeed, an expert.

The VCDX application consists of a set of documents. Aside from the application itself, you must create a documentation bundle that could follow this order:

1. **Create the design document**: This will be the main document of your submission, where you'll likely spend the most time. This is where you'll document requirements, constraints, assumptions, and risks, and map them to the vSphere components of compute, storage, network, management, and the virtual machine, and ensure that the design qualities of availability, manageability, performance, recoverability, and security are addressed for each component.
2. **Create an installation and configuration document**: This document includes step-by-step instructions on how to install and configure the infrastructure described in the design document. This document is written in such a way that it could be handed off to someone with VCP-level knowledge and they could execute it.
3. **Create the implementation document**: This document describes the implementation at a high-level, to include who is participating, what tasks will be performed and when, and prerequisites for implementation, such as racks that may need to be installed, and redundant power that needs to exist in those racks. This is a common document used in projects run by a project manager.
4. **Create a test plan**: The VCDX candidate will need to be able to prove that the implemented design meets the requirements by describing the tests that need to be passed, as shown in this document.
5. **Create the operations document**: This document is also called the standard operating procedures. It describes common operational tasks that result from maintaining the implemented design over time. Common examples of tasks to include here are how to put a host in maintenance mode, how to deploy a virtual machine from a template, or how to view logs.
6. **Build the bill of materials**: An architect must also be able to describe all the hardware and software needed to implement their design. This is usually shown in a bill of materials document.

Once the VCDX application is submitted and the fee paid, a current VCDX will review the application for completeness and content. Incompleteness is cause for immediate application denial. If it's complete, however, the reviewer will look to see proof of design expertise through thoughtful application of design principles with an emphasis on justifying design decisions and how those decisions impact the design. If your application is sufficient, you'll be invited to defend your design live and in-person at a VMware office. Locations typically include Palo Alto, California; Broomfield, Colorado; Staines, United Kingdom; and Sydney, Australia.

The VCDX has evolved over the years and no longer includes a troubleshooting section. Instead, the defense has two parts: the oral design defense, and the ad hoc design. During the oral design defense, the candidate has 75 minutes to present the design and answer questions from the panelists. VMware recommends the initial presentation take no more than 15 minutes, leaving roughly 60 minutes for the panelists to ask questions that allow the candidate to demonstrate how their design meets the requirements and why they made certain design decisions. Most VCDX certification holders will agree that the most important aspect of this part of the defense is to be able to communicate the **why** of each design decision. If you can justify each decision and make it tie into a customer requirement, you're going to do well.

In the ad hoc design portion, the candidate has 45 minutes to demonstrate their design skills by going through an initial design process in front of the panel. The panelists will pretend to be customers and you, as the virtualization architect, will need to be able to gather their requirements, constraints, make assumptions, identify risks, and begin to build a design based on those inputs. The panel doesn't expect you to create a whole design in 45 minutes; rather, they're trying to assess your design method. To do this, the panel recommends the candidate think out loud and make use of the whiteboard as much as possible. You should try to give the panelists a window into your mind while engaged in your design process.

After finishing both sections of the defense, you'll make the long trip back home. If all went well, you'll receive an email within 10 days stating that you have passed, and welcoming you to the elite VCDX club.

There's more…

The VCDX certification is well known these days, and because of that, there are many more resources online to help you. Your first stop should be the blueprint that can be found at `https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/certification/vmw-vcdx6-dcv-blueprint.pdf`. You'll also want to review the handbook that can be found at `https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/certification/vmw-vcdx6-dcv-handbook.pdf`.

Beyond the official documents linked previously, the boot *IT Architect: Foundation in the Art of Infrastructure Design: A Practical Guide for IT Architects* by *VCDX-001, John Arrasjid*, is a good book to read. You can find it on Amazon.com at `https://www.amazon.com/Architect-Foundation-Infrastructure-Practical-Architects/dp/0996647708/ref=sr_1_2?ie=UTF8&qid=1543255581&sr=8-2&keywords=the+it+architect`.

VMware also offers VCDX workshops, held monthly, that educate candidates on the VCDX process and helps to prepare them for the application and defense. The best resources, however, will be VMware community members who are going through the same experiences as the VCDX candidate. You should use Twitter and the VMware Technology Network forums to connect with like-minded technologists who share the same goal of becoming a VCDX and work with them to review your documentation and application and hold mock defenses. Many successful VCDX holders will say that mock defenses helped them to achieve VCDX status.

Identifying what's new in vSphere 6.7

vSphere 6.7 is the latest release of VMware's virtual data center platform. This release includes features that provide increased scalability, enhanced security, increased availability, and simplified management of the virtual data center infrastructure. A few of the new features and enhancements include the following:

- Support for an embedded **Platform Services Controller** (**PSC**) with **Enhanced Linked-Mode** (**ELM**), which simplifies the vCenter architecture
- vSphere Quick Boot, which reduces ESXi upgrade times by rebooting only ESXi and not the server hardware
- 95% feature parity of the HTML5 vSphere Client versus the Flash-based Web Client
- Encrypted vMotion across vCenter Servers and versions, easing cloud or data center migrations

- **Persistent Memory** (**PMEM**), increasing storage performance capabilities
- **Hybrid Linked Mode** (**HLM**), enabling ease of management between an on-premises vCenter and VMware Cloud on AWS
- Per-VM **Enhanced vMotion Compatibility** (**EVC**), enabling easier cloud migrations
- Instant Clones, formerly known as Project Fargo and vSphere vmFork
- Storage enhancements to UNMAP **vStorage APIs for Array Integration** (**VAAI**) primitive, **Virtual Volumes** (**VVOLs**), and more
- With 6.7 Update 1, the new vSphere Health feature in the HTML5 client

These are just a few of the new features and enhancements introduced with the release of vSphere 6.7. A new version of vSphere, with the new features and enhancements, does not directly change the design process of methodology. The enhancements and features provide an architect with more tools and options for meeting requirements, but can also introduce complexity into the design.

How to do it…

It is important for the architect to understand all the new features and enhancements available. This is a simple, but important, process that includes the following:

1. Access the vSphere 6.7 release notes here: `https://docs.vmware.com/en/VMware-vSphere/6.7/rn/vsphere-esxi-vcenter-server-67-release-notes.html`
2. Access the vSphere documentation sets found here: `https://docs.vmware.com/en/VMware-vSphere/index.html`

How it works…

Reading the vSphere 6.7 release notes gives the architect a summary of the additional features, bug fixes, and known issues. There is also information on the upgrade process and workarounds for known issues.

Reviewing the vSphere documentation, including the Installation and Setup Guide, Upgrade Guide, and Administration Guides, gives the architect a deeper look at new features and how to implement new functionality. The documentation also provides specific requirements that must be satisfied in order to enable a new feature or function. These documentation sets are available online or can be downloaded in PDF, EPUB, or MOBI formats.

There's more…

In the VMware communities, `https://communities.vmware.com/`, there are forums available to discuss vSphere Upgrade and Install at `https://communities.vmware.com/community/vmtn/vsphere/upgradecenter`, and ESXi 6.7 located at `https://communities.vmware.com/community/vmtn/vsphere/esxi`, along with other communities dedicated to each vSphere product. In these forums, an architect or administrator can find real-world issues encountered by other vSphere administrators and architects. Questions and discussions can be posted related to features and issues related to all vSphere products. If you run into issues, or have questions about a specific feature, there are people in the community who are always happy to help.

Planning a vSphere 6.7 upgrade

Upgrading an existing vSphere environment to vSphere 6.7 is a fairly simple process, and can be completed with minimal impact to production with the proper planning.

In this recipe, we will look at the steps required to properly plan an upgrade to vSphere 6.7. We will not cover the specifics of upgrading vCenter Server, ESXi hosts, or any other component of the virtual data center. Specific recipes for upgrading vCenter Server and ESXi host have been included in *Chapter 4*, *vSphere Management Design*, and recipes for upgrading virtual machines to the latest hardware are included in *Chapter 9*, *Virtual Machine Design*.

How to do it…

The following tasks should be completed when planning a vSphere 6.7 upgrade:

1. Verify existing hardware is on the VMware **Hardware Compatibility List** (**HCL**) at `https://www.vmware.com/go/hcl`.
2. Check for interoperability between VMware products using the VMware Product Interoperability Matrix at `http://partnerweb.vmware.com/comp_guide2/sim/interop_matrix.php`.

3. Determine interoperability and support between VMware vSphere 6.7 and third-party hardware and software products.
4. Determine the proper upgrade path and sequence.
5. Note that direct upgrades from vSphere 5.x to 6.7 are not supported. You'll need to upgrade your 5.x environment to 6.0 or 6.5 before upgrading to 6.7.

Completing these steps to properly plan a vSphere 6.7 upgrade will ensure the upgrade can be completed successfully.

How it works…

With each release of vSphere, VMware adds support for new hardware and firmware for devices such as disk controllers, server platforms, and **Network Interface Cards** (**NICs**). VMware also removes support for older hardware and firmware. It is important to verify that the hardware is on the supported compatibility list prior to attempting an upgrade. Failure to validate support for hardware on the HCL can cause significant issues after the upgrade; unsupported hardware may not be available for use or may cause instability in the environment. Replacing unsupported hardware or upgrading firmware on current hardware to a supported configuration may be required as part of the upgrade process.

Checking for interoperability between vSphere products will help to ensure there is minimal impact on functionality during and after the upgrade process. Just like the hardware and firmware, the interoperability between vSphere products changes with each version. New support is added for newer products and features, while support may be removed for older, end-of-support products and features. Details on using the VMware Product Interoperability can be found in *Chapter 4*, *vSphere Management Design*.

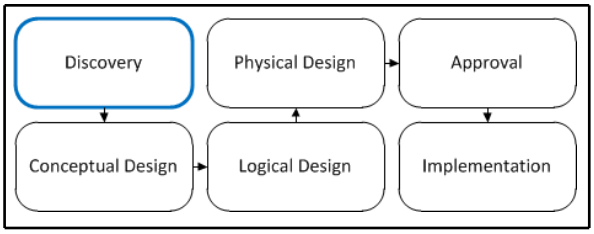
The virtual data center may contain many third-party products that integrate with the vSphere environment. These products often include backup and recovery software, replication software, and management and monitoring applications. Before upgrading to vSphere 6.7, check with each third-party product vendor to validate support for vSphere 6.7 or to determine the requirements for vSphere 6.7 support. This is the step I see missed most often, typically due to not fully understanding dependencies with these products. It is critical to understand what products require integration with the vSphere environment and the impact changes to the environment may have on this products. Again, this is where proper planning from the beginning ensures a successful vSphere 6.7 upgrade.

The final step is to determine the proper upgrade path. If validation of support and interoperability has been completed correctly, this step will likely be the easiest aspect of the process. Once hardware, VMware product, and third-party product interoperability have been validated, a plan can be formulated for upgrading.

Details are important when it comes to the support of hardware and software in the virtual data center. Spending time to properly plan will ensure a successful upgrade to vSphere 6.7.

2 The Discovery Process

This chapter will introduce you to design factors and focus on the discovery phase of the design process. The following diagram displays the phases of the design process:



Phases of the design process

Discovery is the most important phase of the design process. It is also the most time-consuming. The discovery process includes a meeting with the stakeholders to determine business requirements that the design must meet. It also includes current state assessments to determine the technical requirements that the design must satisfy in order to meet customer requirements, which in turn become the design requirements.

During the discovery process, an architect must interact with many different individuals in an organization to collect the necessary information that is needed to begin creating the conceptual design. Decision makers, strategic planners, facilities and maintenance providers, network administrators, storage administrators, application administrators, and application end users can, in some way, be impacted by or gain some benefit from a virtual data center design (some directly and others indirectly). Anyone that may be affected by the design should be identified to be included in the discovery process as early as possible.

The current state assessment is the process of collecting information about the physical resources, such as CPU, memory, and storage, currently supporting the environment. Irrespective of whether the environment is physical servers, virtual servers, or a mix of virtual and physical servers, the current state assessment will identify the total resources available and the total resources actually in use. There are a number of different tools available to perform a current state assessment of an environment. The tool used often depends on the size of the environment. VMware offers a **Capacity Planner** tool that provides a good way to automate this assessment.

For a smaller environment of Windows servers, the Windows **Performance Monitor** (**perfmon**) utility can be used to collect the current state information. For Linux systems, tools such as top, Kinfocenter, and Zabbix can be used to collect and analyze performance data. For environments which are already virtualized on vSphere, the **vSphere Optimization Assessment** (**VOA**) provides useful information on the current state of the environment. If you don't work for VMware or a VMware partner, such as InterVision Systems, which grant access to tools, such as Capacity Planner or the VOA software, very limited sizing information can be discovered by using the free utility RVTools.

Once the design factors have been identified and accepted, the design process continues with logical and physical designs. The logical design maps the requirements to the resources required to satisfy the requirements. The physical design then maps the logical design onto the physical hardware that will provide these resources.

In this chapter, we will cover the following recipes:

- Identifying the design factors
- Identifying stakeholders
- Conducting stakeholder interviews
- Using VMware Capacity Planner
- Using Windows Performance Monitor
- Conducting a VMware optimization assessment
- Identifying dependencies

Identifying the design factors

The design factors are the primary considerations that influence the design. These factors define the function that the design must accomplish, how it should accomplish it, and what may prevent the design from accomplishing it.

How to do it…

The design factors encompass much more than just the physical resources, such as the CPU, memory, and storage, necessary to run workloads in a virtual environment.

Identifying the design factors needs the following requirements:

- Functional and nonfunctional requirements
- Constraints
- Assumptions
- Risks

How it works…

Requirements define what a design must do and how it should do it. Requirements can be business or technical. There are two types of requirements: functional and nonfunctional. The requirements should be clearly defined. A good design requirement is verifiable, traceable, feasible, and specific:

- **Functional requirements**: Identify specific functions of the design or simply what a design must do. Functional requirements can be business or technical in nature. The design must provide a capacity for 10 percent growth over the next three years; this is an example of a functional requirement.
- **Nonfunctional requirements**: Specify how the design must perform or operate. While a functional requirement defines something that the design must do, the nonfunctional requirement defines how or how well it must be done. System response time is an example of a nonfunctional requirement. Nonfunctional requirements become constraints on the design.

- **Assumptions**: These are considered valid until they have been proven otherwise. These factors are considered to be true, but further discovery is required to validate them. As part of the design process, assumptions should be documented and then proven or disproven. Sufficient bandwidth being available between different sites to support site-to-site replication is an example of an assumption, if the bandwidth available between the sites or the bandwidth required for replication has not yet been identified.
- **Constraints**: These place limits on the design choices. Constraints can be business policies or technical limitations. Using a specific vendor for a server's hardware is an example of a technical constraint. The project's budget and the deadlines are also common constraints. Nonfunctional requirements, since they specify how the design must perform or behave, will also become constraints on the design.
- **Risks**: These may prevent the design from being successful. Risks should be clearly identified to minimize surprises that may prevent the successful implementation of the design. A good design will address and mitigate risks.

Since the focus of this chapter is on design discovery, I felt it was important to provide this brief introduction to the design factors. We will dive much deeper into determining and defining the requirements, constraints, assumptions, and risks in *Chapter 3, The Design Factors*.

Identifying stakeholders

A stakeholder is anyone who has an interest in or benefits from the design. A virtual data center design will have at least some impact on many, if not all, areas of an organization and not just those associated with technology.

How to do it…

Identify the key stakeholders, including the following:

- Project sponsors
- Application owners and providers
- System, network, and storage administrators
  Application users

How it works…

Understanding the role of the stakeholders helps an architect to identify who can provide the information necessary to design a successful virtual data center solution. The details of the stakeholders and their roles are specified in the following table:

Stakeholders	Roles
• C-level executives • **Chief Executive Officer (CEO)** • **Chief Financial Officer (CFO)** • **Chief Technology Officer (CTO)**	• Strategic planning for the organization • Setting up business policies and goals • Budget approval • Project sponsorship
Business unit managers or directors	• Strategic planning for the business unit • Managing day-to-day operations • Influencing business policies and goals • Making and/or influencing decisions
Application owners	• Consumers of IT infrastructure • Documenting the application and dependencies • Managing the application functions • Providing day-to-day support for the application
IT	• Technical **Subject Matter Experts (SMEs)** • Network administrators • System administrators • Storage administrators • Help desk
Application or end users	• Consumers of application services • Relying on the infrastructure and applications to accomplish tasks efficiently

Project sponsors are typically C-level executives, **Vice Presidents** (**VPs**), or directors. The project sponsor may also be a committee formed by an organization to evaluate the solutions to business problems or to explore new business opportunities. These stakeholders are often the best resource for obtaining the business requirements that a design must satisfy. If there is a project or a need to explore opportunities, there is a business goal or need driving it. Project sponsors may make the final decision on whether a design has to be approved and accepted for implementation, or they may provide the recommendations for acceptance.

There's more…

Stakeholders or the project team will ultimately be the ones that sign off on or approve the design factors that will be the basis for the logical and physical design. These design factors are identified by analyzing the data collected from the stakeholder interviews and the current state assessments.

The stakeholder's consensus and acceptance of the design factors must be obtained before proceeding with the design process. If you skip this step, you will end up wasting your time and the time of the stakeholders, having to rework areas of the design when requirements are missed, changed, added, or removed.

Define the design factors and obtain acceptance from the project team or stakeholders before taking the next steps in the design process.

Conducting stakeholder interviews

During the discovery process, the primary source of information will be stakeholder interviews. These interviews can be face-to-face meetings or can be done over the phone (or the web). Interviews are not only helpful in collecting information about the business needs and technical requirements, but also keep the stakeholders engaged in the project.

How to do it…

The following are examples of the questions that should be asked in order to determine the business requirements that will influence the design:

- What are the business initiatives, challenges, and goals?
- Are there **Service-Level Agreements** (**SLAs**) in place? What are they?
- What are the **Recovery Time Objective** (**RTO**) and **Recovery Point Objective** (**RPO**) requirements?
- Are there any compliance requirements?
- Who are the SMEs associated with the project?
- Who are the stakeholders?
- Who are the decision makers?
- Are there deadlines that the project must meet?
- Is there a budget for the project? What is the budget for the project?

The following are examples of the questions that should be asked in order to determine the technical requirements that will influence the design:

- Are there any current issues or technical pain points within the environment?
- What are the technology initiatives, challenges, and goals?
- How many servers will be virtualized as part of this project?
- Is there a preferred vendor for the server, network, or storage?
- Have any servers already been virtualized? What hypervisor is being used to host the already-virtualized servers?
- What type of growth is expected over the next three-five years?
- What **Operational Level Agreements** (**OLAs**) are in place?
- Is there a current network, system, storage, and application documentation?

How it works…

Meetings and interviews with stakeholders should maintain some type of structure or formality. Even if it is just a quick call, you should have some type of agenda. I know this may sound like overkill, but it will help you to keep the call or meeting on track and, more importantly, help to ensure that you collect the information you need from the call or meeting.

There are some key items that will help determine the design factors, which are explained as follows:

- **SLAs**: These are a part of a service contract where a service, its availability (uptime and access), and its performance (application response and transaction processing) are defined
- **Service Level Objective (SLO)**: This defines specific objectives that must be achieved as part of the SLA
- **RTO**: This is the amount of time in which a service must be restored after a disruption or disaster
- **RPO**: This is the maximum amount of data loss acceptable due to a disruption or disaster
- **OLAs**: This is an internal agreement that defines relationships between support groups

Do not expect to complete the discovery in a single meeting or interview, especially for a large enterprise project. There will be follow-up questions that may need to be asked, and there will likely be questions that require more research to be answered.

In situations where more research is required, make sure that someone has been assigned with the responsibility to complete the research. Set an expectation on when the research should be completed and the information should be available. You want to avoid the *I thought so-and-so was getting that* situations and keep the discovery process moving forward.

Using VMware Capacity Planner

VMware's Capacity Planner is an inventory and planning tool available to VMware partner organizations, which collects resource utilization information from systems, analyzes the data against industry-standard reference data, and provides the information needed to successfully consolidate the servers into a virtualized environment.

How to do it…

Follow these steps to complete a Capacity Planner engagement:

1. Determine the amount of time for which the Capacity Planner engagement should run based on the business cycle
2. Choose the type of Capacity Planner assessment to be run: a **Consolidation Estimate** (**CE**) or a **Capacity Assessment** (**CA**)
3. Deploy the Capacity Planner collector in the environment to be assessed
4. Verify whether the collector is collecting performance metrics for the systems to be analyzed
5. Collect metrics for the duration of the business cycle
6. Generate Capacity Planner reports

How it works…

A Capacity Planner engagement should typically run for at least 30 days to ensure that it covers a complete monthly business cycle. Thirty days is considered typical since this covers a monthly business cycle where the demand for resources increases during the end-of-month or beginning-of-month processing. It is important that the Capacity Planner capture these increases. The time frame for a Capacity Planner engagement can vary depending on the size and nature of the business.

There are two types of Capacity Planner assessments: CE and CA. The CE assessment provides the sizing estimates of the current environment, while the CA assessment provides a more detailed analysis of the current environment. The CE assessment helps to demonstrate what can be achieved by virtualizing physical workloads, and the CA assessment provides guidance on how systems may be virtualized.

A Capacity Planner collector is installed in the environment that is being assessed. The collector runs as a Windows service and is configured using the VMware Capacity Planner Data Manager. The collector must be installed on a Windows machine, but inventory and performance data can be collected from both Windows and Linux/Unix servers. More than one collector may need to be installed for larger environments. A single collector can collect data from a maximum of 500 systems.

The collector or collectors discover systems in the environment and collect inventory and performance data from the systems. The inventory includes information about the installed physical hardware, operating systems, and installed software.

If running the VMware Capacity Planner Data Manager on a Windows 7 workstation, use **Run as Administrator**.

Performance data metrics are collected on CPU utilization, RAM utilization, disk capacity, and disk I/O. This data is then sent securely to the VMware Capacity Planner Dashboard to be analyzed.

There can be some challenges to setting up the VMware Capacity Planner. Issues with setting up the correct credentials required for data collection and configuring Windows Firewall and services to allow the data collection are common issues that may be encountered.

The following table includes the services and ports that must be open on target systems to allow the Capacity Planner collector to collect data:

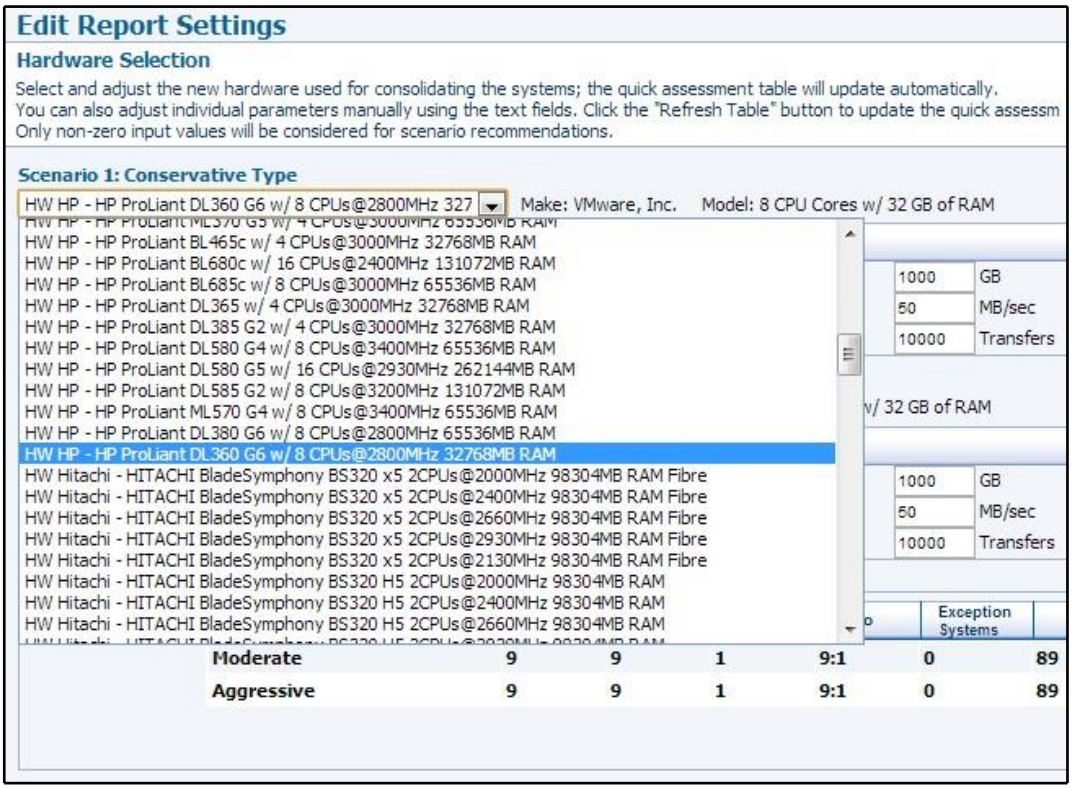
Service	**Port**
Remote Procedure Call (**RPC**)	TCP/135
NetBIOS Name Service (**NBNS**)	TCP/137
NetBIOS Datagram Service (**NBDS**)	TCP/138
NetBIOS Session Service (**NBSS**)	TCP/139
Microsoft-DS	TCP/445
Secure Shell (**SSH**) (Unix/Linux only)	TCP/22

In order to collect data from Windows systems, **Windows Management Instrumentation** (**WMI**), remote registry, and perfmon must be enabled on the target system. For data collection on Linux or Unix systems, port 22 must be open and the **Secure Shell Daemon** (**SSHD**) must be running. Account credentials provided must have at least local administrator rights on the target systems.

There's more…

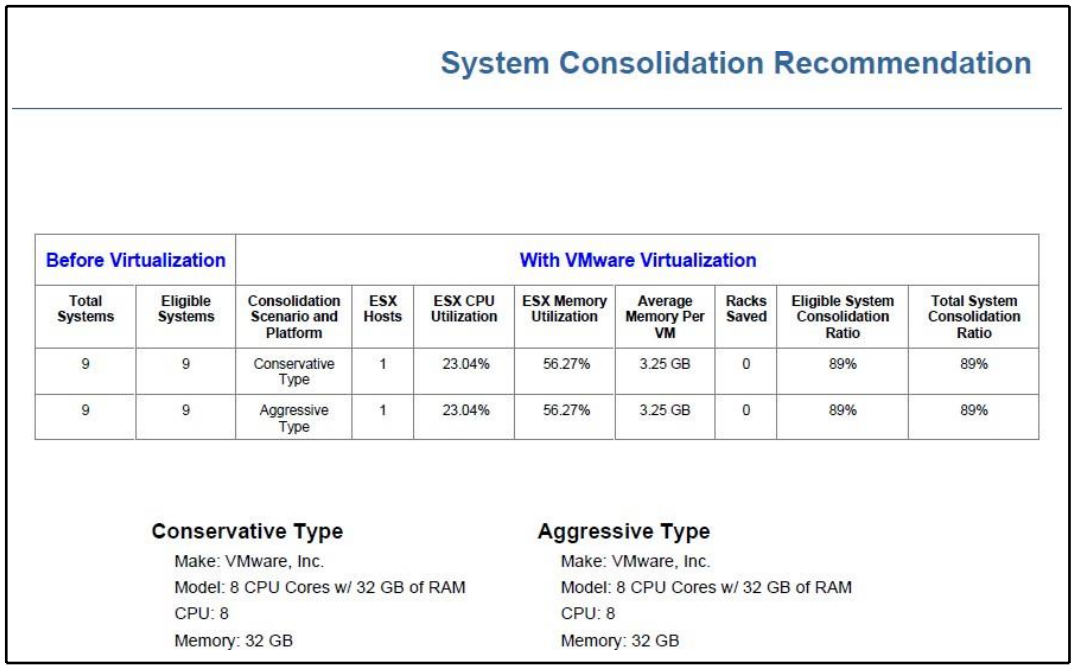
Once the inventory and performance data has been collected, the results can be analyzed and reports can be generated. Some of this information can be viewed and exported from the VMware Capacity Planner Data Manager, but detailed analysis reports are generated from the VMware Capacity Planner dashboard.

If server hardware constraints have been identified during the discovery process, report settings can be adjusted. These constraints will then be applied to the Capacity Planner reporting to determine and show the consolidation ratios that can be obtained using the different hardware configurations. The following screenshot shows the report settings:



Report Settings in Capacity Planner

The reports that are available include the progress report, which provides an overview of the status of the assessment; the executive summary presentation, which provides a high-level summary of the assessment; and the assessment report, which provides information on consolidation ratios and recommendations. Custom reports can also be generated. The following screenshot shows consolidation recommendations:



System Consolidation Recommendation

Before Virtualization		With VMware Virtualization							
Total Systems	Eligible Systems	Consolidation Scenario and Platform	ESX Hosts	ESX CPU Utilization	ESX Memory Utilization	Average Memory Per VM	Racks Saved	Eligible System Consolidation Ratio	Total System Consolidation Ratio
9	9	Conservative Type	1	23.04%	56.27%	3.25 GB	0	89%	89%
9	9	Aggressive Type	1	23.04%	56.27%	3.25 GB	0	89%	89%

Conservative Type

Make: VMware, Inc.
Model: 8 CPU Cores w/ 32 GB of RAM
CPU: 8
Memory: 32 GB

Aggressive Type

Make: VMware, Inc.
Model: 8 CPU Cores w/ 32 GB of RAM
CPU: 8
Memory: 32 GB

Capacity Planner consolidation recommendations

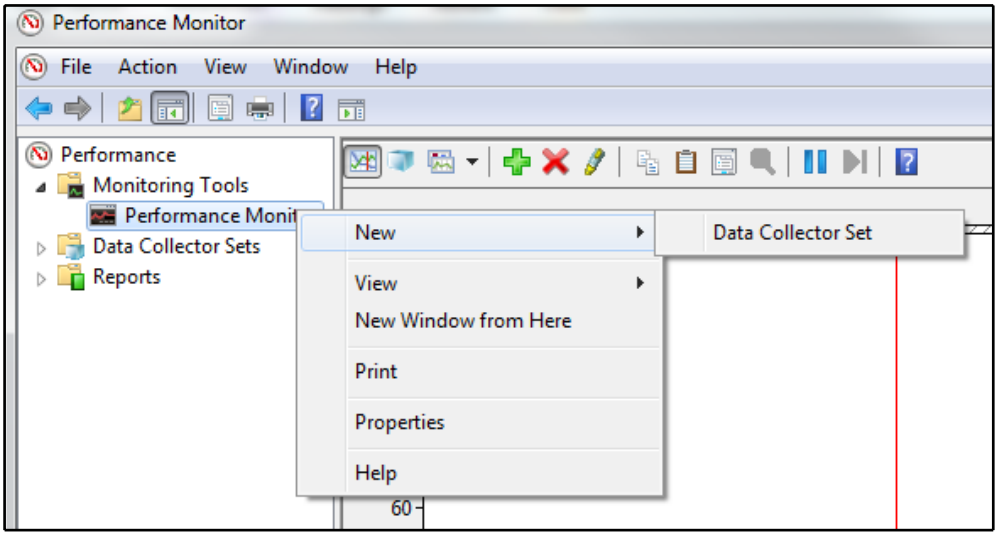
Using Windows Performance Monitor

The Microsoft Windows perfmon can be used to collect performance information, such as CPU utilization, memory utilization, and disk I/O utilization of the Windows servers.

How to do it…

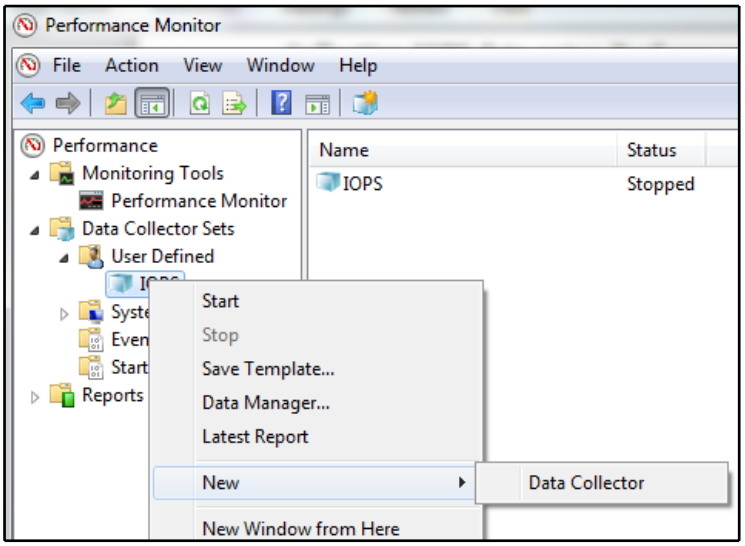
In this example, Microsoft Windows perfmon is used to collect disk I/O metrics, with the following steps:

1. Open **Performance Monitor** and use the **Data Collector Set** wizard to create a user-defined data collector, as displayed in the following screenshot:



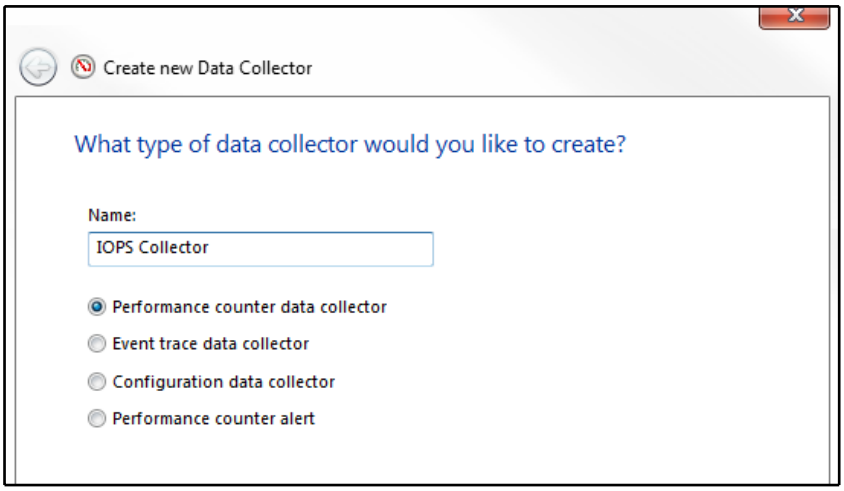
Creating a user-defined data collector in Performance Monitor

2. Once the **Data Collector Set** application has been created, add **New | Data Collector** to the **Data Collector Set**, as shown in the following screenshot:



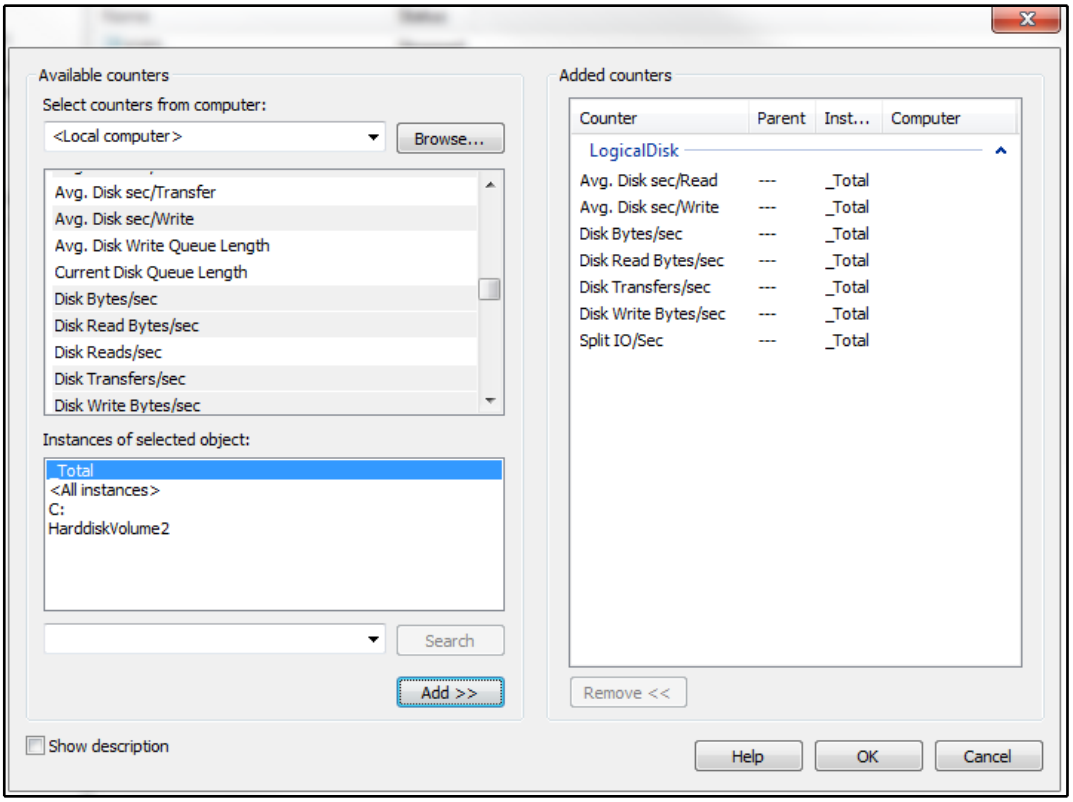
Adding a new Data Collector in Performance Manager

3. Name the new **Data Collector** and select the **Performance counter data collector** radio button, as shown in the following screenshot:



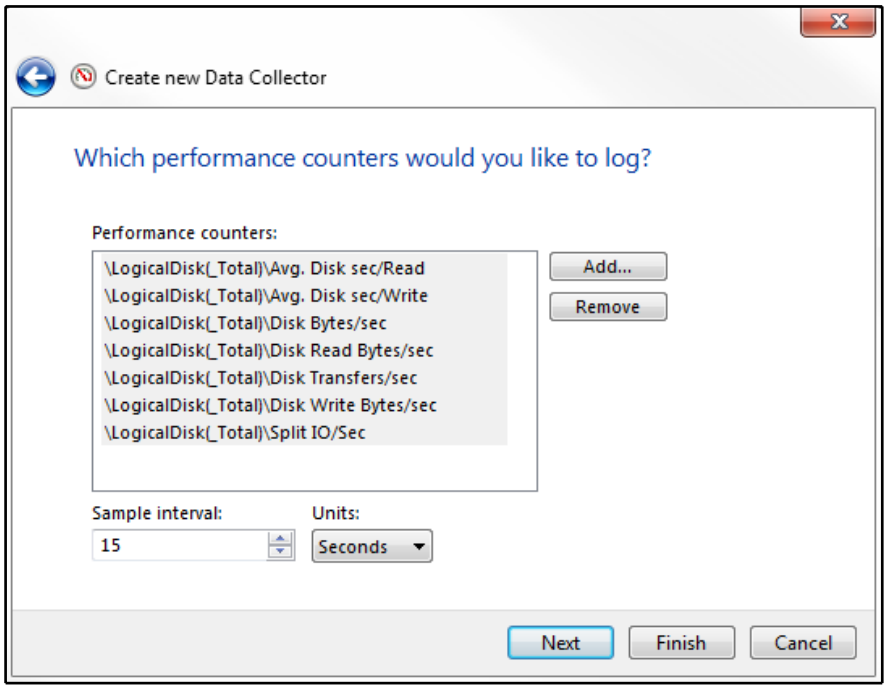
Creating a Data Collector in Performance Manager

4. Add the following counters for the `object _Total` instance to the data collector:



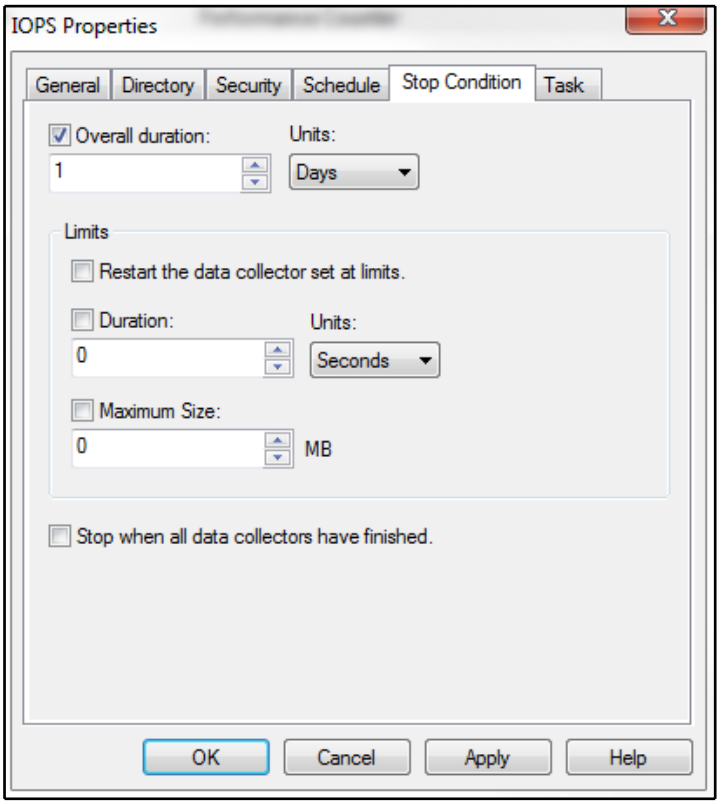
Adding counters in Performance Monitor

5. We will add performance counters as shown in the following screenshot:



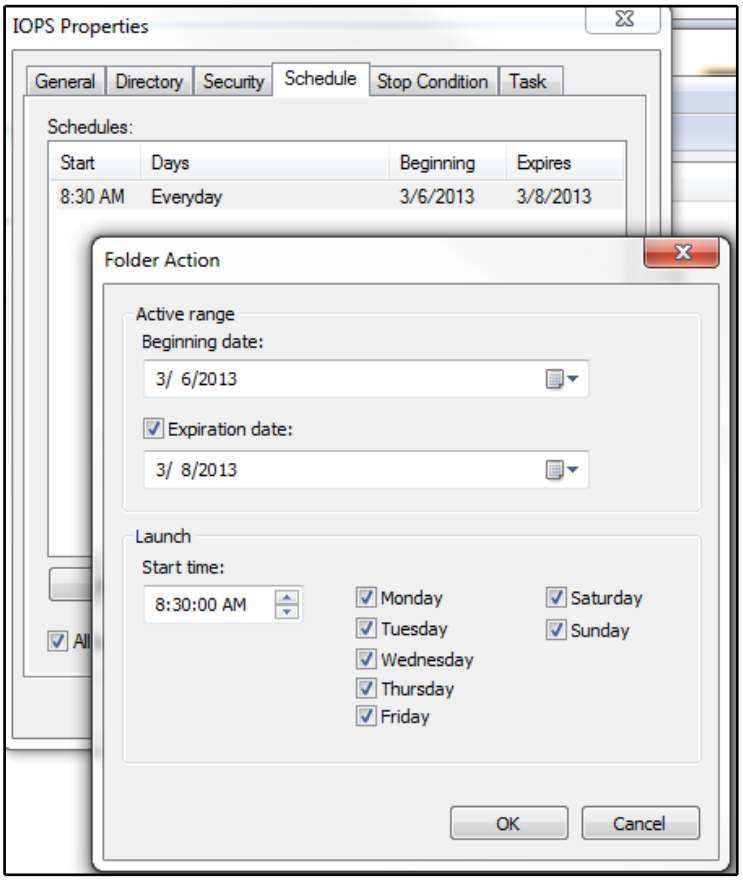
Adding performance counters in Performance Monitor

6. Right-click on the new data collector set, select the **Stop Condition** tab, and change the stop condition to the period of time for which you want to monitor the **Input/Output operations Per Second** (**IOPS**), as shown in the following screenshot:



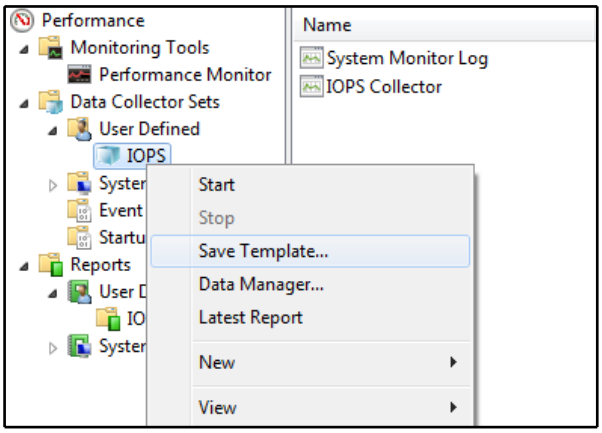
Setting stop conditions in Performance Monitor

7. Data collection for the **Data Collector Set** can be configured to start manually or can be scheduled to start at a future date or time. The following screenshot displays setting a **Schedule** for the **Data Collector Set**:



Configuring the monitoring schedule in Performance Monitor

8. Once the collection process has been completed, you can view the report using the **Reports** section of **Performance Monitor**.
9. A template of the **Data Collector Set** application can be created in order to easily import the **Data Collector Set** on other servers/workstations. This is shown in the following screenshot:



Saving the Data Collector set for use later

How it works…

The total number of IOPS and the I/O profile of a server are necessary to architect the storage required for a virtualized environment correctly. The IOPS and I/O profile are helpful in determining which **Redundant Array of Independent Disks** (**RAID**) level to use with the number and type of disks to be used in order to support the server storage workload.

Windows perfmon can also be configured to collect metrics associated with CPU and memory usage by simply adding the associated counters to the data collector set.

There's more…

Most organizations will have some form of network-or resource-monitoring system in place, such as Nagios, SolarWinds, Splunk, or vRealize Operations Manager. The information monitored and collected by these systems will be useful for the current state assessments. The SMEs should be asked whether there is monitoring in place and for access to the data collected by these systems.

Many vendors also perform free infrastructure assessments. Often, these free assessments are not thorough enough to provide the details necessary for a complete current state assessment, but they can provide some good information. Again, the project SMEs will be asked whether any type of assessments have been done.

Conducting a VMware optimization assessment

The **VMware Optimization Assessment** (**VOA**) is an enhanced evaluation of vRealize Operations Manager, which includes reports providing information about the configuration, capacity, and performance of a vSphere environment. This information is useful for an administrator or architect validating an existing vSphere deployment or planning an expansion to an existing vSphere deployment.

The VOA will provide useful insights into a virtual environment by providing detail analytics, including the following:

- Providing information on misconfigured clusters, hosts, and virtual machines
- Identifying potential performance problems with root cause analysis
- Analyzing virtual machine resources to identify undersized and/or oversized virtual machines, providing opportunities to attain a right-size environment

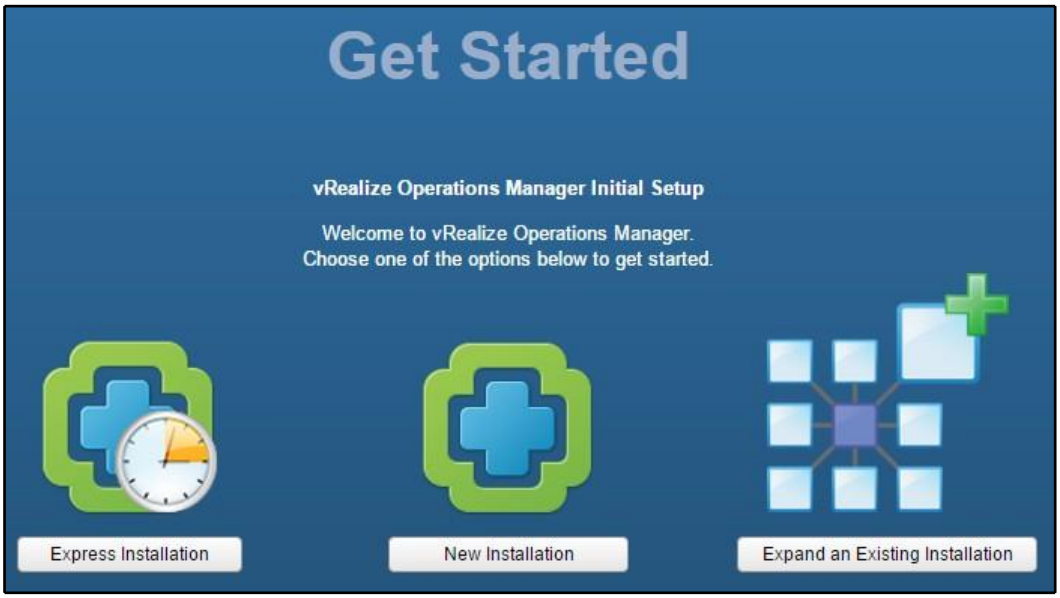
The information gathered during a VOA will allow an administrator or architect the ability to quickly identify health issues, risks to the environment, and areas where efficiency can be improved.

How to do it…

Follow these steps to obtain, deploy, configure, and conduct an optimization assessment using the VOA appliance:

1. Visit `https://www.vmware.com/assessment/voa` and download the VOA appliance.
2. Import the VOA Appliance OVA into the vCenter inventory.

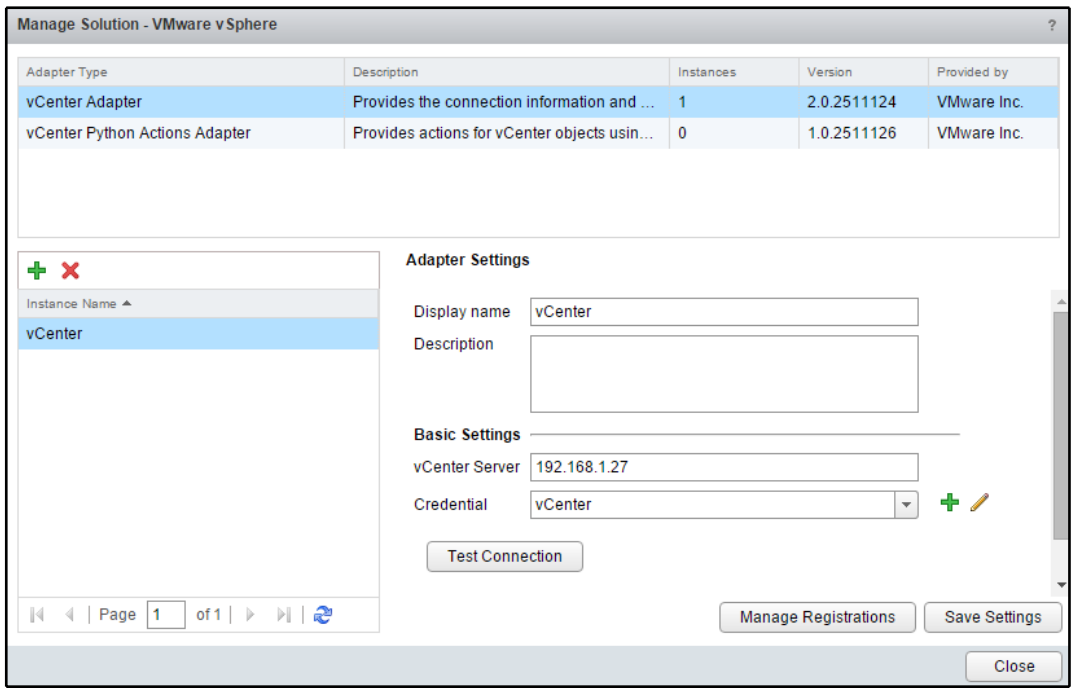
3. Power on the VOA appliance and access the VOA appliance IP address with a web browser to launch the **vRealize Operations Manager Initial Setup** wizard and choose **Express Installation**, as shown in the following screenshot:



The VOA splash page

4. Set the Administrator Password when prompted by **the vRealize Operations Manager Initial Setup** wizard.

5. Configure the vCenter Adapter by providing a **Display name**, **vCenter Server** IP address, and **Credential**. Use the **Test Connection** button to test connectivity and credentials. Be sure to **Save Settings** once the test is successful. The **vCenter Adapter** configuration window is shown in the following screenshot:



Registering the vCenter adapter in VOA

6. Once the **vCenter Adapter** is configured, it will begin collecting performance and configuration information from the configured vCenter Server.
7. Access the VOA appliance with a web browser to view information on environment health, risks, and efficiency.

How it works…

Once deployed and configured, the VOA appliance begins collecting information about the virtual environment. The information is analyzed and displayed as part of the VOA dashboard. The **Health**, **Risks**, and **Efficiency** of the environment is displayed in the VOA dashboard, as shown in the following screenshot:

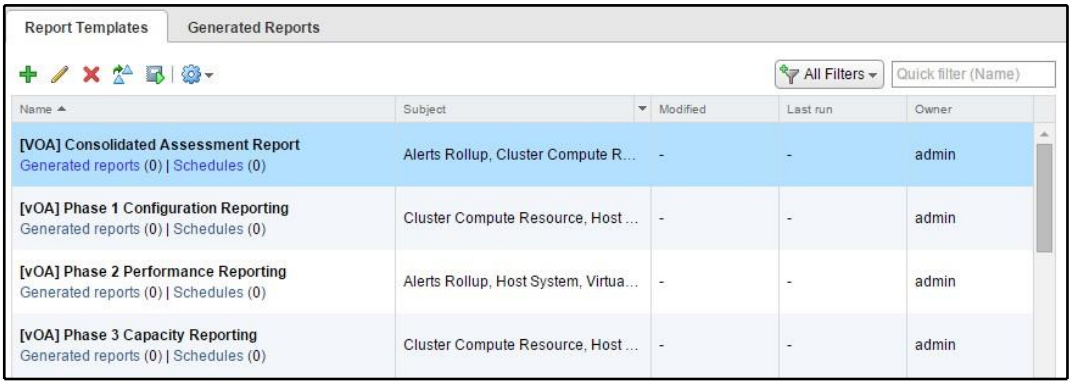
The VOA dashboard

The VOA reporting is split up into three phases. The phases correspond with specific metrics, which will be available over time as the VOA appliance collects and analyzes information from the environment. The following phases make up the optimization assessment:

- **Phase one**: This is the configuration phase. Phase one provides analysis of the configuration of the environment and corresponds with the **Environment Health** dashboard. The information in this phase is available within 24 hours of deploying the VOA appliance.
- **Phase two**: This is the performance phase. Phase two provides analysis of performance information in the environment and identifies risks associated with exceeding available capacity and performance. Phase two requires VOA collection for five to seven days.

- **Phase three**: This is the optimization phase. In this phase, the areas where capacity and performance can be optimized are identified. This includes details such as virtual machines with resources that have been over-allocated. This final phase requires the collection of environment data by the VOA over a period of about 21 days.

Preconfigured reports are included for each phase of the VOA. These reports can be generated for the VOA appliance, as shown in the following screenshot:



Default reports in VOA

These preconfigured reports provide valuable insight that will assist an architect in determining what will be necessary to meet requirements around growth or expansion of an existing vSphere environment.

Identifying dependencies

A dependency is a relationship among systems or services. During the discovery process, dependencies should be identified and documented. In *Chapter 1*, *The Virtual Data Center*, we discussed the importance of taking a holistic view when designing a virtualized environment. Identifying dependencies is the key to the holistic approach of designing.

How to do it…

An architect must identify dependencies in order to understand what effect a design decision or change may have on other services. The architect should identify the following dependencies:

- Physical infrastructure dependencies
- Application and service dependencies

How it works…

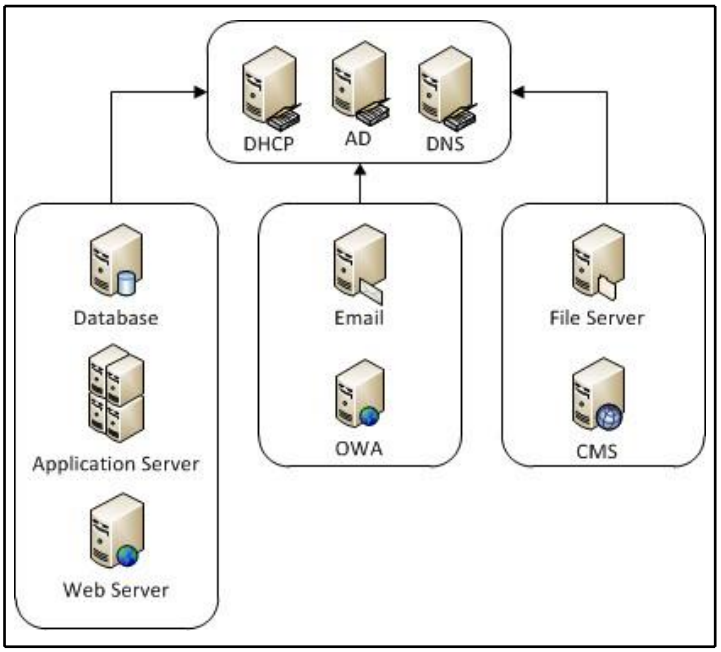
Dependencies can be service-to-service; for example, a web application depends on a frontend web server and a backend database. Dependencies can be service-to-infrastructure; for example, a web application requires a static IP address and a minimum of 10 MB of network bandwidth.

Physical and infrastructure dependencies are generally easier to discover and are commonly documented. Applications will have dependencies, which include server resources, network resources, and storage resources. Infrastructure dependencies that are not documented are often readily discovered as part of the current state assessment. The following table is an example of how physical application dependencies can be documented:

Application	**OS**	**CPU cores**	**Speed (GHz)**	**RAM (GB)**	**Network (GBps)**	**Network (VLAN)**	**Storage**
IIS	Win2k8 R2	4	2.7	16	1	22	50 GB
SQL database	Win2k8 R2	8	2.7	32	1	22	1 TB

Service-to-service or application-to-service dependencies can be a bit more difficult to discover. Application owners, application developers, application documentation, and application vendors will be the best sources for determining these dependencies. As with Capacity Planner, if you work for VMware or a VMware partner, you're entitled to use VMware **Application Dependency Planner** (**ADP**).

The following diagram is an example of how service dependencies can be mapped and documented:



Example application dependency diagram

Understanding the dependencies will help an architect to understand how a change made to one area of the design may have an effect on another area of the design. Mapping and documenting application dependencies will provide the necessary information to properly design a solution for business continuity and disaster recovery. Understanding the dependencies will also aid in troubleshooting issues with the design implementation.

Beware, there may be undocumented dependencies that are not easily discovered. This can often be a risk to the design, especially in an organization with a legacy of unsupported applications or applications developed in-house that have not been properly documented.

I have seen issues where a specific configuration, such as an IP address or a file location, has been hardcoded into an application and not documented. A change is made to the environment and hence the application becomes unavailable. Dependencies of this type can be extremely difficult to plan for and discover.

There are tools available that can help you to discover application dependencies automatically. If you work for VMware or a VMware partner, you have free access to the VMware **Application Dependency Planner** (**ADP**) software. ADP ships as a pre-built OVF virtual appliance and comes with several components. There's a database to store all the dependency information, collector VMs to receive Ethernet traffic, and an Aggregator VM which acts as a manager and central configuration point of the application. The idea of ADP is to sniff Ethernet traffic to understand the communication paths between servers in an environment. ADP then builds a visual map that helps consultants to easily understand application dependencies. The map includes IP addresses, hostnames, port numbers, and protocols which helps to quickly identify which servers are talking with one another.

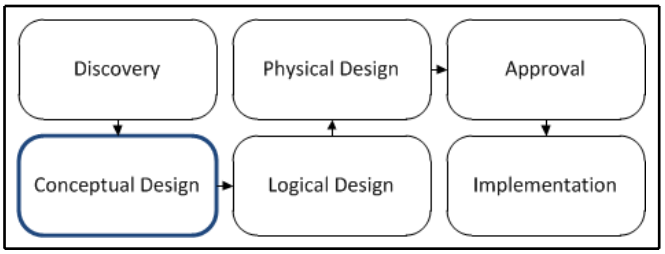
3
The Design Factors

During the vSphere design discovery process, information is collected on the business and technical goals of the virtualization project. This information must be analyzed in order to determine the vSphere design factors.

The vSphere design factors that must be determined are as follows:

- Requirements
- Constraints
- Assumptions
- Risks

Determining the requirements, making and proving assumptions, determining constraints, and identifying risks forms the conceptual design and provides the foundation to build on for the logical design. Business and technical design factors that are identified as part of the conceptual design will be mapped to the resources that are necessary to satisfy them during the logical design process.
The conceptual design phase is highlighted within the overall design and implementation flow diagram:



Conceptual design phase of the overall design and implementation workflow

In our example design, after conducting interviews with stakeholders and performing technical assessments of the environment, the following information has been collected about the project's goals, current environment, and business factors that will influence the design:

- Currently, there are 100 physical servers, each hosting a single application. Each application services 10 customers.
- The business expects to add 50 new customers over the next year.
- The solution must support growth over the next 5 years.
- Application uptime and accessibility is very important.
- Consolidate physical servers to reduce hardware costs associated with the maintenance and deployment of new application servers.
- No more than 20 application servers, or 200 customers, should be affected by a hardware failure.
- There should be a 1-hour maintenance window each month for application and hardware maintenance. Hardware maintenance is currently a challenge. Since hardware and application maintenance cannot be performed at the same time, the maintenance window does not typically provide the time that's required to perform both application and hardware maintenance.
- Application servers run Microsoft Windows 2016 as the operating system.
- Each application server is configured with 8 GB of memory. The peak usage of a single application server is approximately 65 percent or approximately 5.2 GB.
- Each application server is configured with two dual-core 2.7 GHz processors. The peak usage of a single application server is approximately 10 percent of the total processing power, or approximately 1 GHz.
- Each application server is configured with 100 GB of disk space. Peak disk capacity usage of a single application server is approximately 65 percent of the total disk space, or 65 GB. Peak disk performance of a single application server is 50 IOPS with an IO profile of 90 percent read and 10 percent write.
- Currently, the stakeholders are using HP DL380 servers. The infrastructure team is very familiar with the management and maintenance of these servers and wants to continue using them.
- Currently, there is no shared storage. The current system and infrastructure administrators are unfamiliar with the shared storage concepts and protocols.

- Cisco switches are used for network connectivity. Separate VLANs exist for management connectivity and production application connectivity.
- Currently, each physical server contains a single gigabit network interface card. Peak network usage is 10 Mbps.
- Server logs are auditable and must be retained for 6 months. All logs should also be sent to a central syslog server that is already in place.
- If an application server fails, the current recovery time is around 8 hours. The solution should reduce this time to less than 4 hours.
- The management team expects the implementation to be completed before the third quarter of the year.
- There is an approved project budget of $200,000.

In this chapter, we will use this information to determine the design factors so that we can create the conceptual design. Throughout the design process, each design decision is mapped back to these design factors.

In this chapter, we will cover the following topics:

- Identifying design requirements
- Identifying design constraints
- Making design assumptions
- Identifying design risks
- Considering infrastructure design qualities
- Creating the conceptual design

Identifying design requirements

The design requirements specify the functions that the design must perform and the objectives that the design must meet.

There are two types of requirements: functional requirements and nonfunctional requirements. Functional requirements specify the objectives or functions that a design must meet. Nonfunctional requirements define how the design accomplishes the functional requirements.

Typical functional requirements include the following:

- Business goals
- Business rules
- Legal, regulatory, and compliance requirements
- Application system requirements
- Technical requirements
- Administrative functions

Typical nonfunctional requirements include the following:

- Performance
- Security
- Capacity
- Availability
- Manageability
- Recoverability

While identifying and defining the requirements, separate the functional requirements from the nonfunctional requirements; nonfunctional requirements are design constraints and will be documented separately.

Since functional requirements define what the design must accomplish, once identified and approved, these requirements typically cannot be easily changed during the design process.

How to do it…

The following high-level steps can be used to fully identify design requirements:

1. Analyze the business and technical information that's collected during the discovery process
2. Determine the functional and nonfunctional requirements of the design
3. Document the design requirements

How it works…

While defining the requirements, each requirement should be clearly stated and specified. Define requirements individually; multiple requirements should not be combined into a single requirement.

During the discovery process, the following information about the current size of the existing environment is identified:

- Currently, there are 100 physical servers, each hosting a single application. Each application services 10 customers.
- Not more than 20 application servers or 200 customers should be affected by a hardware failure.
- Consolidate physical servers to reduce the hardware costs associated with maintaining and refreshing the hardware of the existing application servers.

One of the goals of the project is to consolidate the physical servers to reduce hardware costs. An example design requirement to support this might be as follows: consolidate existing physical servers.

This requirement is vague and with the information that's available from the discovery, the requirement should be more specific. Based on the number of existing physical servers and the maximum number of customers that should be impacted during a hardware failure, a better requirement example may be as follows: consolidate the existing 100 physical application servers down to five servers.

Information about the expected growth of the environment was also discovered:

- The business expects to add 50 new customers each year
- The solution must support growth over the next 5 years

Based on this information, there is a requirement that the environment must be designed to provide the capacity that's necessary to support future growth. An example requirement to support this might be as follows: provide sufficient capacity to support growth.

Again, this requirement is very vague and does not provide any information about how much the growth will be or over what period of time is growth expected. From the discovery, it is known that the business expects to add 50 new customers over the next year. Each server hosts a single application, which will provide service for 10 customers. The solution should support growth over the next 5 years.

Using this information, a requirement that specifies the growth that should be supported and the time period over which this growth is expected is as follows: provide capacity to support growth for 25 additional application servers over the next 5 years.

The architect can also determine the availability of requirements for hardware maintenance and application resiliency:

- The solution must allow a 1-hour maintenance window each month for application and hardware maintenance. Hardware maintenance is currently a challenge. Since hardware and application maintenance cannot be performed at the same time, the maintenance window does not typically provide the time required to perform both application and hardware maintenance.
- Application uptime and accessibility is very important.
- Not more than 20 application servers or 200 customers should be affected by a hardware failure.

From this information, the requirement that might be identified is that the server hardware maintenance should not affect application uptime, and redundancy should be maintained during hardware maintenance operations.

The problem with this requirement is that it includes two separate requirements; one requirement is for application uptime, and another requirement is for redundancy. These requirements should be split into two individual requirements.

Server hardware maintenance should not affect application uptime. Provide *N+2* redundancy to support a hardware failure during normal and maintenance operations.

There's more…

Once the functional requirements have been identified and defined, the requirements should be recorded in the design documentation as part of the conceptual design.

There are a number of formats that can be used, such as bulleted lists and numbered lists, but a simple table works well. Assigning an ID to each requirement makes it easier to reference the requirement later in the design document:

ID	Requirement
R001	Consolidate the existing 100 physical application servers down to five servers
R002	Provide capacity to support growth for 25 additional application servers over the next 5 years
R003	Server hardware maintenance should not affect application uptime
R004	Provide N+2 redundancy to support a hardware failure during normal and maintenance operations

Identifying design constraints

Design constraints are factors that restrict the options the architect can use to satisfy the design requirements. Once the functional and nonfunctional requirements have been identified, they are separated. The nonfunctional requirements that define how requirements must be satisfied become the constraints on the design.

Design constraints include the following:

- Technology constraints, such as hardware vendors, software solutions, and protocols
- Operational constraints, such as performance and accessibility
- Financial constraints, such as budgets

Unlike functional requirements, the constraints and nonfunctional requirements may change during the design process. This holds true, especially if the constraint introduces risks into the design. For example, if an identified constraint that requires a specific model of hardware to be used prevents the design from satisfying a functional requirement, the constraint may need to be changed or adjusted.

How to do it…

The high-level steps to identify the constraints of a design are as follows:

1. Analyze the business and technical information that's collected during the discovery process
2. Determine the nonfunctional requirements of the design

3. Nonfunctional requirements are constraints on the design
4. Identify any other constraints on the design
5. Document the design constraints

How it works…

As with functional requirements, when defining the nonfunctional requirements or constraints, they should be clearly stated and specified. Define each constraint individually; do not combine multiple nonfunctional requirements into a single constraint.

Currently, HP DL380 servers are used. The infrastructure team is familiar with the management and maintenance of these servers and wants to continue using them.

This statement does not identify something the design must do. It is placing a constraint on the design by providing a specific type of hardware that should be used. The following is an example of the constraint that can be formed from this statement:

- HP DL380 servers should be used for compute resources

Budgetary constraints affect nearly all the projects. There will likely be a limit on the amount of money a company will want to spend to accomplish a goal.

If a budget has not been established for a project, it is likely that the business has not committed to the project. Beware the infinite budget.

During the design discovery, the following budget was identified for this project: there is an approved project budget of $200,000.

This budget constraint can simply be stated as follows: a project budget of $200,000.

Operational constraints are also common. Often, there will be existing processes or policies in place that will need to be factored into the design. Often, you will need to accommodate the existing monitoring and management applications in the design. An example of an operational requirement is as follows: server logs are auditable and must be retained for 6 months. All logs should also be sent to a central syslog server that is already in place.

Here, a functional and nonfunctional requirement can be identified. The functional requirement is that the server logs are auditable and must be retained for 6 months. This functional requirement defines something the design must do, but there is also a constraint on how the design must accomplish this, and that is by using syslog to send logs to a central server. Based on this information, the constraint is as follows: syslog should be used to send server logs to an existing central syslog server.

There's more…

Constraints should be documented as part of the conceptual design. Just as you used a table to document the design requirements, using a simple table works well when documenting the design constraints. Each constraint is assigned an ID so that it can be easily referenced later in the design document:

ID	Constraint
C001	HP DL380 servers should be used for compute resources
C002	A project budget of $200,000
C003	Syslog should be used to send server logs to an existing central syslog server

Making design assumptions

Assumptions are made by the architect and have not yet been validated. Assumptions are accepted as a fact until they have been validated or invalidated. As part of the design process, each assumption needs to be validated as a fact. If an assumption cannot be validated, a risk will be introduced into the design.

How to do it…

Any assumptions that are made will need to be defined and documented as follows:

1. Identify any assumptions that have been made about the design
2. Document the design assumptions

How it works…

Common assumptions relate to power, space, and cooling. A common example of an assumption that an architect may make is as follows:

- There is sufficient power, cooling, and floor/rack space available in the data center to support both the existing and consolidated environment during the migration

While working through the physical design, the power, cooling, and space requirements will need to be identified and the assumptions validated. A goal of this project is to consolidate the existing physical servers. The overall need for power, cooling, and space will be reduced once the project is complete, but enough of these resources need to be available to support both the existing physical environment and the new consolidated environment during the consolidation process.

Referring to our mock design, a requirement was identified based on the discovery information to provide *N+2* redundancy:

R004	Provide *N+2* redundancy to support a hardware failure during normal and maintenance operations

This requirement was defined based on the following discovery information:

- A 1-hour maintenance window each month for application and hardware maintenance. Hardware maintenance is currently a challenge. Since hardware and application maintenance cannot be performed at the same time, the maintenance window does not typically provide the time that's necessary to perform both application and hardware maintenance.
- Application uptime and accessibility is very important.

What assumption may have been made when defining this requirement?

An assumption was made based on the importance of application uptime and accessibility that there should be sufficient resources to provide redundancy not only during normal operations, but also in the event of a host failure. When a host is unavailable due to maintenance being performed, the following approach should be adopted:

- Resources should be provided to support a host failure during both normal and host maintenance operations

The following requirements relating to growth were also defined:

- The business expects to add 50 new customers over the next year
- Support growth over the next 5 years

ID	Requirement
R002	Provide capacity to support growth for 25 additional application servers over the next 5 years

An expected, the growth of 50 customers over the next year was identified, but the design is expected to support growth over the next 5 years. To create this requirement, an assumption was made that growth would be the same over years two through five, resulting in the documentation of the following assumption: growth is calculated based on the addition of 50 new customers each year over the next 5 years.

The company may have a forecast for growth that exceeds this. If this assumption is incorrect, the design may not meet the defined requirement.

There's more…

Assumptions should be documented in the design document. As with documenting design requirements and constraints, use a table for this. Each assumption is assigned an ID so that it can be easily referenced later in the design document:

ID	Assumption
A001	Sufficient power, cooling, and floor/rack space is available in the data center to support the existing and consolidated environment during the migration
A002	Resources should be provided to support a host failure during both normal and maintenance operations
A003	Growth is calculated based on the addition of 50 new customers each year over the next 5 years

Identifying design risks

Risks include anything that may prevent the design from satisfying the requirements. Design risks include the following:

- Technical risks
- Operational risks
- Financial risks

Risks are often introduced through constraints or assumptions that have not been proven. Risks resulting from assumptions are mitigated by validating them. When risks are not mitigated, the project may not be successful.

How to do it…

Throughout the design process, design decisions should mitigate or minimize risks. The following steps will help you do that:

1. Identify any risks associated with the design requirements or assumptions
2. Validate assumptions to reduce the risks associated with them
3. Determine how design decisions will help mitigate or minimize risks

How it works…

There are a few risks in the design based on the discovery information, assumption, and constraints.

As a part of the discovery process, the following risk was noted:

- Currently, there is no shared storage. The current system and infrastructure administrators are unfamiliar with the shared storage concepts and protocols.

These operational risks were identified during discovery. Operational risks can be minimized by providing implementation and operational documentation.

There is a technical constraint that may also introduce risks, and is as follows:

ID	Constraint
C001	HP DL380 servers should be used for compute resources

This constraint may introduce some risks to the environment if the capabilities of the HP DL380 servers are not able to fulfill the requirements. Can the servers be configured with the processing and memory required by the requirements? Are there enough expansion slots to support the number of network ports or HBAs required? It may be necessary to remove or change this constraint if the HP DL380 server is not able to fulfill the technical requirements of the design.

An assumption was also made with regards to the growth of the environment over the next 5 years: growth is calculated based on the addition of 50 new customers each year over the next 5 years.

If this assumption is not validated and growth is forecasted by the company to be higher in 2 to 5 years, the design will be at risk to not meet the growth requirements. Validating this assumption will mitigate this risk.

Considering infrastructure design qualities

What makes a good infrastructure design? The answer could be summarized by saying that it includes the following qualities:

- Availability
- Manageability
- Performance
- Recoverability
- Security

The infrastructure design qualities, also called design characteristics, should be incorporated into every enterprise design. We saw these when we described them earlier in this chapter as nonfunctional requirements. That is to say, they describe how a design should work. In addition to meeting the customer's requirements and constraints, it may not be a good design overall if the design qualities were not considered.

Throughout the design process, it's a good idea to understand how each design decision impacts the infrastructure design qualities. Continually ask yourself: if the design qualities are my end goal, how does this design decision affect them?

How to do it…

The following steps describe the process you can take to ensure your designs take the infrastructure design qualities into consideration for each requirement, constraint, or assumption:

1. Understand the requirement, constraint, or assumption
2. Make a design decision to meet the requirement, constraint, or assumption
3. Document the impact of the design decision on each infrastructure quality

How it works…

Let's take requirement R003 from earlier in this chapter as an example. This requirement states that server hardware maintenance should not affect application uptime. To meet this requirement, each vSphere cluster will be designed with *N*+2 redundancy to support a host failure during the maintenance of another host. The following table describes how you might document this design decision against the infrastructure qualities:

Infrastructure quality	**Impact**
Availability	Increases availability by providing additional hosts on which vSphere HA can restart VMs
Manageability	Introduces additional hosts in each cluster that need be managed, patched, and backed up
Performance	Provides additional resources from which VMs can be run
Recoverability	Negligible impact to recoverability
Security	Increased attack surface by introducing an additional host in each cluster

There's more…

So far, we have requirements, constraints, and assumptions mapping to design decisions, which map to infrastructure design qualities. In addition, you should ensure that each component in your design addresses each design quality. Typically, the vSphere design components are identified as follows:

- Compute
- Storage
- Network

- Virtual machine
- Management

So, a simple way to ensure a complete design is to build the following table, which maps each vSphere component to each infrastructure design quality. While building your design, you can check off the mapping of component to quality as you address it in your design. Don't expect to be able to fill in the table with a short description of how you approached each mapping. The topics should be far too large to do so. Rather, use the table as a reference to ensure that each area is addressed in your design:

vSphere component	**Availability**	**Manageability**	**Performance**	**Recoverability**	**Security**
Compute	Yes	Yes	Yes	Yes	No
Storage	No	No	No	No	No
Network	Yes	Yes	No	Yes	Yes
Virtual machine	No	yes	Yes	yes	No
Management	Yes	Yes	Yes	Yes	Yes

Creating the conceptual design

The conceptual design is created with the documentation of the requirements, constraints, and assumptions. The design documentation should include a list of each of the design factors. The conceptual design guides the design. All logical and physical design elements can be mapped back to the conceptual design to provide justifications for design decisions.

How to do it…

To create the conceptual design, follow these steps:

1. Use the design factors to form the conceptual design
2. Organize the design factors to be easily referenced during the design process
3. Create high-level diagrams that document the functional blocks of the design

How it works…

The conceptual design should include a brief overview that describes the key goals of the project and any factors that may drive the business decisions related to the project. The conceptual design includes all the identified requirements, constraints, and assumptions.

The following pointers explain an example of conceptual design:

- The primary goal of this project is to lower hardware cost through the consolidation of physical application servers. The design will increase application uptime and resiliency and reduce application recovery time.
- The design will attempt to adhere to the standards and best practices when these align with the requirements and constraints of the design.

Design requirements

Requirements are the key demands on the design. The design requirements are as follows:

ID	Requirement
R001	Consolidate the existing 100 physical application servers down to five servers
R002	Provide capacity to support growth for 25 additional application servers over the next 5 years
R003	Server hardware maintenance should not affect application uptime
R004	Provide *N*+2 redundancy to support hardware failure during normal and maintenance operations

Design constraints

Constraints limit the logical decisions and physical specifications. Constraints may or may not align with the design objectives. The design constraints are as follows:

ID	Constraint
C001	Covered in the *Identifying design constraints* recipe and its *There's more…* section of this chapter
C002	A project budget of $200,000
C003	Syslog should be used to send server logs to an existing central syslog server

Assumptions

Assumptions are the expectations of a system that have not yet been confirmed. If the assumptions are not validated, risks may be introduced. Assumptions are listed as A001, A002, and A003.

There's more…

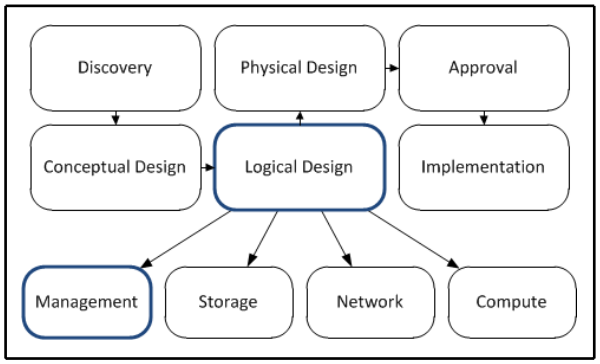
The conceptual design can also include diagrams that provide high-level overviews of the proposed design. Conceptual diagrams of the functional blocks of the design include the virtualization infrastructure, storage, servers, and networking. A conceptual diagram does not include specifics about the resources that are required, or hardware vendors.

The conceptual diagram should show, at a very high level, how servers will be placed in a vSphere **High Availability** (**HA**)/**Distributed Resource Scheduler** (**DRS**) cluster. The existing physical network infrastructure will be leveraged to provide connectivity for IP storage and the virtual machine networks. The diagram does not include any specifics about the type of servers, type of array, or the resources required, but it does provide an overview of how the different parts of the design will work together.

4
vSphere Management Design

This chapter will cover the design considerations that should be taken into account when designing the management layer of the virtual infrastructure. We will look at the different components that make up vCenter, how to size them correctly, and how to ensure compatibility between the VMware products that are deployed in the environment. This chapter will also cover the different deployment options for vCenter and its components, as well as the importance of the availability, recoverability, and security of these components.

The following diagram displays how management design is integrated into the design process:



Management design within the vSphere design workflow

Questions that the architect should ask and answer during the management design process include the following:

- What components are necessary to manage the virtual environment?
- How will management components be deployed?
- What resources are required to support the management components?
- What impact will the loss of a management component have on the environment?
- How can we recover from the loss of a management component?
- How can we upgrade and patch management components?

In this chapter, we will cover the following recipes:

- Identifying vCenter components and dependencies
- Selecting a vCenter deployment option
- Determining vCenter resource requirements
- Selecting a database for the vCenter deployment
- Determining database interoperability
- Choosing a vCenter deployment topology
- Designing for management availability
- Designing a separate management cluster
- Configuring vCenter mail, SNMP, and alarms
- Using Enhanced Linked Mode
- Using the VMware Product Interoperability Matrix
- Backing up vCenter Server components
- Planning vCenter HA to increase vCenter availability
- Upgrading vCenter Server
- Designing a vSphere Update Manager Deployment

Identifying vCenter components and dependencies

The vCenter Server provides the central configuration and management of the ESXi servers and the services provided by the virtual infrastructure. vCenter 6.7 is composed of several components and services, such as the **Platform Services Controller** (**PSC**), the vCenter Server database, and the vCenter Server.

How to do it…

Following are the procedure to identify vCenter components and dependencies:

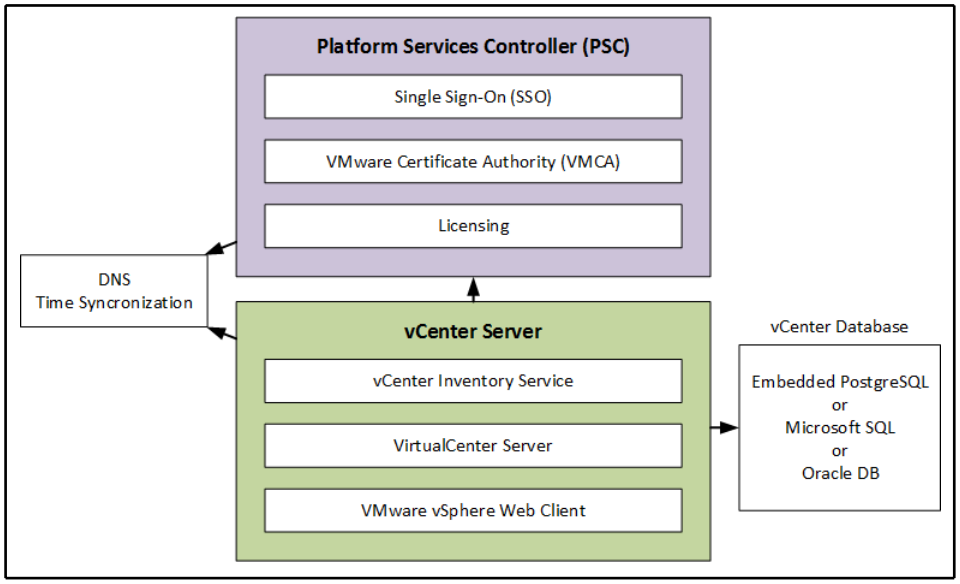
1. Identify the following core components and services of vCenter 6.7:
    - The VMware PSC was introduced in version 6.0. The PSC handles security functions in the vSphere infrastructure. The PSC provides the vCenter **Single Sign-On** (**SSO**) service, licensing management, registration services, and the **VMware Certificate Authority** (**VMCA**). The PSC can be deployed as a standalone server or embedded on the same server with other required vCenter components.
    - vCenter SSO is deployed as a part of the PSC in vSphere 6.x. SSO provides identity management for administrators, users, and applications that interact with the VMware vSphere environment. **Active Directory** (**AD**) domains and **Open Lightweight Directory Access Protocol** (**OpenLDAP**) authentication sources can be added to provide authentication to the vCenter management components.
    - VMware Certificate Authority issues certificates for users accessing vCenter services, machines providing vCenter services, and ESXi hosts. The VMCA service is new to vSphere 6.x and is deployed with the PSC. The VMCA not only issues and manages certificates to vSphere services and components, but also acts as the **Certificate Authority** (**CA**) for these certificates. The VMCA can be used as a subordinate CA in an enterprise CA environment.

- **VMware vCenter Server**: This provides the configuration, access control, and performance monitoring of ESXi/ESX hosts and virtual machines that have been added to the inventory of the vCenter Server. In vSphere 6.x, the VMware vCenter Inventory Service, the VMware vSphere Web Client, the VMware Content Library Service, and other services not provided by the PSC, are all installed with the vCenter Server.
- **VMware vCenter Inventory Service**: This maintains application and inventory data so that inventory objects, including data centers, clusters, folders, and virtual machines, can be searched and accessed. In a vCenter 6.x deployment, the vCenter Inventory Service is installed on the vCenter Server.
- **VMware vSphere Web Client**: This allows the connections made to vCenter to manage objects in its inventory by using a web browser. Many of the new features and capabilities since vSphere Version 5.1 can only be configured and managed using the VMware vSphere Web Client. To access and configure new features in vSphere 6.x, the vSphere Web Client is required. The vSphere Web Client Server is installed with the vCenter Server.
- **vCenter Database**: vCenter Server requires a database to store configurations, logs, and performance data. The database can be an external Microsoft SQL or Oracle database server or the embedded vPostgreSQL database. An external Microsoft SQL database is only supported with a Windows vCenter deployment.

2. Identify the common dependencies required to install vCenter and the PSC:
   - **DNS**: Forward and reverse name resolution should be working properly for all systems. Ensure that systems can be resolved by the **Fully Qualified Domain Name** (**FQDN**), the short name or hostname, and the IP address.
   - **Time**: Time should be synchronized across the environment.

How it works…

Each vCenter Server component or service has a set of dependencies. The following diagram illustrates the core vCenter Server dependencies:



Core vCenter Server dependencies

As with earlier vCenter versions, the vCenter 6.7 Windows installation media includes several other tools that provide support and automation to deploy, manage, patch, and monitor the vSphere virtual environment. These tools can be installed on the same server as other vCenter Server components, or on a separate server. The tools that are included are as follows:

- **VMware vSphere Update Manager (VUM)**: This provides a central automated patch and version management for ESXi hosts and virtual appliances. VUM can be installed on the vCenter Server when running on Windows, but it must be installed on a separate Windows server when using the VCSA.
- **ESXi Dump Collector**: This collects memory dumps over the network in the event of an ESXi host encountering a critical error.
- **VMware vSphere Syslog Collector**: This enables network logging and combines the logs from multiple hosts.
- **VMware vSphere Auto Deploy**: This provides the automated deployment and configuration of ESXi hosts.

Selecting a vCenter deployment option

There are a number of deployment options available for deploying vCenter. The vCenter Server can be deployed on a dedicated physical server running a 64-bit Windows server operating system, on a virtual machine running a 64-bit Windows server guest operating system, or as a Linux-based virtual appliance. vCenter components can be installed on a single server, or the components can be installed on separate virtual or physical machines.

How to do it…

Regardless of the deployment option that's selected, the vCenter Server components must be installed and configured in a specific order, so that the service dependencies are met.

The order of installation of the vCenter Server components is as follows:

1. Deploy the VMware **Platform Services Controller** (**PSC**)
2. Deploy the vCenter Server
3. Deploy the other supporting components: VMware Update Manager, the VMware Syslog Service, ESXi Dump Collector, and so on

How it works…

Deploying the vCenter Server components on a virtual machine is a VMware recommended practice. When vCenter is deployed on a virtual machine, it is possible to take advantage of the portability and availability provided by the virtual infrastructure. One of the primary advantages of deploying vCenter components on virtual machines is that VMware **High Availability** (**HA**) can be leveraged to protect the management environment from a hardware failure or a virtual machine crash.

The **vCenter Server Appliance** (**VCSA**) is a preconfigured, Linux-based virtual machine that has been optimized to run the vCenter Server and the associated services. It includes a PostgreSQL-embedded database. A remote database connection can be configured to support larger deployments.

A limitation of the VCSA is that Microsoft SQL is not supported as a remote database. In previous versions of vCenter, VUM had to be installed on a separate Windows server. Since 6.7, VUM is installed within the VCSA, so a separate Windows server just for VUM is no longer needed.

vCenter Linked Mode creates groups of vCenter Servers that can be managed centrally. Logging in to one member of the vCenter Linked Mode group allows an administrator to view and manage the inventories of all the vCenter Servers in the group. vCenter 6.7 provides an Enhanced Linked Mode that allows for linking both VCSA and Windows vCenter Server deployments.

The PSC, vCenter Server services, and other supporting components can all be installed on a single Windows server, or each component can be installed on a separate server. Installing all of the components on a single server simplifies deployment to support a small environment. Installing each component on a separate server adds some complexity, but allows the resources for each service to be adjusted as necessary and provides flexibility for larger deployments.

Determining vCenter resource requirements

The minimum system requirements for the vCenter Server are dependent on the size of the environment that's managed by the vCenter Server. Sizing vCenter Server correctly will ensure proper operation. The size of the vCenter inventory, the number of hosts, and the number of virtual machines all have an impact on the amount of resources required. Running multiple vCenter Server components on a single server (like an embedded PSC, for example) also determines the amount of resources that will need to be allocated to the vCenter Server.

How to do it…

The following steps will help you to determine the vCenter system requirements:

1. Estimate the number of hosts and virtual machines that will be managed by the vCenter Server
2. Determine whether all of the vCenter Server components will be installed on a single server, or on separate servers
3. Size the vCenter Server to support the managed inventory

How it works…

vCenter Server 6.7 with embedded PSC resource requirements are shown in the following table and ordered by inventory size.

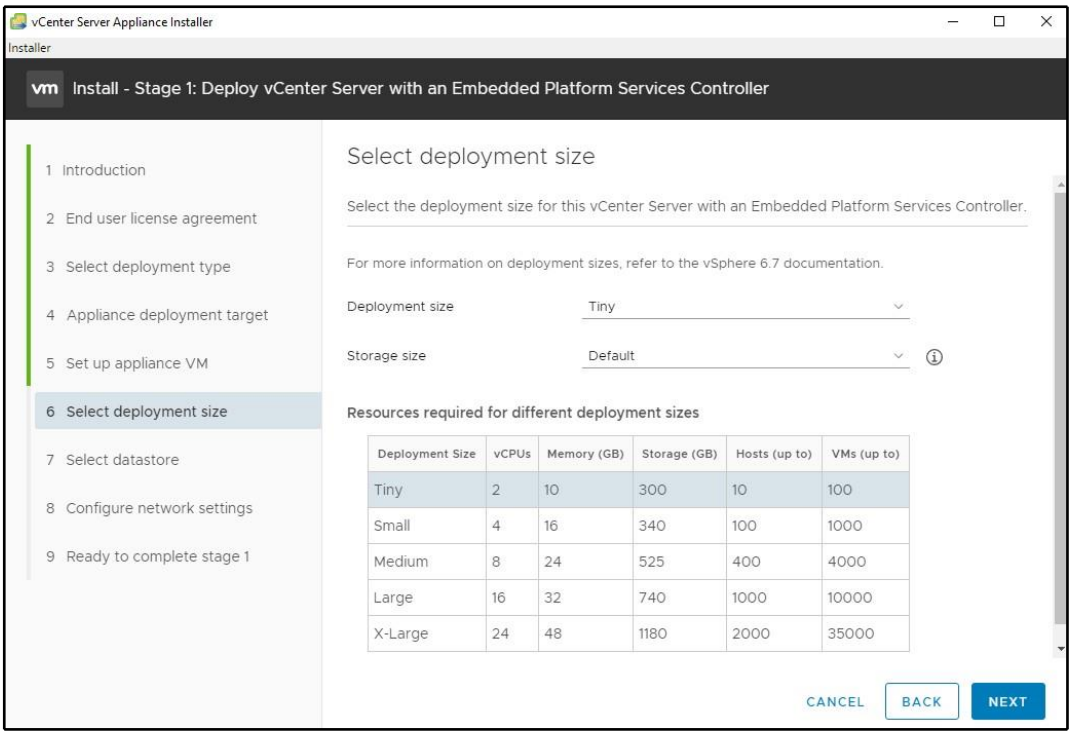
Inventory Size	Number of vCPUs	Memory
Tiny 10 hosts/100 virtual machines	2	10 GB
Small 100 hosts/1,000 virtual machines	4	16 GB
Medium 400 hosts/4,000 virtual machines	8	24 GB
Large 1,000 hosts/10,000 virtual machines	16	32 GB
X-Large 2,000 hosts/35,000 virtual machines	24	48 GB

The PSC can be installed on separate physical or virtual machines. The following table lists the minimum requirements for the PCS, if it is installed on separate physical or virtual machines:

Component	2 GHz CPU cores	Memory
PSC	2	2 GB

If the databases are installed on the same machine, additional CPU, memory, and disk resources will be necessary.

In vSphere 6.7, the VCSA sizing requirements mirror those of the vCenter on Windows Server, based on the size of the managed environment. When you are deploying the VCSA, the inventory size is selected (**Tiny**, **Small**, **Medium**, **Large**, or **X-Large**), and the VCSA appliance is configured with the required resources, as shown in the following screenshot:



Choosing the VCSA deployment size

There's more…

VMware and third-party plugins and applications may require their own resources. For example, if VUM is installed on the same machine as other vCenter components, the CPU, memory, and disk capacity requirements will need to be adjusted to support the additional resources required.

Selecting a database for the vCenter deployment

The vCenter Server requires a supported database to be deployed to store virtual infrastructure configuration information, logging, and performance statistics. The VCSA and the vCenter Server on Windows both support an embedded or external database.

How to do it…

Perform the following steps to select a database for the vCenter deployment:

1. Estimate the number of hosts and virtual machines that will be managed by the vCenter Server
2. Choose a supported database platform that is suitable to support the vCenter inventory

How it works…

The database stores configuration and performance information. The three database deployment options are as follows:

- Use the embedded vPostgreSQL database on the VCSA or the bundled vPostgreSQL database if installing vCenter Server on Windows
- Install a full database server locally on the same server as the vCenter Server components
- Connect to a database hosted on a remote server

The embedded database included with the VCSA can support an inventory of up to 2,000 hosts and 35,000 virtual machines, which makes it a suitable option, even for very large deployments. The embedded vPostgreSQL on Windows, which can be deployed as a part of the vCenter Server Windows installation, is intended for smaller deployments of up to 20 hosts and 20 virtual machines. If a Windows vCenter Server is deployed using the embedded databases where the inventory is expecting growth beyond 20 hosts and 200 virtual machines, a different supported database option should be selected.

The Microsoft SQL Express Database is no longer supported in vCenter 6.x. When upgrading a vCenter 5.x server that was deployed using the embedded Microsoft SQL Express Database, the vCenter database will be migrated to the vPostgreSQL database as a part of the upgrade process.

Some reasons to use the embedded vPostgreSQL database when deploying a Windows vCenter Server are as follows:

- A small environment of fewer than 20 hosts and 200 virtual machines
- Easy installation and configuration
- Free! no need to license a separate database server software

Databases are created as a part of the installation process when you are using the bundled vPostgreSQL and vCenter Server. If a full installation of a database server is used, these databases (and the ODBC connections required for them) must be manually created prior to the installation.

Installing a full Microsoft SQL or Oracle database locally (on the same Windows server as the vCenter components) is supported, but this increases the amount of resources that are necessary for the vCenter Server. Additional resources may be required, depending on the size of the vCenter inventory. Hosting the database locally (on the same server) is fully supported, and this can provide faster access, since the access to the database does not rely on network resources.

A full installation of Microsoft SQL or Oracle can also be performed on a separate physical or virtual machine. The vCenter components access the databases hosted on the remote database server. The creation of the databases and the configuration of the vCenter components is the same as with a full database installation on the same server as vCenter. Accessing the databases requires network resources; because of this, network congestion or a network outage can affect the accessibility to the databases.

Some reasons to choose a remotely installed database are as follows:

- Leverage an existing database server that's already available in the environment
- For a separation of roles; database administrators are responsible for administering the database servers, while virtual administrators are responsible for administering the virtual environment
- High availability can be provided to the databases by using Microsoft or Oracle clustering
- It reduces the amount of resources that need to be allocated to the vCenter Server

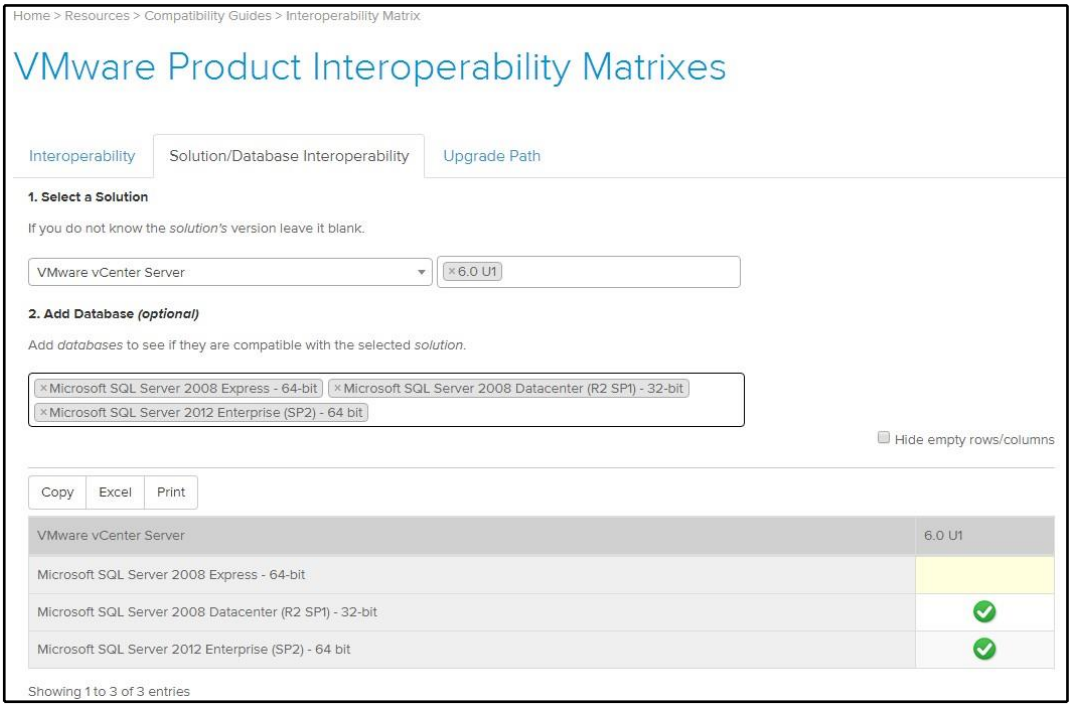
Determining database interoperability

VMware provides an online Interoperability Matrix to make it easy to determine which database versions are compatible and supported with which versions of VMware products.

How to do it…

To determine database interoperability with VMware products, perform the following steps:

1. Visit: `https://www.vmware.com/resources/compatibility/sim/interop_matrix.php`
2. Select the **Solution/Database Interoperability** tab
3. In the **Select a Solution** option, select **VMware vCenter Server** and a **Version** from the respective dropdown boxes
4. **Add Database** versions by using the **Database** dropdown box. You can add multiple database versions
5. The database's compatibility with the selected product will be displayed in the table, as shown in the following screenshot:



Example of using the VMware Product Interoperability Matrix

How it works…

Verifying database product interoperability ensures the supportability of the database and the version that has been selected for use with a specific VMware product. The VMware Product Interoperability matrices are regularly updated by VMware when new databases or VMware product versions are released.

Database and product interoperability should be checked for new installations, and this should be done prior to upgrading VMware products or applying service packs to database servers.

There's more…

The Interoperability Matrix can be used to determine database operability for all supported VMware products and solutions. It can also be used to determine supported upgrade paths and interoperability between different VMware solutions.

Choosing a vCenter deployment topology

The deployment topology for a vCenter 6.7 deployment is dependent on the size of the environment, the number of vCenters that will be deployed, the number of sites, and the availability required.

How to do it…

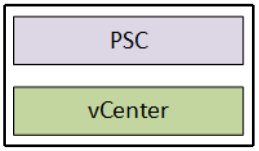
To determine the vCenter deployment topology for a vCenter 6.7 deployment, follow these steps:

1. Identify the use cases for each vCenter deployment topology. Factors to consider include the following:
    - The size of the environment
    - The number of vCenters
    - The number of sites
2. Select the vCenter deployment topology based on the environment requirements.

How it works…

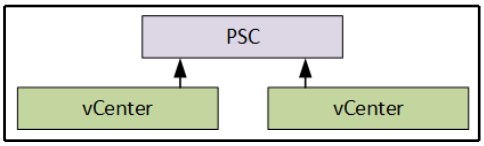
vSphere 6 supports up to 10 vCenters linked together in Enhanced Linked Mode, and up to eight PSCs to support the environment. vCenters and PSCs can be deployed on the same site, or across multiple sites. In a small environment with a single vCenter Server, the PSC and vCenter Server can be combined on a single appliance.

The embedded deployment is the topology with the least complexity. The embedded deployment topology is suitable for a small, single-site, single-vCenter environment. In this topology, the PSC and vCenter Server are installed on the same virtual or physical machine. This is represented in the following diagram:



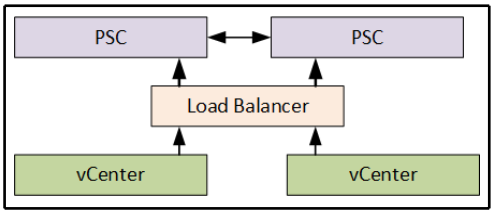
Representation of vCenter with embedded PSC

Multiple vCenter Servers can be deployed with a single external PSC. This deployment topology is used for a small single site with multiple vCenter Servers. These vCenter Servers can be VCSA or Windows, or a mix of both. This enables single-screen management of the environment with Enhanced Linked Mode. This topology is represented in the following diagram:



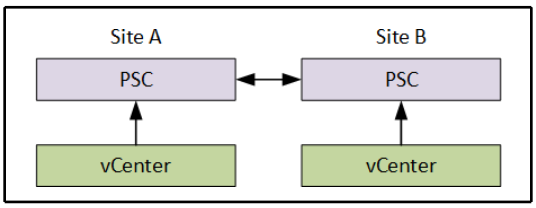
Logical view of multiple vCenters pointing to a single PSC

Multiple PSCs can be deployed to provide high availability to the PSC services. A single vCenter or multiple vCenters can access the PSCs within the same site through a load balancer. There can be up to four PSCs per site, behind a load balancer. The following diagram is a representation of a topology where multiple PSCs are deployed for high availability:



Logical view of a load-balanced PSC

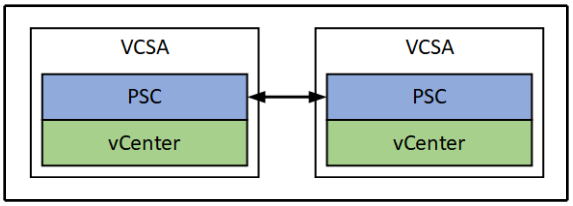
In a multi-site topology, the PSC is deployed at each site. This provides replication of the PSC between sites, and it also enables Enhanced Linked Mode between the vCenters at both sites. The following diagram represents a multi-site deployment topology:



Logical representation of a multi-site PSC topology and Enhanced Linked Mode

Choose a deployment topology that supports the size and requirements of the environment. If Windows-based vCenters are used, the PSC should not be deployed embedded with a vCenter. VMware does not support replication between embedded PSCs for Windows vCenter Servers or using an embedded PSC to provide services to external vCenter Servers.

The latest supported and recommended topology is the VCSA with embedded PSC, as shown in the following diagram, which supports Enhanced Linked Mode for up to 15 VCSAs:



Latest recommended topology with VCSA and embedded PSCs

Designing for management availability

The availability of the management functions of an environment becomes more critical in environments like those that support virtual desktops or self-service provisioning. In these environments, if the vCenter Server is unavailable, so is the ability to provide the provisioning of services.

If the environment does not provide these types of services, the ability to manage the environment, especially during a failure or disaster, is also critical. How can you troubleshoot an issue with a virtual machine (or a group of virtual machines) if the primary tool that is used to manage the environment is unavailable?

How to do it…

To properly design for management availability, follow these steps:

1. Identify the management environment dependencies, as follows:
    - Infrastructure dependencies, including storage, networking, and host hardware
    - Service dependencies, including DNS, DHCP, and Active Directory
    - VMware product dependencies, including the PSC, the vCenter Server, and other supporting components
2. Identify the potential single points of failure in the management environment.
3. Create a management design that ensures the high availability of the management components.

How it works…

When designing the management network, single points of failure should be minimized. Redundant network connections and multiple network interfaces connected to separate physical switches should be configured to provide connectivity.

The storage that hosts the management components should be configured to support the capacity and performance of the management components. The storage should also be configured to be highly available so that a disk or path failure does not interrupt management operations.

In environments where the vCenter Server provides provisioning, such as a virtual desktop or self-service cloud environments, vCenter uptime is critical.

If the vCenter Server is running on a virtual machine, it can be protected with HA. If the host that vCenter is running on or the operating system crashes, the vCenter Server is restarted on a surviving host. There will be some downtime associated with the failure, but when they are designed correctly, the vCenter Server services will be quickly restored.

Sufficient resources should be dedicated to the vCenter Server and its components. We discussed the correct sizing of the vCenter Server earlier in this chapter. Sizing vCenter correctly and reserving resources ensures not only the performance, but also the availability. If a virtual machine is running on the same host as the vCenter Server or one of its components and it consumes too many of the host resources, it may impact the performance and availability of the vCenter Server services. Applying resource reservations to the vCenter Server will prevent resource contention.

Another means of preventing resource contention is to design a separate cluster to host the management components. Management cluster design will be discussed in the *Designing a separate management cluster* recipe.

Designing a separate management cluster

The management components of a virtual environment can be resource intensive. If you are running vCenter and its dependencies as virtual machines in the same cluster as the cluster managed by the vCenter server, the resources required by the management infrastructure must be factored into the capacity calculations of the logical design. Creating a separate management cluster separates the resources required by the vCenter and other management components from the resources required by the applications hosted in the virtual infrastructure.

While a separate management cluster can be beneficial for capacity planning, it can increases the costs associated with building a vSphere environment. A separate management cluster is not required, but it may be a good idea if you need to separate management components from other workloads.

How to do it…

Refer the following steps to design a management cluster:

1. Identify management cluster best practices as follows:

1. Having the CPU and memory resources to support management applications
2. Having multiple network interfaces and multiple physical network switches to minimize the single points of failure in the management network
3. Having multiple paths to the storage to minimize the single points of failure in the storage network
4. Having storage that's designed to support both the capacity and the performance required for management applications

2. Correctly size the management cluster and identify the services that will be hosted in the cluster. The following questions also need to be answered to size the management cluster.
    1. What is the deployment topology of the vCenter Server environment?
    2. How many PSCs and vCenter Servers will be deployed to support the environment?
    3. Will the cluster also provide the resources needed for the vCenter databases?
    4. What about other management tools, such as vCenter Operations Manager, vCenter Log Insight, or other third-party management tools?

How it works…

The design of a management cluster follows the same process as designing a cluster hosting the production applications. Requirements need to be identified, and a logical design process for storage, networking, and computing resources must be followed. The functional requirements for the management network will likely include high availability, minimizing single points of failure, and quickly recovering failed components.

There's more…

Affinity rules can be used to keep virtual machines together. For example, having the virtual machine running the vCenter Server and the virtual machine running the vCenter Server database on the same host reduces the load on the physical network, since all communication between the two servers never leaves the internal host network.

Anti-affinity rules can also be used to separate virtual machines across hosts or groups of hosts. In an environment where multiple PSCs are deployed to provide high availability, separating the PSCs by using anti-affinity rules will ensure that a single host failure does not impact the services provided by the PSCs.

If you are hosting vCenter in the same cluster as other virtual machine workloads, affinity and anti-affinity rules can be used to keep the vCenter Server running on specific hosts, creating a pseudo-management cluster, so that it can easily be located in the event of the vCenter Server becoming unavailable. If such rules are used for vCenter, consider using should rules and not must rules to allow HA to violate the rules, if required, during an HA event to ensure vCenter gets restarted.

Configuring vCenter mail, SNMP, and alarms

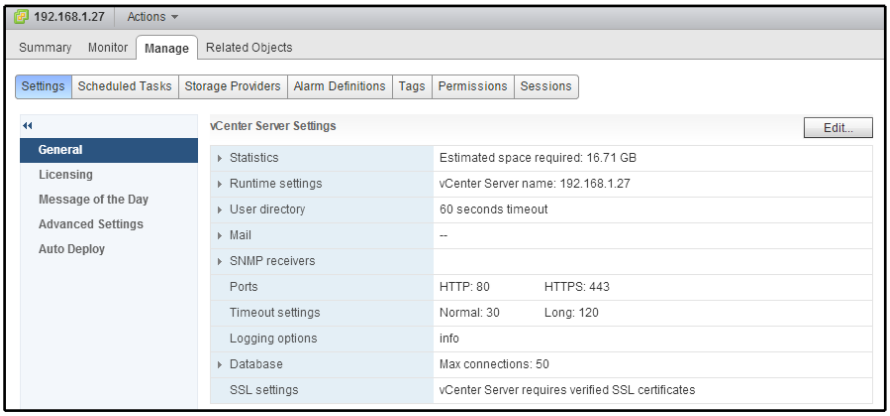
Alarms can be used to notify an administrator of issues (or potential issues) in a vSphere environment. This notification allows an administrator to take corrective actions. Alarms can be configured to send email notifications and/or SMNP traps when conditions are triggered. Alarm definitions contain a trigger and an action. Triggers include issues like hardware failures, or states like increased CPU or memory utilization.

Properly designing alarm notifications can ensure successful ongoing operations in a vSphere environment.

How to do it…

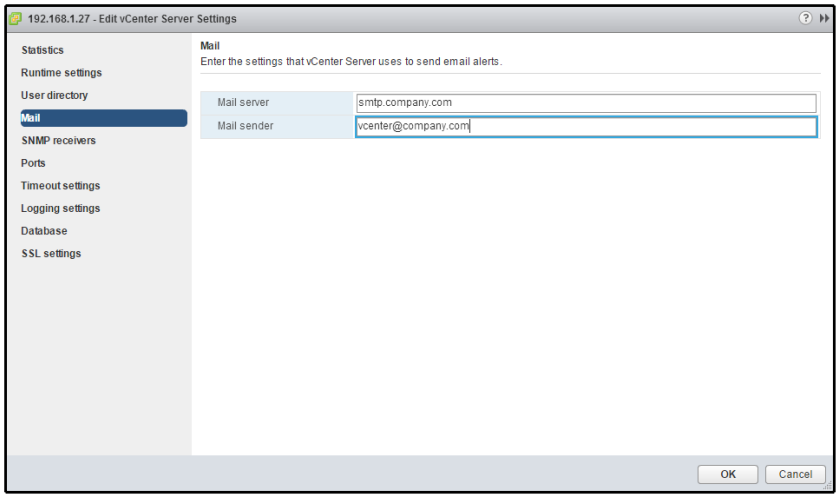
The following steps will configure the **Mail** and **SNMP** settings for a vCenter Server, and will configure a defined alarm to send an email or SNMP notification:

1. Using the vSphere Web Client, access **Manage** | **Settings** | **General** for the vCenter Server, as shown in the following screenshot:



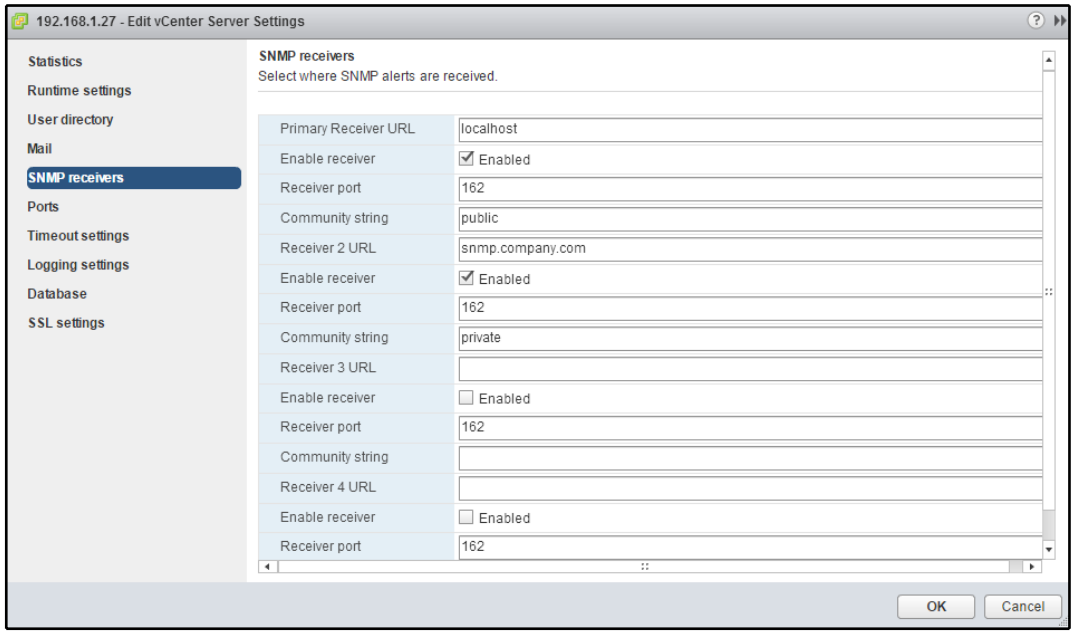
View General Settings in vCenter using the vSphere Web Client

2. Select **Edit** and **Mail**. Provide the **Mail server** FQDN or IP address and the **Mail sender** address. The vCenter **Mail** configuration is shown in the following screenshot:



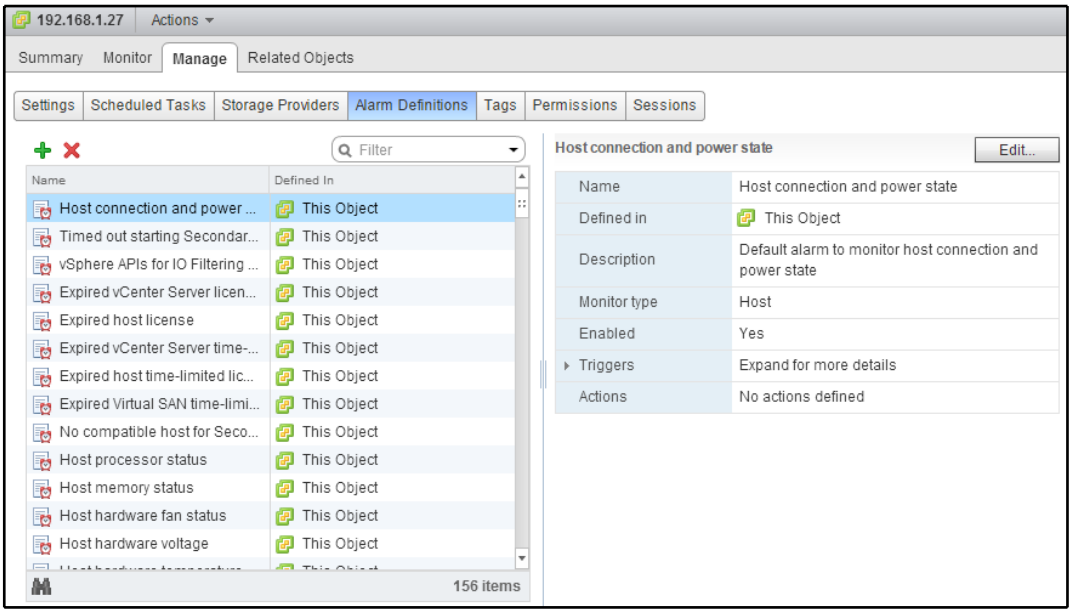
Configure mail in vCenter using the vSphere Web Client

3. To configure SNMP, select the **SNMP receivers** and configure the **Receiver URL**, **Receiver port**, and **Community string**. Select the checkbox to **Enable** the receiver, as shown in the following screenshot:



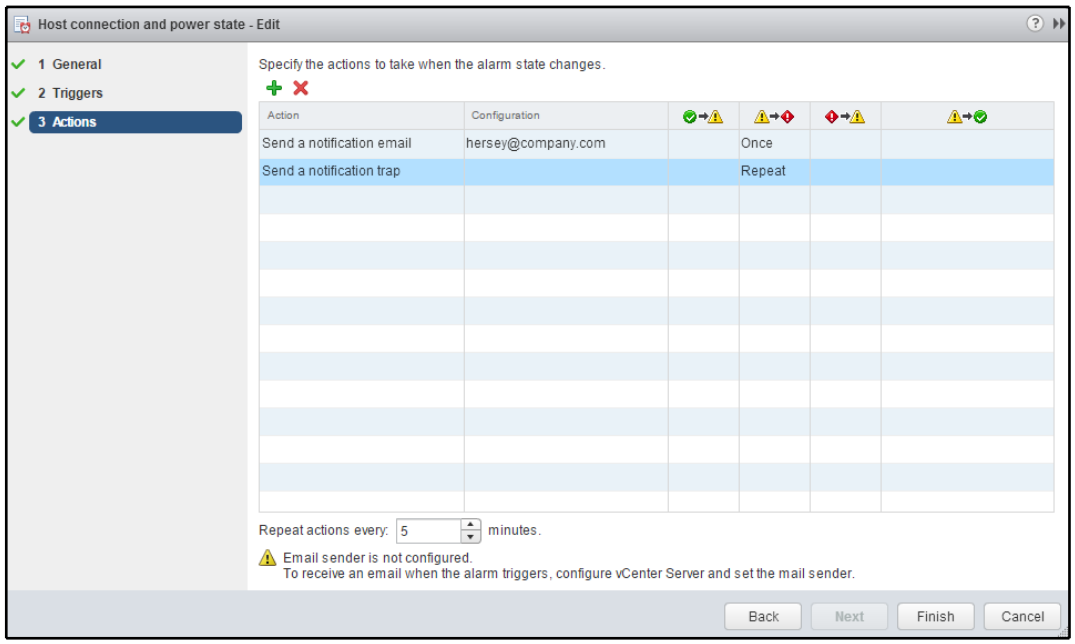
Configure SNMP in vCenter using the vSphere Web Client

4. To configure an alarm, select **Manage** | **Alarm Definitions**. Select the alarm and click on the **Edit** button:



Configure alarms in vCenter using the vSphere Web Client

5. The **Send a notification email** and **Send a notification trap** actions can be configured in the alarm **Actions** section. When configuring the **Send a notification email** action, the email address to send the alert to is configured in the **Configuration** field. Multiple actions can be configured for an alarm. **Actions** can be configured to be executed **Once**, or they can **Repeat** over a configured period of time, as shown in the following screenshot:



Configure alarm actions in vCenter using the vSphere Web Client

How it works…

For the **Send notification email** alarm action to work, the vCenter **Mail** settings must be configured with both the **Mail server** and the **Mail sender** address. The **Mail sender** address is the mail from address included on the vCenter alarm notification. The **Mail server** is the server that the SMTP mail will be relayed through. The **Mail server** that's specified must be configured to accept and relay mail from the vCenter Server.

Configured SNMP receivers will receive notifications from alarms that have been configured with the **Send a notification trap** action. The SNMP configuration includes the receiver URL, the receiver port, and the receiver community string. Multiple SNMP receivers can be configured and enabled.

There is an extensive list of preconfigured **Alarm Definitions**. Custom alarm definitions can also be created. By default, the **Send a notification email** action is not configured for any of the preconfigured definitions. When an alarm is triggered and the **Send a notification email** action is configured, an email will be sent to the email address (or addresses) in the configuration for the action.

Alarms actions can be configured to send a single notification, or to send a repeated notification. Repeated notifications can be configured to repeat over different intervals while the alarm state is triggered.

Using Enhanced Linked Mode

Enhanced Linked Mode allows for multiple vCenter Servers to be connected together to provide a single point of management. Enhanced Linked Mode enables the ability to view, search, and manage multiple vCenter Servers, and provides the replication of roles, permissions, licenses, and policies between vCenter Servers. This simplifies the management of large environments, with multiple vCenter Servers deployed in the same site or across multiple sites. vCenter 6.x supports linking vCenter Servers that have been deployed as VCSAs and as Windows Servers with external PSCs. Recall from earlier in this chapter that Enhanced Linked Mode is only supported between VCSAs when you are using embedded PSCs.

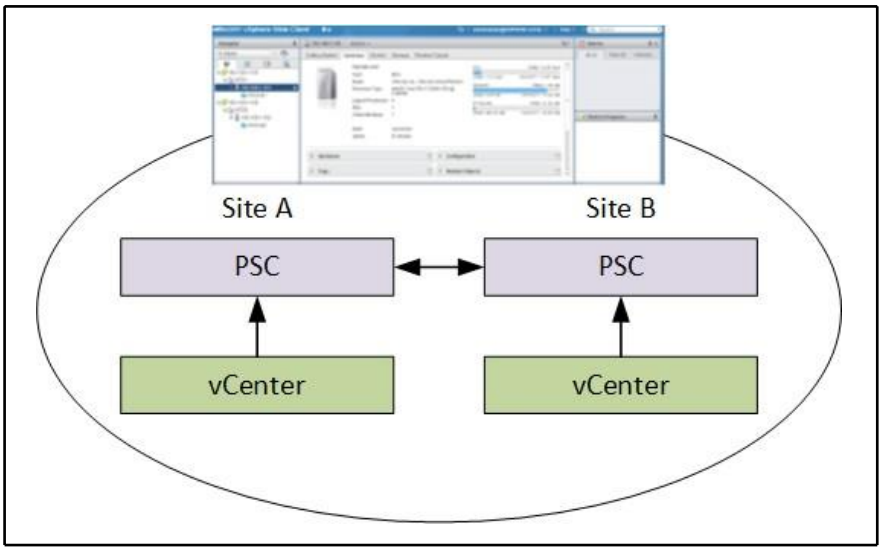
How to do it…

To enable Enhanced Linked Mode, follow these steps:

1. Ensure that the Enhanced Linked Mode requirements are met:
    - Ensure that all PSCs are in the same vSphere single sign-on domain
2. Deploy PSCs and vCenter Servers in a supported deployment topology

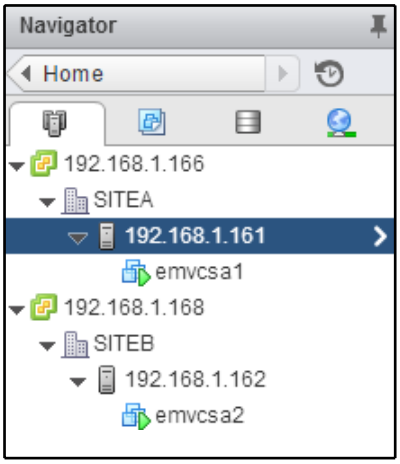
How it works…

Enhanced Linked Mode enables a single point of management across all vCenter Servers in the same vSphere single sign-on domain. This allows an administrator to easily manage the different environments (for example, a virtual server environment and a virtual desktop environment) across multiple sites:



Enhanced Linked Mode in the vSphere Web Client

Once enabled, the inventories of all vCenters in the same single sign-on domain will be linked in Enhanced Linked Mode. The management of these vCenters will then be accessible from a single Web Client interface, as follows:



Hosts and Clusters view of Enhanced Linked Mode in the vSphere Web Client

Using the VMware Product Interoperability Matrix

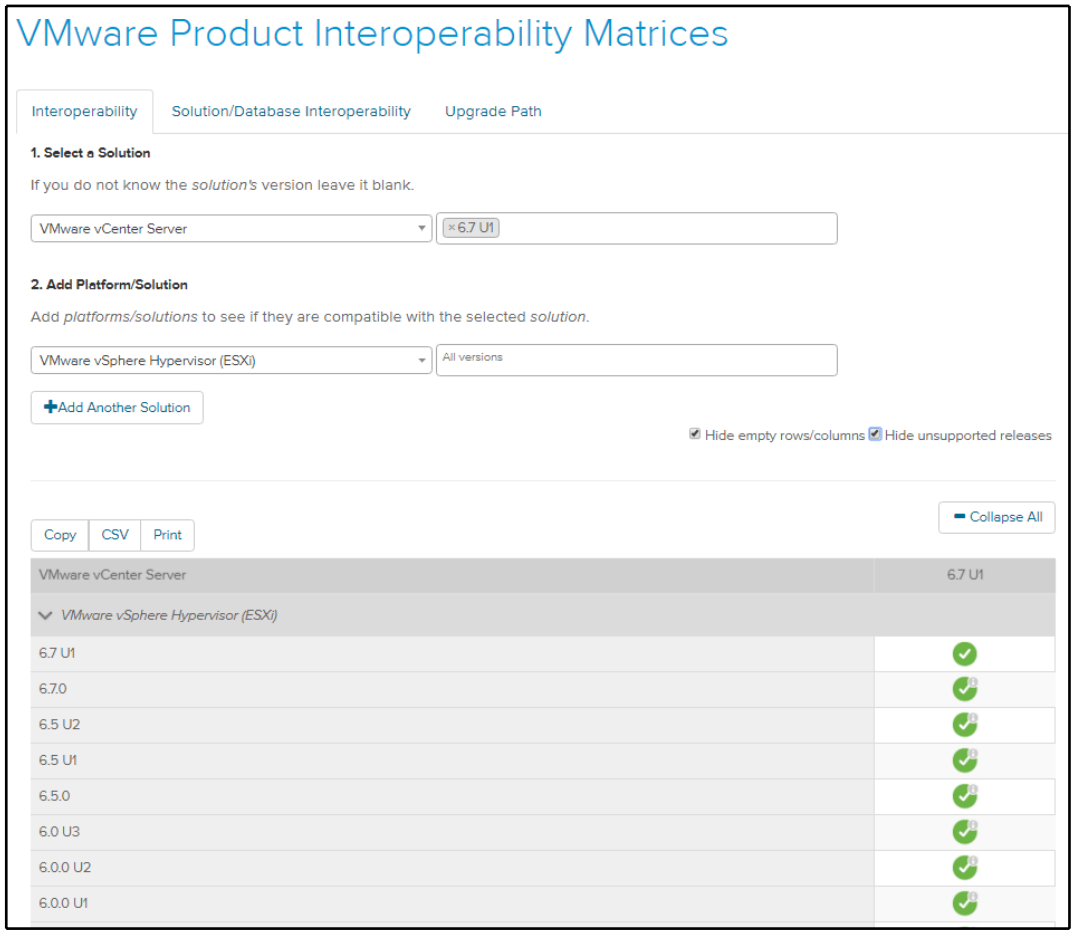
The VMware Product Interoperability Matrix allows you to ensure compatibility between VMware products. It is important to check for compatibility before deploying or upgrading the components of a vSphere environment to ensure support and interoperability between product versions.

How to do it…

Perform the following steps to validate the interoperability of VMware products in a vSphere deployment:

1. Visit `https://www.vmware.com/resources/compatibility/sim/interop_matrix.php`
2. Select the **Interoperability** tab
3. Under the **Select a Solution** option, select the VMware product and version from the respective dropdown boxes
4. Select Add **Platform/Solution** by using the dropdown box. You can add multiple solutions and versions

5. The interoperability with the selected products and solutions will be displayed in the table, as shown in the following screenshot:



Example use of the VMware Product Interoperability Matrix

How it works…

Verifying product interoperability ensures supportability and interoperability between different VMware products and versions. The VMware Product Interoperability matrices are regularly updated by VMware when new products and versions are released.

Product interoperability should be checked for new installations, and this should be done prior to upgrading VMware products.

There's more…

In many environments, third-party products for monitoring, automation, and protection are used. In a new vSphere design, there will likely be requirements or constraints for integration with these third-party components. It is important to verify interoperability with these products before deploying or upgrading a vSphere environment. The VMware Product Interoperability matrices only include VMware products. Third-party product interoperability will need to be verified with the product vendors.

Backing up the vCenter Server components

vCenter and its components have become a critical piece of the virtual infrastructure. The vCenter Server is no longer just a management interface. Provisioning, protection, and the overall availability of the environment rely on vCenter Server availabilty.

To recover the vCenter Server components in the event of an outage that results in data loss or data corruption, it is necessary to make backups of the databases and the vCenter Server configurations. The PSC and vCenter Server each have specific configuration information that should be backed up.

The frequency of backups depends on the **Recovery Point Objective** (**RPO**) that has been defined for the management environment. The time to recover the vCenter Server, or the **Recovery Time Objective** (**RTO**), is also a critical piece of designing a vCenter backup strategy. The RPO defines the maximum period of data loss that can be tolerated as a result of an outage.

If the RPO has been determined to be four hours, this means that backups should occur at least every four hours. The RTO determines how quickly the vCenter must be available after an outage.

How to do it…

Follow this process to design a backup and recovery strategy for the vCenter Server environment:

1. Determine the RPO and RTO requirements for the vCenter Server and the supporting components
2. Develop a backup and recovery strategy that ensures the RPO and RTO requirements are met

How it works…

VMware recommends creating full virtual machine backups of the PSC and vCenter Server when these components are running in virtual machines. There are many third-party backup software products that can also be used to make full virtual machine backups. This allows the virtual machines to be restored quickly in the event of a failure.

If the PSC or vCenter Server is running as a physical machine, a third-party backup application can be used to take a full bare-metal backup. It is important to realize that this type of backup will take longer to restore, impacting the RTO.

File-based backups are also supported for the VCSA. File-based backups are configured through the **Virtual Appliance Management Interface** (**VAMI**) and sent to remote servers via FTP, FTPS, HTTP, or SCP.

Configuration and performance data is stored in the vCenter Server database. How backups are done depends on the database software that is used to host the database. For example, if the database is a Microsoft SQL database, a backup can be performed on demand in the SQL Management Studio, or as a scheduled SQL job. Third-party backup tools can also be used to back up the vCenter databases.

If the vCenter is using the embedded vPostgreSQL database on either a Windows vCenter Server or the VCSA, it can be backed up using a script from the VMware KB Article 2091961, located at `http://kb.vmware.com/kb/2091961`. There are separate scripts to support a Windows or VCSA vCenter deployment.

The vCenter Server database should be backed up regularly, based on the RPO that has been defined for the management components.

Planning vCenter HA to increase vCenter availability

VCHA is a feature that uses a three-node cluster to protect the vCenter Server from hardware, operating system, or application failures. The three nodes are referred to as active, passive, and witness. VCHA only supports VCSA deployments, not vCenter on Windows, and both embedded and external PSCs are supported. It's important to note that if used with external PSCs, VCHA is not protecting the PSCs—only the vCenter Server itself. Load balanced PSCs would be needed to provide high availability to external PSCs. Keep in mind that it likely doesn't make sense to use vCenter HA if you're not also using load-balanced PSCs, since the idea is to create a highly available management plane.

VCHA is useful when you want to increase vCenter's uptime and you don't necessarily want to only rely on vSphere HA to protect against a host failure. VCHA also protects against service failures. When using an embedded PSC, VCHA will not only monitor vCenter services for failures, but for PSC service failures as well. If a service fails, the passive node becomes the active node.

There are a few simple system requirements for VCHA. Only one vCenter Server license is required, and the vCenter deployment type must not be tiny due to the additional resources that are required.

There are two options to configure VCHA: **Automatic** and **Manual**. If you choose an **Automatic** configuration, the configuration wizard will clone the existing vCenter Server and configure the cluster networking. If you choose the **Manual** method, you must clone the vCenter Server and configure the cluster networking yourself. As of vCenter 6.7 Update 1, however, only the **Automatic** method is available.

After several years of not having a high availability option for vCenter (after VMware deprecated the vCenter Heartbeat product), VCHA provides an internally developed option for those environments that require maximum uptime for their vCenter Server.

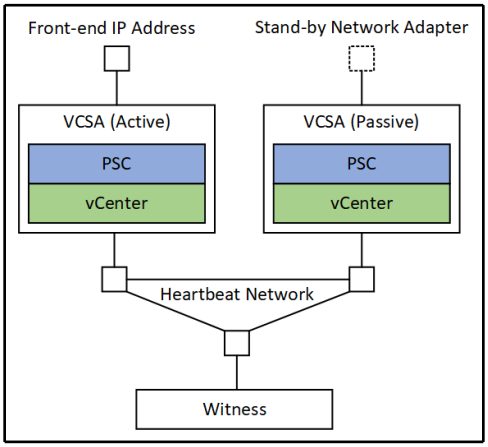
How to do it…

To plan and implement vCenter HA, use the following process:

1. Ensure that the VCHA system requirements are met
2. Determine the deployment topology for vCenter and the PSC
3. Choose the **Automatic** or **Manual** configuration method

How it works…

VCHA relies on a heartbeat network and quorum between the three nodes to avoid a split-brain scenario. The active node is the only node with a frontend or production IP address. The passive and witness nodes only have active IP addresses on the heartbeat network. This architecture is shown in the following diagram, with the embedded PSC deployment topology:



vCenter HA networking overview

There are several failure scenarios that will result in a failover from the active node to the passive node:

- vCenter service failure (or PSC service failure, if an embedded PSC is used)
- VCSA operating system failure
- Entire VM crashes
- Underlying ESXi host crashes due to hardware failure or hypervisor crashes
- Active node isolation on the heartbeat network

There are also several failure scenarios that won't cause a vCenter HA failover from the active node to the passive node:

- Passive node VM failure
- Witness node VM failure
- Frontend network interface failure on the active node

- External PSC failure
- vSphere Client service failure:
    - Services that rely on APIs stay available during this type of failure

Upgrading vCenter Server

Today, most environments will already contain at least some virtualization. A vSphere design will likely involve upgrading an existing environment to enable new features to meet new requirements for availability, security, performance, and manageability.

The management environment for vSphere has become more complex. The vCenter Server and its components have become a critical part of the environment. In the virtualized data center, the vCenter Server is no longer just a management interface; it also provides provision, availability, security, and other services. Other vSphere and third-party components require vCenter Server to operate correctly. Because of this, upgrading a vCenter Server must be planned correctly.

How to do it…

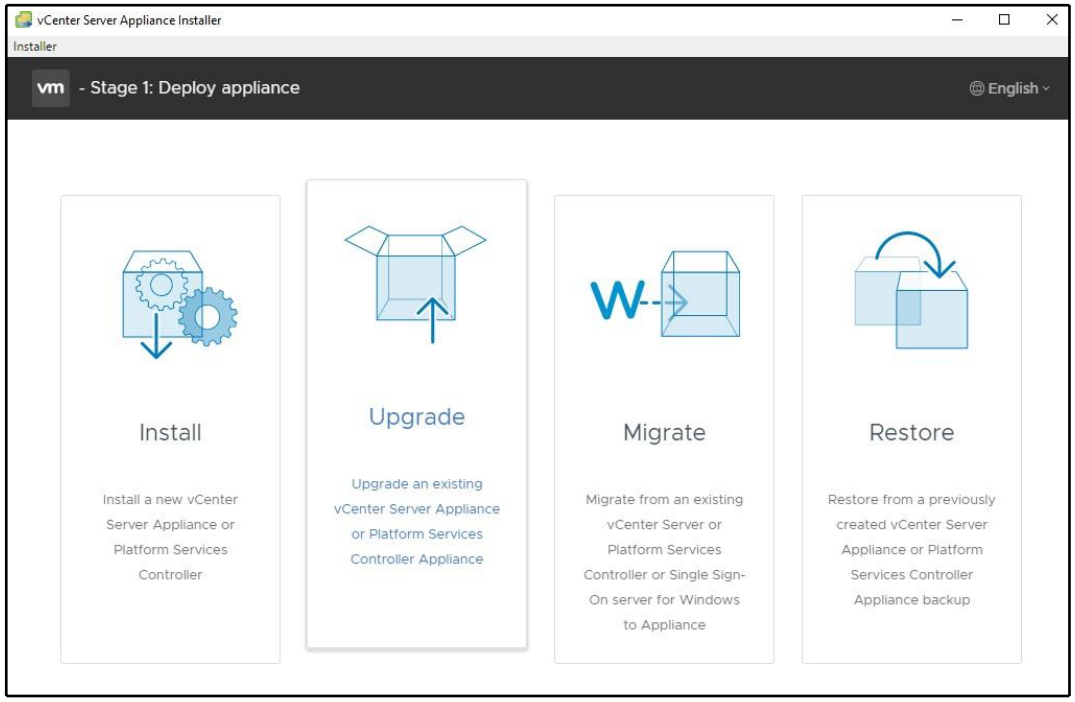
Follow this high-level process to upgrade a vCenter Server:

1. Identify the products and services that depend on vCenter Server, and those that vCenter Server depends on.
2. Verify product interoperability for all of the components and the upgraded vCenter version by using the VMware Product Interoperability matrices. Remember to also validate the compatibility with the third-party products integrated with vCenter.
3. Verify database support for the upgrade version by using the VMware Product Interoperability matrices.
4. Determine the proper upgrade path for upgrading VMware products dependent on vCenter by using the VMware Product Interoperability matrices.
5. Determine the upgrade order to ensure the interoperability of all components.
6. Upgrade vCenter and its supporting components.

How it works…

It is important to validate the support and compatibility of all vCenter Server dependencies before upgrading the vCenter Server. This is the most important process. Secondly, determining the correct upgrade order will ensure that compatibility and interoperability are maintained throughout the upgrade process.

Once the dependencies and interoperability are validated, the upgrade order for components has been determined, and the supporting components have been upgraded to ensure interoperability, the process of upgrading the vCenter Server itself will be a simple process. The Windows installer for vCenter Server on Windows and the VCSA installer both include upgrade installers to upgrade previous versions of vCenter. The following screenshot shows the VCSA installer with the **Upgrade** option:



VCSA Installer

To upgrade a Windows vCenter Server, simply run the installer from the installation media. The installer will detect the previous version of vCenter Server and perform an in-place upgrade. This means the existing application will be updated on the existing server. This is different than the Blue-Green upgrade model used for the VCSA.

When you are upgrading an existing vCenter Server environment, consider the following:

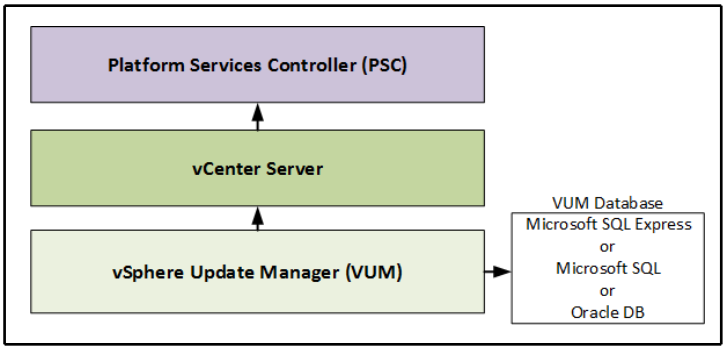
- Upgrading a Windows vCenter Server that was deployed using the **Simple Install** will upgrade the vCenter Server with an embedded PSC
- Upgrading a VCSA deployed with embedded SSO will upgrade the VCSA with the embedded PSC
- If Microsoft SQL Express was used for the vCenter deployment, the vCenter database will be migrated to the embedded vPostgreSQL database
- A vCenter Server cannot be downgraded after the upgrade. Back up the vCenter Server databases and other supporting components, in the event that you need to revert back to the previous version after the upgrade

Designing a vSphere Update Manager Deployment

VMware regularly releases patches and updates to provide bug fixes so that it can address security vulnerabilities or add new features. Regularly patching an environment is important to the security and stability of the environment.

VMware **vSphere Update Manager** (**VUM**) is an optional vCenter component when it is installed on Windows, and it provides the patching and upgrading of ESXi hosts, VMware Tools, and VMware Guest Hardware. VUM ensures that compliance is maintained through patch and upgrade baselines. VUM also allows for the remediation of hosts or virtual machines that are not in compliance with the configured baselines.

VUM must be deployed on a Windows server, and it requires a supported database, either embedded or external. The VUM architecture is shown in the following diagram:



vSphere Update Manager architecture

There is a one-to-one relationship between VUM and vCenter Servers. VUM 6.x is fully integrated into the vSphere Web Client.

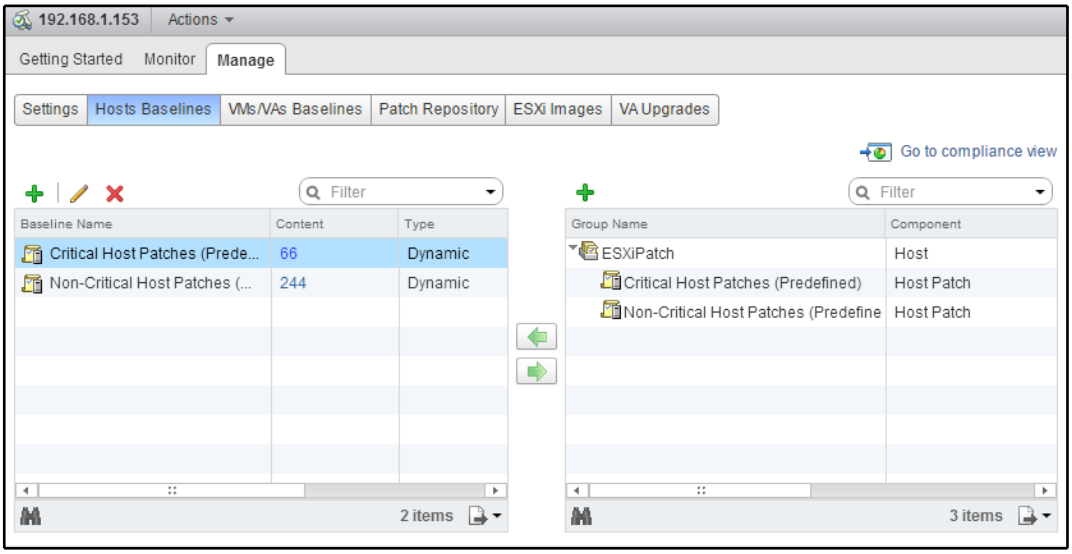
VUM is always installed in an embedded fashion with the VCSA.

How to do it…

To deploy VUM in a vSphere environment, perform the following steps:

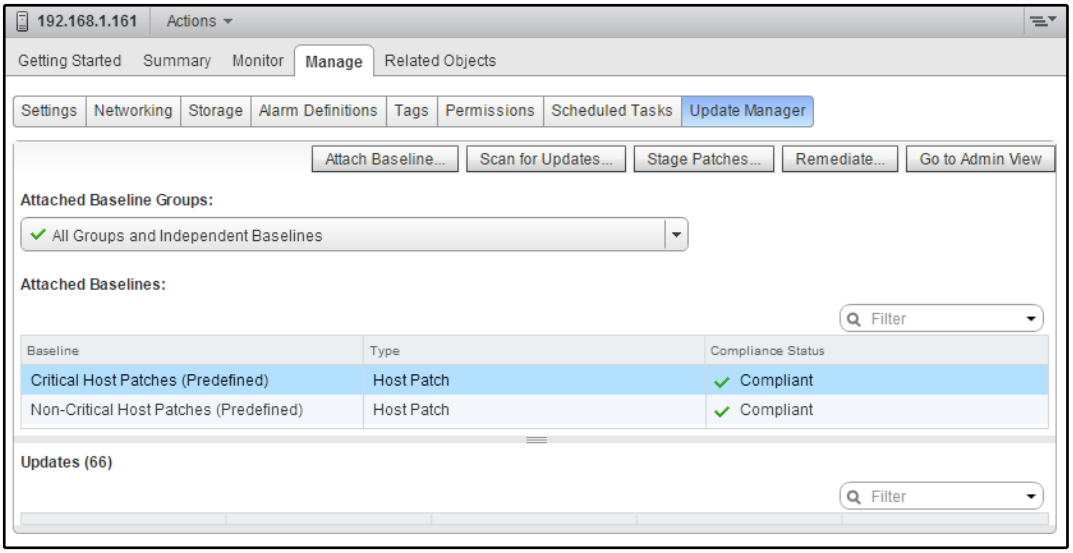
1. Verify product and database interoperability by using the VMware Product Interoperability matrices.
2. Determine the location and type of database to host the VUM database.
3. Allocate the required compute and storage resources to support the VUM server.
4. Run the vSphere Update Manager Server installation on the Windows server selected for VUM.

5. Once it's deployed, use the vSphere Web Client to create baselines and attach hosts and virtual machines to these baselines. The following screenshot provides an example of critical and non-critical host patch baselines associated with a group of hosts:



Critical and non-critical host patch baselines

6. Scan for updates and verify compliance with the attached baselines. The following screenshot displays a host in compliance with the attached critical and non-critical patch baselines:



Compliance example in VUM

7. Remediate the hosts or VMs that are not in compliance.

How it works…

VUM supports an embedded or external database. Microsoft SQL Express is included with the VUM installation media. The embedded Microsoft SQL Express database is suitable for small deployments of five hosts and 50 virtual machines. Larger deployments require a Microsoft SQL or Oracle DB, which can be installed on the same server or an external one.

VUM cannot be deployed on the same server as the VCSA. VUM can be installed on the same server as a Windows vCenter Server, as long as sufficient resources are allocated. The following table lists the minimum requirements for VUM:

Component	**vCPUs**	**Memory**
VMware Update Manager (**VUM**)	2	2 GB

The disk space required to support VUM will depend on the size of the environment. VMware provides a VUM Sizing Estimator for vSphere 6.7, which can be downloaded from `https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-update-manager-documentation-671.zip`

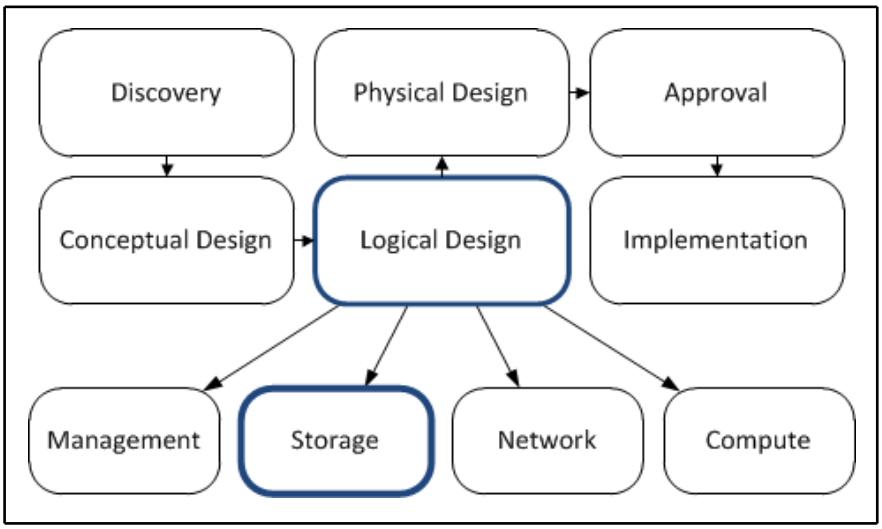
Patch and upgrade baselines contain patches (or groups of patches). These baselines can be fixed or dynamic. Critical and non-critical patch baselines are included by default. These are dynamic baselines that are regularly updated. Baselines can be attached to a virtual machine, a group of virtual machines, a host, a group of hosts, a cluster, or a data center. Hosts, clusters, and data centers can be scanned against the attached patch baseline, and then remediated.

There's more…

The default preconfigured dynamic patch baselines poll an external, internet-accessible repository for updates and to download the updates that are required for remediation. For vSphere environments without access to the internet, the **Update Manager Download Service** (**UMDS**) can be used to download the patches and updates and then export the updates and patch information to a repository that's accessible by the isolated network.

5
vSphere Storage Design

Storage is an essential component of vSphere design and provides the foundation for the vSphere environment. A solid storage design that addresses capacity, performance, availability, and recoverability is the key to a successful vSphere design. The following diagram displays how a storage design is integrated into the design process:



Storage design phase of in the vSphere design workflow

Several storage options and protocols are supported in the vSphere environment. The architecture that's chosen for a vSphere deployment depends on the capabilities and features needed to meet the design requirements.

This chapter will cover calculating the storage capacity and performance requirements, sizing datastores, and selecting a storage protocol. The calculations for the recipes in this chapter will be based on the following requirements that were identified in *Chapter 3*, *The Design Factors*:

- There are 100 application servers.
- Each application server is configured with 100 GB of disk space. The peak disk capacity usage of a single application server is approximately 65 percent of the total or 65 GB. The average disk performance of a single application server is 65 IOPS with an IO profile of 90 percent read and 10 percent write.
- Provide capacity to support growth for 25 additional application servers over the next 5 years.

Several new storage features are available with the release of vSphere 6.7. These new storage features include improvements to **Virtual SAN** (**VSAN**) and **Virtual Volumes** (**VVOL**), and support for **Persistent Memory** (**PMEM**) and **Remote Direct Memory Access** (**RDMA**) **over Converged Ethernet** (**RoCE**). This chapter will also provide an overview of these new storage options so that they can be incorporated into a vSphere 6 design.

In this chapter, we will cover the following recipes:

- Identifying RAID levels
- Calculating storage capacity requirements
- Determining storage performance requirements
- Calculating storage throughput
- Storage connectivity options
- Storage path selection plugins
- Sizing datastores
- Designing VSAN for virtual machine storage
- Designing **Virtual Volumes** (**VVOL**) for virtual machine storage
- Incorporating storage policies into a design
- NFS v4.1 capabilities and limits
- Using persistent memory to maximize VM performance

Identifying RAID levels

A **Redundant Array of Independent Disks** (**RAID**) combines multiple physical disks into a single unit of storage. The advantages in speed, reliability, and capacity can be realized, depending on which RAID level is selected. RAID provides the first level of protection against data loss due to a disk failure.

How to do it…

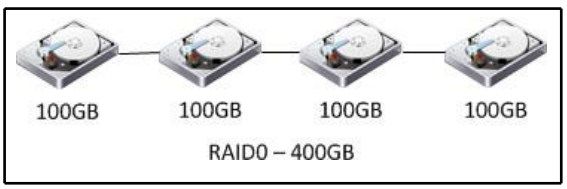
To select the proper RAID level to support the virtual workloads, you need to perform the following steps:

1. Identify the different RAID levels and capabilities
2. Select an appropriate RAID level to support a virtualized workload based on capacity and performance requirements

How it works…

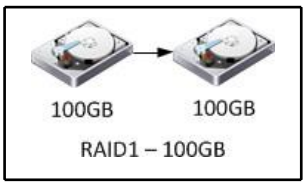
RAID0 stripes disks together to appear as a single disk with a capacity equal to the sum of all the disks in the set. RAID0 provides excellent performance and capacity efficiency, but offers no data protection. If a disk fails in a RAID0 set, the data is lost and must be recovered from a backup or some other source. Since this level offers no redundancy, it is not a good choice for production or mission-critical storage.

The following diagram illustrates the disks in a RAID0 configuration:



RAID 0

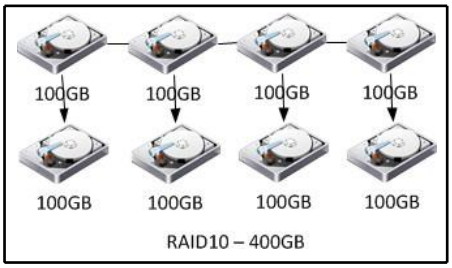
RAID1 duplicates or mirrors data from one disk to another. A RAID1 set consists of two disks and data is written on both the disks, which can then be read from either disk. If one of the disks fails, the mirror can be rebuilt by replacing the disk. The following diagram illustrates disks in a RAID1 configuration:



RAID 1

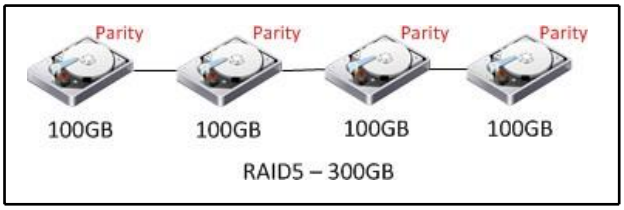
RAID1+0, RAID1/0, or RAID10 is a stripe of multiple mirrors. RAID10 provides excellent redundancy and performance, making this the best option for mission-critical applications. RAID10 is well-suited for applications with small, random, write-intensive IOs, such as high transaction applications, large messaging applications, or large transactional database applications. A RAID10 set can recover from multiple drive failures, as long as two drives in the same mirror set do not fail.

Both RAID1 and RAID10 have a capacity efficiency of 50 percent, since half of the disks in a RAID1 or RAID10 set are used to store the mirrored data. The following diagram illustrates disks in a RAID10 configuration:



RAID 10

In RAID5, data is striped across several drives and parity is written equally across all the drives in the set. This parity allows for recovery from a failure of a single drive in the set. This level offers a balance of performance and capacity and is suitable for storing transactional databases, web servers, application servers, file servers, and mail servers. The following diagram illustrates disks in a RAID5 configuration:



Parity distribution in RAID 5

The capacity efficiency of a RAID5 set is calculated using the formula *[(n - 1) / n] * 100*, where *n* is equal to the number of disks in the set. For example, a RAID5 set containing four 100 GB disks would provide 75 percent of the total capacity, or 300 GB:

*[(4-1)/4]*100 = 75%*

*(100GB * 4) * .75 = 300GB*

RAID6 is similar to RAID5, except that the two parity blocks are written and distributed equally across all the drives in the set. The second parity increases the write penalty but protects against two drive failures in the set. File archiving and file servers are common workloads that are hosted on RAID6.

The capacity efficiency of a RAID6 set is calculated using the formula *[(n - 2) / n] * 100*, where *n* is equal to the number of disks in the set. A RAID6 set containing six 100 GB disks would provide approximately 67 percent of the total capacity, or 400 GB:

*[(6 - 2) / 6] * 100 = ~67%*

*(100GB * 6) * .67 = ~400GB*

There's more…

To increase the redundancy of a RAID set, hot spares should be configured in the array. Hot spares are used to automatically replace a failed drive in a RAID set temporarily, until the failed drive can be replaced. A single hot spare can be configured to provide protection for multiple RAID sets.

Calculating storage capacity requirements

Capacity is typically measured in **gigabytes** (**GB**) or **terabytes** (**TB**). Capacity should include the total space needed to support the current requirements, the space needed to support growth, the space needed for virtual machine swapfiles, and the additional slack space for snapshots, logs, and other virtual machine data.

How to do it…

To calculate the storage capacity requirements, you need to perform the following steps:

1. Determine the capacity required to support the current workloads
2. Determine the capacity required to support future growth

How it works…

Capacity is calculated to support the current and future growth based on the design requirements, as follows:

Current capacity = 100 virtual machines x 100 GB = 10 TB

Growth capacity = 25 virtual machines x 100 GB = 2.5 TB

20% slack space = 12.5 TB x .20 = 2.5 TB

Capacity = 12.5 TB + 2.5 TB = 15 TB

Each virtual machine will have a swapfile or `.vswp` file that is created when the virtual machine is powered on. The size of the `.vswp` file for each virtual machine is equal to the size of the allocated memory, minus the memory reservation:

vSwap capacity = (100 virtual machines + 25 future virtual machines) x 8 GB of memory = 1 TB

The total capacity needed to support these requirements is 16 TB.

There's more…

The application servers are configured with 100 GB of disk space, but the maximum space that is actually consumed by a server is only 65 GB. Resizing the virtual machine disk or using thin provisioning can reduce the required amount of storage capacity significantly.

Since only the actual used space is consumed, thin provisioning virtual machine disks allows for the disk capacity to be over- allocated, which means that more capacity can be allocated to the virtual machine disks than what is actually available on the datastore. This increases the amount of management oversight required to monitor the capacity. vCenter datastore alarms can be configured to monitor over-allocation and datastore usage to assist in capacity management.

Determining storage performance requirements

Storage performance is an important factor of storage design. The storage must be designed to meet not only the capacity requirements but also the performance requirements for writes and reads to disk. Disk performance is measured in **Input/Output per Second** (**IOPS**). One disk read request or one disk write request is equal to one IO. The storage performance must support the current requirements and growth.

How to do it…

The IOPS required to support an application is calculated based on the percentage of read IO, the percentage of write IO, and the write penalty of the RAID level the workload will be hosted on.

To calculate the IOPS requirements, perform the following steps:

1. Determine the number of IOPS a workload requires
2. Identify the percentage of read IO to write IO for the workload
3. Determine the write penalty of the RAID level that will host the workload
4. Calculate the IOPS the storage must be capable of providing to support the workload

How it works…

To get the total amount of required IOPS, multiply the number of workloads by the number of functional application IOPS:

*Total IOPS = (100 current workloads + 25 future workloads) * 65 IOPS = 8125 IOPS*

To calculate the functional IOPS required for a specific workload, use the following formula:

*Functional workload IOPS = (workload IOPS * %reads) + ((workload IOPS * %writes) * write penalty)*

The write penalty is based on the number of IO operations a specific RAID configuration requires for a single write request. Writing data to multiple disks in mirror or parity calculations in a RAID5 or RAID6 configuration adds IO operations to the write request.
The write request is not completed until the data and parity are written to the disks.

The following table illustrates the write penalty based on the RAID levels:

RAID	**Write penalty**
0	1
1	2
5	4
6	6
10	2

Based on the requirements of 65 IOPS per workload with 90 percent reads and 10 percent writes on the storage configured in RAID5, the actual workload IOPS would be 85 IOPS:

*Functional application IOPS = (65 * .90) + ((65* .10) * 4) = ~85 IOPS*

Each disk in a storage array is able to provide a number of IOPS. The number of IOPS a single disk can deliver is calculated from the average latency and the average seek time of the disk. The formula to calculate disk performance is as follows:

IOPS = 1 / (average latency in milliseconds + average seek time in milliseconds)

The following table lists some approximate IOPS provided, based on the spindle speed and the drive type:

Drive speed	**~ IOPS**
SSD	> 2500
15k SAS/FC	175
10k SAS/FC	125
7,200 NL-SAS/SATA	75
5,400 SATA	50

Based on the number of IOPS required, there will be a need of *47 15k SAS drives* to support the workload:

8125 IOPS / 175 IOPS per drive = 46.4 or 47 15k SAS drives

The same workload on drives configured in RAID10 sets would require *52 15k SAS drives* to provide the required IOPS:

*Functional workload IOPS = (65 * .90) + ((65* .10) * 2) = 72 IOPS*

*Total IOPS = (100 current workloads + 25 future workloads) * 72 IOPS = 9,000 IOPS*

9,000 IOPS / 175 IOPS per Drive = 51.4 or 52 15k SAS drives

There's more…

Many arrays provide a caching mechanism using memory or SSD disks to increase the number of IOPS the array can deliver. This allows a few slow drives to deliver a higher number of IOPS. This caching can greatly reduce the number of drives needed to deliver the same number of IOPS by writing to a faster cache instead of writing directly to disks. Fast cache of EMC and flash cache of NetApp are examples of vendor-specific SSD caching technologies that can be used to increase storage IO performance.

Calculating storage throughput

The data transfer rate or throughput is the rate at which data can be read from or written to the storage device and is typically measured in MB/s. Storage adapters, connectivity, and array controllers will need to support the storage throughput requirements.

How to do it…

Throughput should be calculated to ensure that the storage controllers and disk can support the required data transfer rates. Throughput is also used to correctly size the storage connectivity bandwidth.

To calculate storage throughput requirements, perform the following steps:

1. Determine the IO size of the workload
2. Determine the number of IOPS required to support the workload
3. Calculate the throughput required

How it works…

Throughput is calculated by multiplying the IO size of the workload by the number of IOPS. Transactional databases and application servers typically have an IO size between 4 k and 64 k, whereas file archiving applications, backup applications, and media streaming applications typically have larger IO sizes from 64 k to 1,024 k.

To calculate the throughput, the following formula is used:

*Throughput = functional workload IOPS * IO Size*

Using the functional workload IOPS from the previous recipe and an IO size of 8 k, the throughput required can be calculated as follows:

*Throughput = 9,000 * 8k = 72 MB/s*

Network interface card bandwidth is usually expressed in Mbps. To convert MB/s to Mbps, simply multiply by *8*:

*Bandwidth Mbps = 72 MB/s * 8 = 576 Mbps*

The array would need to support a throughput of at least 72 MB/s, and the connectivity bandwidth would need to be sufficient enough to support at least 576 Mbps.

Storage connectivity options

vSphere supports multiple storage protocols and connectivity options. Storage can be directly connected to a host, or storage can be centralized and shared with multiple hosts. Shared storage is required when implementing many vSphere features, such as VMware **High Availability** (**HA**), VMware **Fault Tolerance** (**FT**), and VMware **Distributed Resource Scheduling** (**DRS**).

How to do it…

To determine the storage connectivity requirements, perform the following steps:

1. Identify the supported storage protocols and connectivity options.
2. Select the storage protocol and connectivity that supports the design requirements.

How it works…

Performance, availability, and costs are all factors that should be considered when choosing a storage connectivity option. The following table provides a quick overview of the different storage connectivity options and how they compare with each other in terms of performance, availability, and costs:

Protocol	**Performance**	**Availability**	**Costs**
Local storage	Good	Fair	Low
Fibre channel	Excellent	Excellent	High
iSCSI	Good	Excellent	Medium
NFS	Good	Good	Low
FCoE	Excellent	Excellent	High

Direct attached or local storage is storage directly attached to a host. Since this storage is not shared, many VMware features will not be available for virtual machines hosted on the local storage.

Best practices when using direct attached or local storage are as follows:

- Configure RAID to provide protection against a hard disk failure
- Use a hardware RAID controller that is on the VMware HCL

Fibre Channel (**FC**) is a block-level, low latency, high-performance storage network protocol that is well-suited for workloads with high I/O requirements. The FC protocol encapsulates the SCSI commands into the FC frames. A Fibre Channel **Host Bus Adapter** (**HBA**) is required to connect the host to the storage network or fabric. FC HBAs can provide a throughput of 2, 4, 8, or 16 Gbps, depending on the capabilities of the HBA and the FC network. FC uses zoning and LUN masking to configure which hosts can connect to which targets on the SAN.

The cost of deploying FC-connected storage can be significantly higher than other options, especially if an existing FC infrastructure does not already exist.

The best practices when using FC are as follows:

- Use multiple HBAs in the host to provide multiple paths from load balancing and redundancy.
- Ensure all HBAs and switches are configured for the same speed. Mixing the speed of HBAs and switches can produce contention at the FC switch and SAN.
- Use single-initiator single-target zoning. A single HBA, the initiator, is zoned to a single array target, the target. A separate zone is created for each host HBA.
- Mask LUNs are presented to ESXi hosts from other devices.
- Ensure firmware levels on FC switches and HBAs are up-to-date and compatible.

iSCSI provides block-level storage access by encapsulating SCSI commands in TCP/IP. iSCSI storage can be accessed with the iSCSI software initiator, which is included with ESXi through a standard network adapter, or using a dependent or independent iSCSI HBA:

- A dependent iSCSI adapter depends on VMware networking and iSCSI configuration for connectivity and management.
- An independent iSCSI HBA provides its own networking and configuration for connectivity and management. Configuration is done directly on the HBA through its own configuration interface.

Throughput is based on the network bandwidth, the speed of the network interface card (1 Gbps or 10 GbE), and the CPU resources required to encapsulate the SCSI commands into TCP/IP packets.

The cost of implementing iSCSI is typically significantly lesser than implementing FC. Standard network adapters and network switches can be used to provide iSCSI connectivity. Using dedicated iSCSI HBAs not only increases performance, but also increases cost. The price of 10 GbE switches and 10 GbE adapters continues to drop as the deployment of these becomes more widespread.

The best practices when using iSCSI are as follows:

- Configure multiple vmks bound to multiple vmnics to provide load balancing and redundancy for iSCSI connections.
- Use network cards with **TCP/IP Offload Engine** (**TOE**) enabled to reduce the stress on the host CPU.
- Use a physically separate network for iSCSI traffic. If a physically separate network is not available, use VLANs to separate iSCSI traffic from other network traffic.
- Enable jumbo frames (MTU 9000) on the iSCSI network.

The **Network File System** (**NFS**) **protocol** can be used to access virtual machine files stored on a **Network Attached Storage** (**NAS**) device. Virtual machine configuration files, disk (VMDK) files, and swap (`.vswp`) files can be stored on the NAS storage. vSphere 5.5 supports NFS Version 3 over TCP, and vSphere 6 added support for NFS v4.1. The capabilities and limitations of NFS v4.1 will be discussed in a separate recipe later in this chapter.

Throughput is based on the network bandwidth, the speed of the network interface card (1 Gbps or 10 GbE), and the processing speed of the NAS. Multiple paths can be configured for high availability, but load balancing across multiple paths is not supported with NFS.

The cost of implementing NFS connectivity is similar to iSCSI. No specialized network hardware is required. Standard network switches and network adapters are used and there is no need for specialized HBAs.

The best practices when using NFS-connected storage are as follows:

- Use a physically separate network for NFS traffic. If a physically separate network is not available, use VLANs to separate NFS traffic from other network traffic.
- Hosts must mount NFS version 3 shares and non-Kerberos NFS version 4.1 shares with root access.
- Enable jumbo frames (MTU 9000) on the NFS network.

Fibre Channel of Ethernet (**FCoE**) encapsulates Fibre Channel in Ethernet frames. A **Converged Network Adapter** (**CNA**) that supports FCoE is required, or a network adapter with FCoE capabilities can be used with the software FCoE initiator included with ESXi.

A common implementation of FCoE is with Cisco UCS blade chassis. The connectivity for TCP/IP network and FCoE storage traffic is converged between the chassis and the Fabric Interconnects. The Fabric Interconnects splits out the traffic and provides the connectivity paths to the TCP/IP network and storage network fabrics.

The best practices when using FCoE are as follows:

- Disable the **Spanning Tree Protocol** (**STP**) on the switch ports connected to FCoE adapters
- Ensure that the latest microcode is installed on the FCoE network adapter
- If the FCoE network adapter has multiple ports, configure each port on a separate vSwitch

Storage path selection plugins

Mutlipathing allows more than one physical path to be used to transfer data between the ESXi hosts and the storage array. In the event of a failure in a storage path, the host or hosts can switch to another available path. Multipathing also provides load balancing by distributing the storage IO across multiple physical paths.

How to do it…

To determine the multipathing policy, perform the following steps:

1. Identify the different native multipathing policies available and the capabilities of each policy.
2. Select a multipathing policy based on the number of paths and the array type used.
3. Change the default multipathing policy using the `esxcli` command.
4. Configure the multipathing policy on the storage devices presented to the ESXi host.

How it works…

The VMware **Native Multipathing Plugin** (**NMP**) is the built-in multipathing plugin for ESXi. It only supports storage arrays listed on the **Hardware Compatibility List (HCL)**. The NMP automatically detects the type of storage array used and sets the appropriate path selection policy by associating a set of physical paths with a storage device or LUN.

The **Storage Array Type Plugin** (**SATP**) monitors the available storage paths, reports changes in the path status, and initiates failover between paths when needed. The **Path Selection Plugins** (**PSP**) determine which available path to use for IO. There are three native multipathing PSPs available:

- **Fixed**: The host always uses a preferred path if the preferred path is available. If the preferred path fails, another available path is selected and used until the preferred path becomes available. This is the default policy for active/active storage devices.
- **Most Recently Used (MRU)**: The host uses the most recently used path. If the current path fails, another path is selected. IO does not revert to the previous path when it becomes available. This is the default policy for active/passive storage devices.
- **Round Robin (RR)**: IO is rotated through all active paths. This provides load balancing across all physical paths available to the hosts. This PSP can be used on active/passive or active/active arrays.

Array vendors may provide their own path selection plugins to provide storage multipathing. The use of third-party MPPs will depend on array-and-vendor best practices. The NMP can be used for any supported array.

The optimal PSP to choose is dependent on the recommendations of the array vendor.

By default, a PSP is set based on the SATP used for the array. The SATP to use is identified by the **Pluggable Storage Architecture** (**PSA**) using a set of claim rules that base the selection on the vendor and model of the array. The SATP then determines the default PSP to be used.

The NMP PSP policies are as follows:

- `VMW_PSP_MRU`: For most recently used
- `VMW_PSP_FIXED`: For fixed
- `VMW_PSP_RR`: For round robin

The default `PSP` for an `SATP` can be changed using the following `esxcli` command:

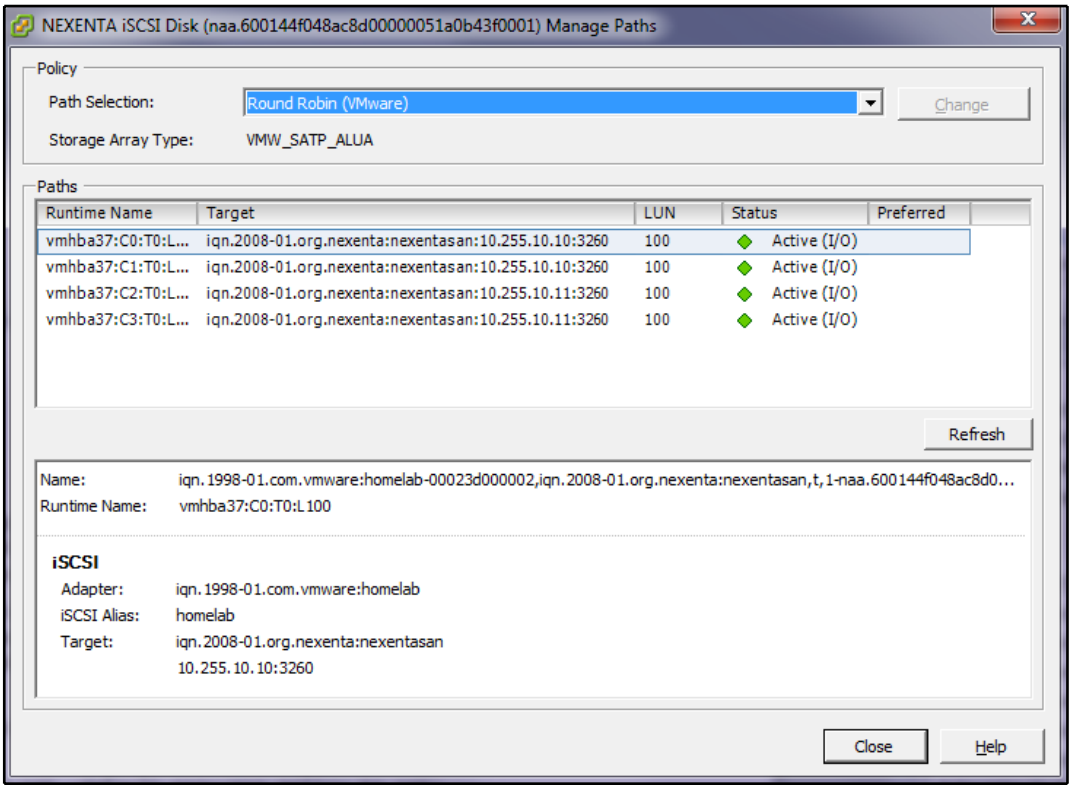
```
   esxcli storage nmp satp set -default-psp=<psp policy to set>
--satp=<SATP_name>
```

Using the command line changes the default PSP for all new devices identified by the SATP. The PSP for an individual device or LUN can also be changed. This can be done in the vSphere Client or vSphere Web Client by managing the paths for a single storage device on a host.

To change the PSP of a device using the vSphere Client, navigate to **Host and Clusters** | **Hosts** | **Configuration** | **Storage Adapters** and select the storage adapter that services the paths you want to modify. In the **Details** section of the **Storage Adapters** window, right-click on the device you want to modify the PSP on and select **Manage Paths**.

From the **Manage Paths** window, select the **Path Selection** with the dropdown menu and click on **Change**.

The following screenshot displays how to select and change the path selection policy for a device using the vSphere Client:



Path Selection Policy in the vSphere Client

Changing the default PSP for an SATP or the PSP for a device can be done without impacting normal operations. A change that's made to the PSP for a single device takes effect immediately. Changing the default PSP for the SATP changes only the settings of newly discovered devices and not the PSP settings of the current devices.

Sizing datastores

A datastore is a logical representation of storage that's presented to an ESXi host where virtual machine files are stored. A datastore can be a VMFS formatted volume, an NFS export, a **Virtual Volume** (**VVOL**) datastore, a **Virtual SAN** (**VSAN**) datastore, or a path on the local ESXi filesystem.

How to do it…

Design requirements, virtual machine disk size, IOPS, and recovery are all factors that can determine the number of virtual machines to store on a single datastore. The size of the datastore is calculated based on the number of virtual machines per datastore and the size of the virtual machines:

1. Determine the number of virtual machines per datastore based on the capacity, performance, and recovery requirements
2. Understand the impact the SCSI reservations may have on datastore sizing
3. Understand how recovery time impacts datastore sizing

How it works…

A design factor that was identified in *Chapter 3*, *The Design Factors*, specified that no more than 20 application servers should be affected by a hardware failure. Applying the same requirement to datastore sizing would mean that no more than 20 application servers should be hosted on a single datastore:

*Number of VMs per datastore * (VM disk size + .vswp size) + 20% = Minimum datastore size*

The datastore size for 20 application server workloads, each with 100 GB of disk storage and 8 GB of RAM with no reservations, plus 20 percent for slack, would be approximately 2.5 TB:

*20 * (100 GB + 8GB) + 20% = 2,592 GB or ~ 2.5 TB*

The storage backing the datastore has to provide enough IOPS to support the virtual machines running on it. If a virtual machine generates 50 IOPS and there are 20 virtual machines on the datastore, the storage must be able to support 1,000 IOPS.

The maximum size of a VMFS6 datastore is 64 TB.

Block storage formatted as a VMFS volume is susceptible to SCSI reservations or locking of the entire LUN for a very short period of time by a single host. A few operations that cause SCSI reservations to occur are as follows:

- Creating a VMFS datastore
- Expanding a VMFS datastore
- Powering on a virtual machine
- Creating a template
- Deploying a virtual machine from a template
- Creating a virtual machine
- Migrating a virtual machine with vMotion
- Developing a virtual machine disk
- Creating or deleting a file

With the introduction of VMFS5, along with the **vStorage APIs for Array Integration** (**VAAI**) hardware-assisted locking feature, the impact of SCSI reservations is minimized. If an array does not support the VAAI hardware-assisted locking feature, then the number of virtual machines per datastore may need to be decreased to reduce the impact of LUN locking for SCSI reservations.

The **Recovery Time Objective** (**RTO**) must also be taken into account when determining the size of a datastore. If the datastore is lost or becomes inaccessible, how long will it take to restore the virtual machines that were running on it?

Size of datastore / GBs recovered per hour <= RTO

If 500 GB is to be recovered per hour, the time to recover a failed datastore can be calculated as follows:

2.5 TB / 500 GB = 5 hours to recover

If the RTO for the applications or workloads running on the datastore is less than 5 hours, the datastore would need to be resized to ensure that recovery would take place within the defined RTO.

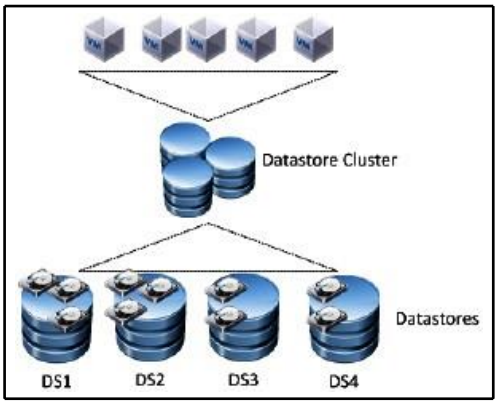
There's more…

Multiple datastores can be aggregated to create a datastore cluster. A datastore cluster is a collection of the member datastore resources with a shared management interface. vSphere Storage DRS manages the datastore cluster resources to determine the initial placement and ongoing balancing of virtual machine VMDKs across the datastores in the cluster. Datastore clusters are supported for both VMFS and NFS datastores.

A few recommended practices when using datastore clusters and Storage DRS are as follows:

- Cluster datastores with similar IOs and capacity characteristics
- Use separate datastore clusters for replicated and nonreplicated datastores
- Do not mix NFS and VMFS datastores in the same datastore cluster
- Do not place datastores shared across multiple data centers in a datastore cluster

When a virtual machine is placed on a datastore cluster, Storage DRS determines which datastore in the cluster the files will be stored in, based on space utilization and/or performance. The following diagram is a logical representation of the virtual machines
placed on a datastore cluster:



Virtual machines in a datastore cluster

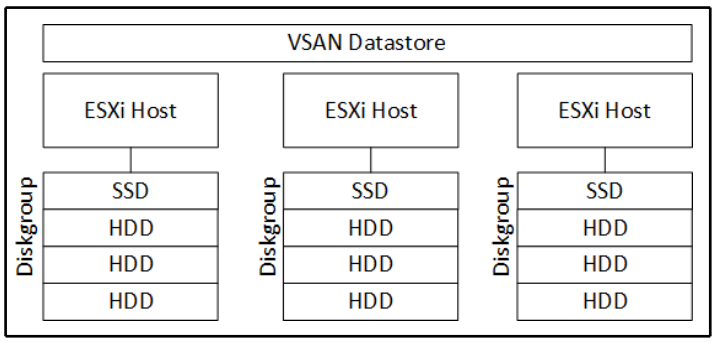
The best practices when using datastore clusters are as follows:

- Datastores in a cluster should have similar performance capabilities
- Keep similar virtual machine IO workloads together on the same cluster
- Do not mix replicated and nonreplicated datastores in the same cluster
- Do not mix NFS and VMFS datastores in the same datastore cluster
- Use VMDK affinity rules to keep virtual machine disk files together on the same datastore within the datastore cluster
- Use VM anti-affinity rules to ensure that virtual machines run on different datastores within the datastore cluster

The *VMware vSphere Storage DRS Interoperability* whitepaper can be found at `http://www.vmware.com/files/pdf/techpaper/vsphere-storage-drs-interoperability.pdf`. This whitepaper provides an overview of the datastore cluster best practices and interoperability of datastore clusters, along with other VMware products.

Designing VSAN for virtual machine storage

VMware **Virtual SAN** (**VSAN**) is integrated into the ESXi hypervisor. VSAN virtualizes and aggregates the local direct-attached disks in ESXi hosts. This creates a single pool of storage resources from the local disks with each host that is shared across all hosts in the VSAN cluster, as shown in the following diagram:



VSAN diskgroups

How to do it…

To use VSAN for storage in a vSphere virtual infrastructure design, follow these steps:

1. Identify the hardware requirements to support VSAN
2. Verify that the disks and controllers are on the VSAN **Hardware Compatibility List** (**HCL**)
3. Size VSAN to support performance and availability
4. Enable VSAN on the vSphere Cluster

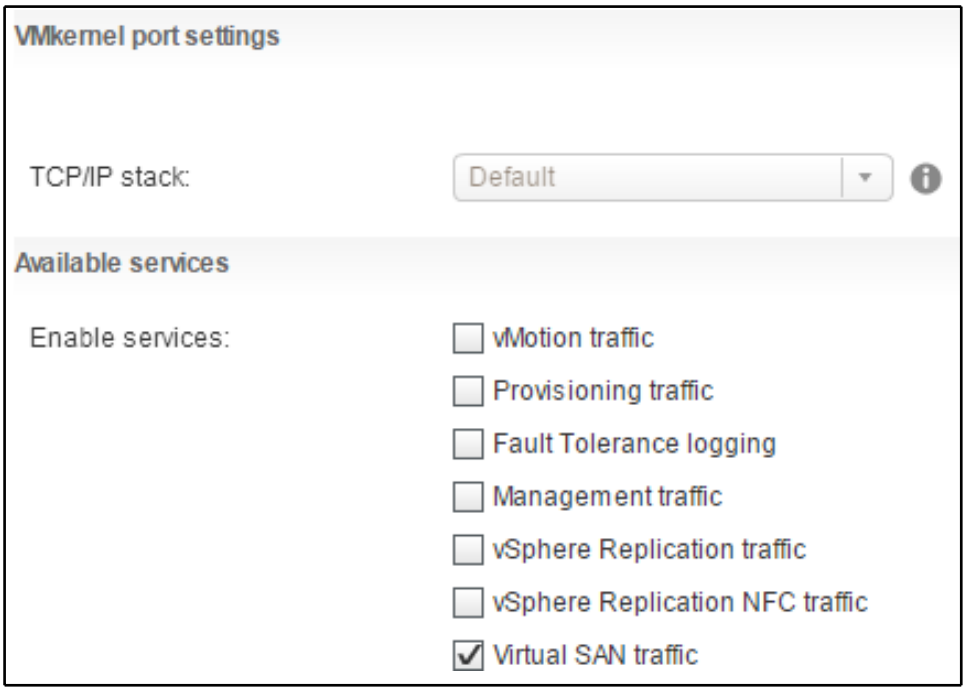
How it works…

VSAN presents shared storage to ESXi hosts across a vSphere Cluster. Each host providing storage to the VSAN cluster requires the following:

- **Solid-State Disks** (**SSD**) to provide performance
- **Hard Disk Drives** (**HDD**) or SSDs to provide capacity
- A disk controller
- Network connectivity between hosts

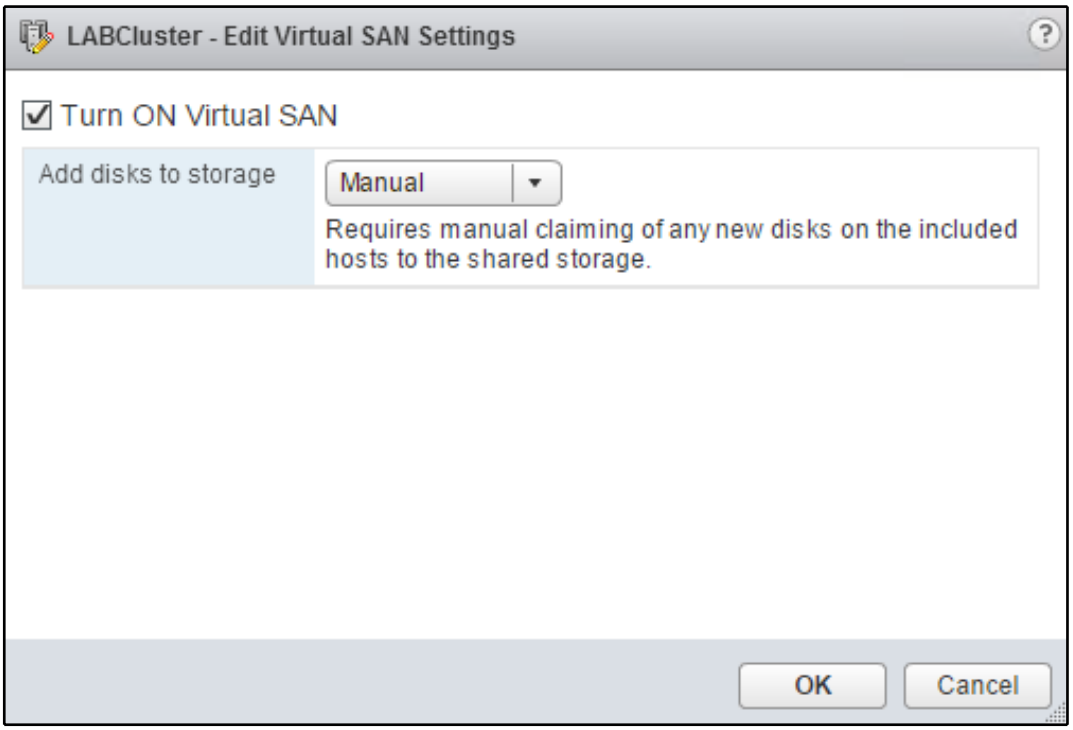
As with all hardware in a vSphere environment, the hardware supporting VSAN must be verified on the **Hardware Compatibility List** (**HCL**) located at `http://www.vmware.com/resources/compatibility/search.php?deviceCategory=vsan`. Compatibility of the SSD, HDD, and disk controller, including the firmware, should all be validated on the HCL. Many server hardware vendors offer VSAN-ready nodes that have been preconfigured with supported hardware/firmware.

VSAN can be deployed as a hybrid, SSD and HDD disks, or as All-Flash. VSAN requires network connectivity between hosts. A 10 GbE network should be used for VSAN to provide the best performance, but 1 GbE is supported in a hybrid VSAN. 10 GbE is required for an All-Flash VSAN. A VMkernel is configured and enabled for VSAN traffic, as shown in the following screenshot:



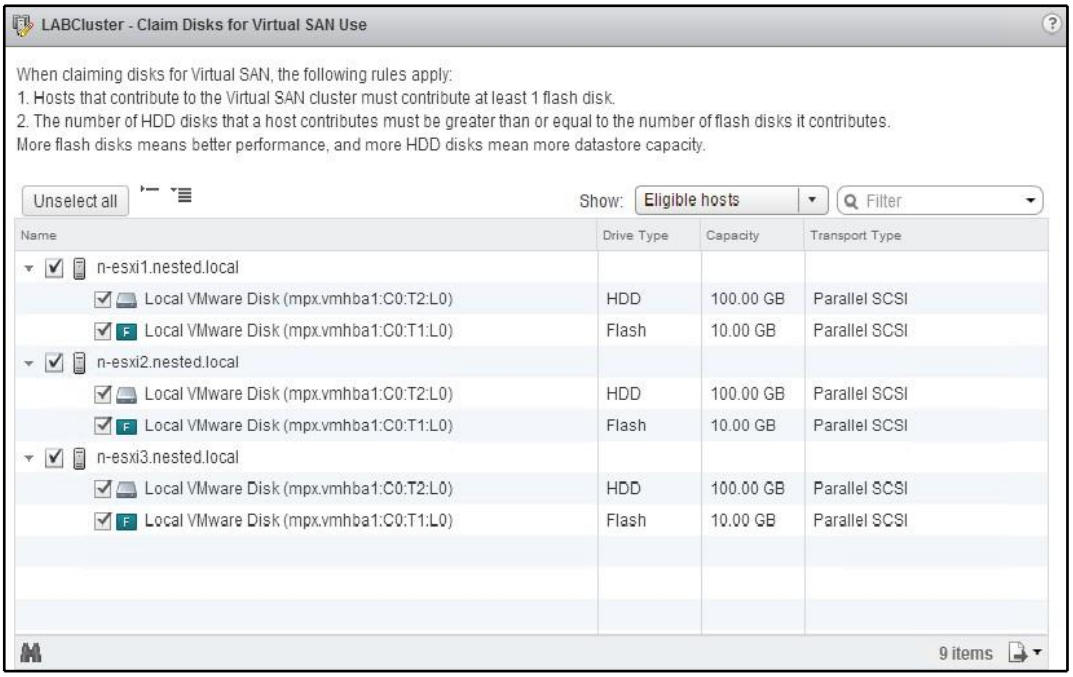
Enabling VSAN traffic

VSAN is enabled on a vSphere Cluster. ESXi hosts participating in VSAN, regardless of whether the host is providing storage or consuming storage, must be in the same cluster. The VSAN storage cannot be directly consumed by hosts outside the cluster. VSAN is enabled by simply turning on VSAN for a vSphere Cluster, as shown in the following screenshot:



Enabling VSAN

When enabling VSAN, the disk can be automatically or manually claimed. Disks are claimed to be used by VSAN and placed into disk groups. A disk group must contain at least one flash (SSD) disk and one or more HDDs. A single host can be configured with up to five disk groups and each disk group, can contain up to eight disks, one SSD, and seven HDDs. All Flash disk groups are also supported. In a VSAN disk group, the Flash disk provides performance and the HDD disk provides capacity. Claiming disks for VSAN is shown in the following screenshot:



Claiming disks for VSAN

When sizing VSAN, VMware recommends the SSDs be sized to 10 percent of the HDD capacity in a disk group. For example, if there is 1 TB of HDD capacity, the SSD should be at least 100 GB. The capacity of the disk group is the sum of the HDD capacity. If there are three 1 TB HDDs in the disk group, the group will provide 3 TB of capacity storage. The size of the VSAN in the datastore is the aggregate of all disk groups claimed by VSAN across all hosts.

Three hosts, participating in a VSAN cluster, each with a disk group of three 1 TB HDDs to provide capacity, will present a VSAN datastore with approximately 9 TB of usable capacity.

When sizing VSAN, it is important to take into account **Failures To Tolerate** (**FTT**) and Fault Domain policies for virtual machines. A virtual machine consuming 100 GB of storage stored on a VSAN datastore with the FTT policy configured to 1 will consume 2 x the capacity, or 200 GB. This is due to the virtual machine storage being duplicated across two hosts so that the virtual machine disks will be available in the event that a single host fails. If the FTT is set to 2, then a virtual machine will consume 3 x the capacity. We will take a deeper look at storage policies later in this chapter.

There's more…

VMware's latest version of VSAN, 6.7 Update 1, includes a number of significant features and improvements, including the following:

- Guided workflows for VSAN cluster creation and node additions
- VMware Update Manager firmware and driver updates for ReadyNodes
- Automated UNMAP and guest integration

VMware continues to develop and improve the capabilities, efficiencies, and performance of VSAN storage, making it a suitable alternative to traditional storage for virtual machine workloads. More details on VSAN design and sizing can be found in the *VMware VSAN Design and Sizing Guide* at `https://storagehub.vmware.com/t/vmware-vsan/vmware-r-vsan-tm-design-and-sizing-guide-2/http://www.vmware.com/files/pdf/products/vsan/VSAN_Design_and_Sizing_Guide.pdf`.

Using VMware Virtual Volumes

Virtual Volumes (**VVOL**) is a virtual disk management and array integration framework that was introduced with vSphere 6. VVOL enables policy-based storage for virtual machines. A datastore is presented as backed by raw storage supporting multiple different capabilities, such as snapshotting, replication, deduplication, raid level, performance, and so on. These capabilities are exposed to the vSphere environment. Policies are created and assigned to virtual machines. When a virtual machine is placed on a VVOL datastore, the placement on the array is based on requirements that are defined in the policies.

How to do it…

To successfully incorporate VVOL as part of a vSphere infrastructure design, perform the following steps:

1. Identify the components and characteristics of VVOL
2. Identify the limitations and interoperability of VVOL with other vSphere components
3. Create a new storage provider in vCenter
4. Add a VVOL datastore

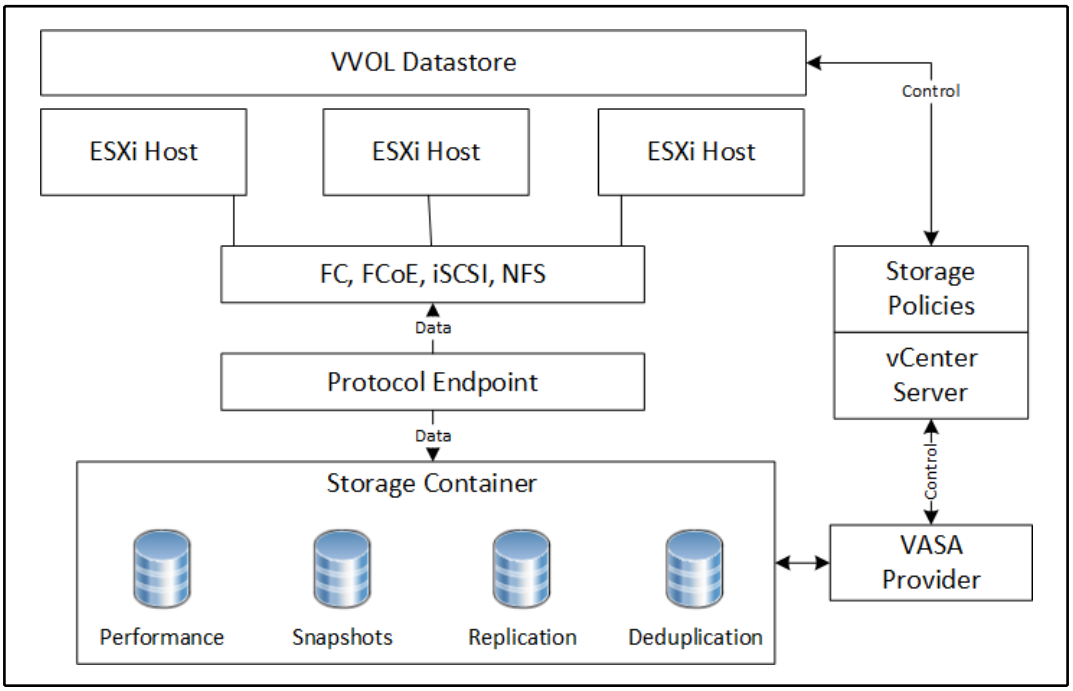
How it works…

The following table outlines the different components that are required to make up a VVOL environment:

Component	Description
vSphere APIs for Storage Awareness (VASA)	The VASA provider is a software component that was developed by the storage array vendor. The VASA provider provides the storage capability awareness to vCenter and the ESXi hosts.
Storage Containers (SC)	A pool of storage capacity and storage capabilities on the array. A storage container represents a virtual datastore.
Protocol Endpoint (PE)	A logical IO proxy that provides a data path from virtual machines to the virtual volumes.
VVOL Objects	Encapsulation of virtual machine files and disks. Objects are stored natively on the array storage containers.

VVOL differs from other vSphere storage in the fact there is no filesystem. The storage container is comprised of raw storage capacity grouped by capabilities. This storage container is presented as a datastore to the vSphere environment through the protocol endpoint. The protocol endpoint provides a data path and supports IP-based (iSCSI, NFS, FCoE) and FC connectivity. The VASA provider communicates with vCenter and the ESXi hosts to expose the storage capabilities of VVOL.

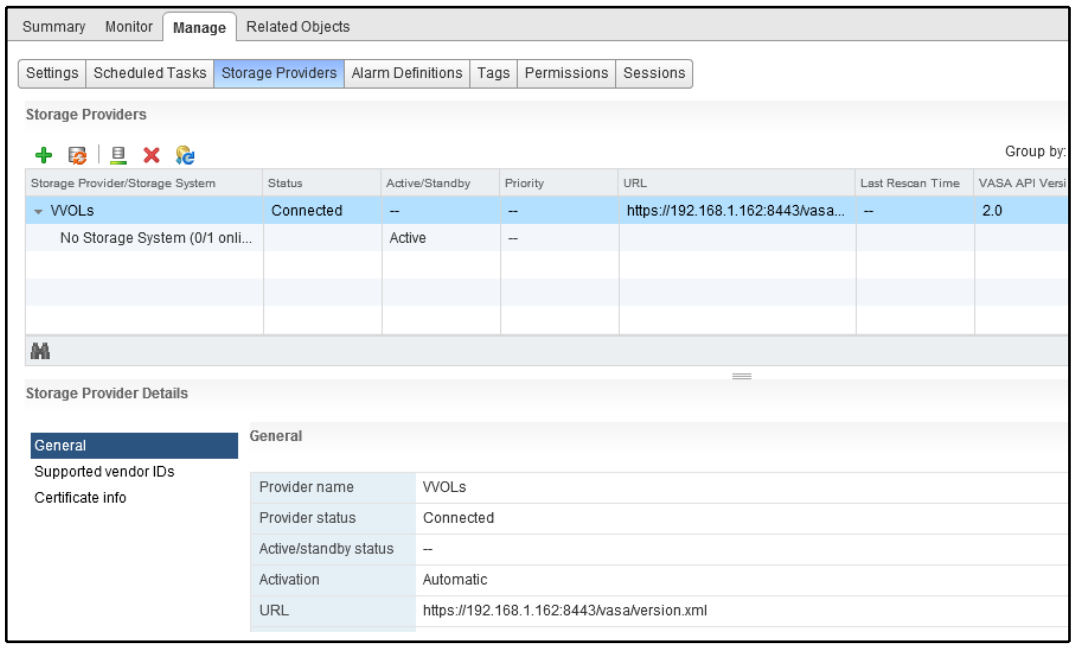
The following diagram provides a logical overview of VVOL and the connectivity between the components:



Entity relationships with VVOLs

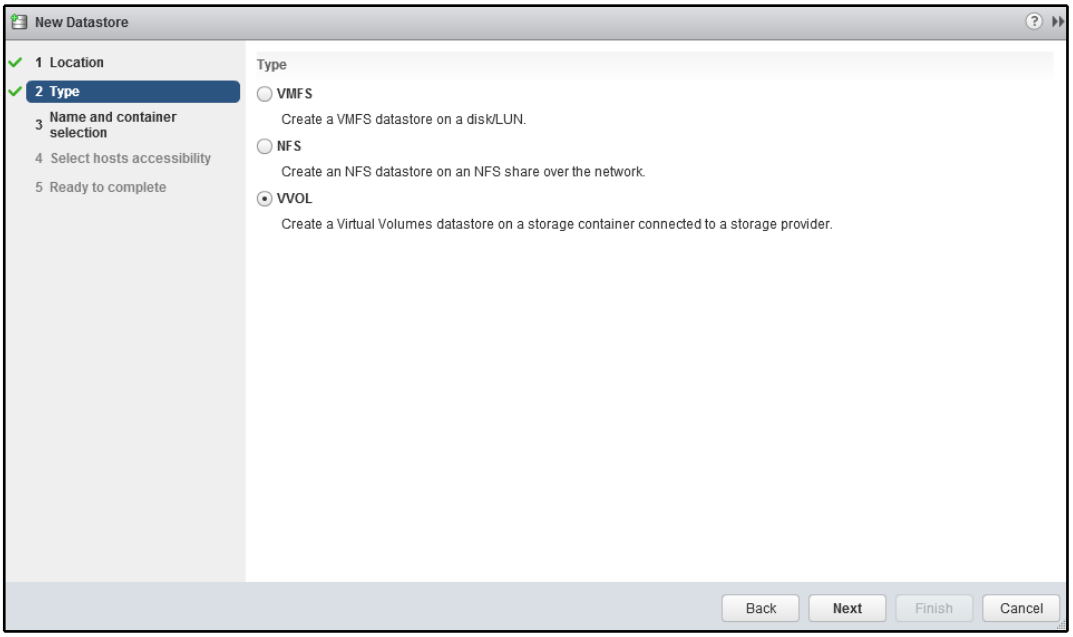
VVOL is a new feature that has only just been introduced with vSphere 6. There are still a number of limitations regarding the features and products that are supported. For example, features such **Storage IO Control** (**SIOC**), IPv6, **Fault Tolerance** (**FT**), and **Raw Device Mapping** (**RDM**) are not supported on VVOL. Products such as **vSphere Data Protection** (**VDP**) and VMware **Site Recovery Manager** (**SRM**) do not currently support using VVOL, although VMware has committed to supporting this combination in a future release. At the moment, if a vSphere design requires these features or products, VVOL will likely not be a viable choice to provide storage to the environment. For a full list of supported/unsupported features and products up to 6.5, refer to this VMware Knowledge Base article: `https://kb.vmware.com/kb/2112039`.

To create a new storage provider, the name of the provider, the URL for the VASA 2.0 provider, and a username and password or a certificate for authentication are required. Storage providers are configured per vCenter server, as shown in the following screenshot:



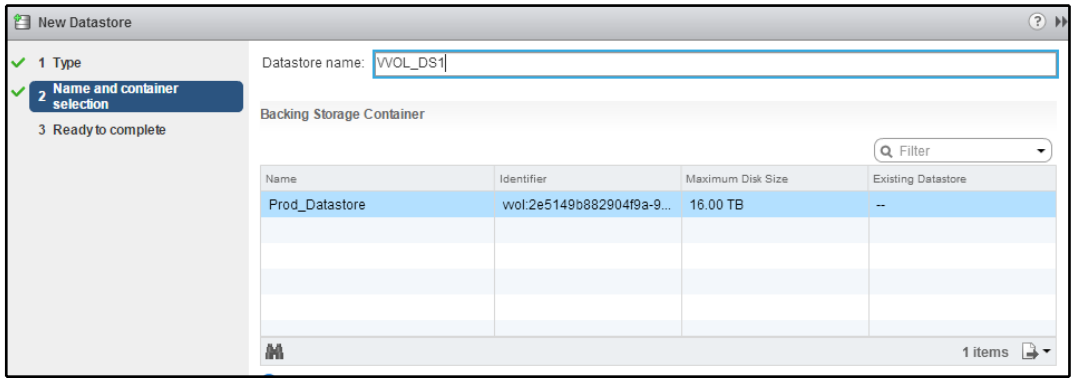
Creating VVOL storage providers

Once the storage provider has been configured, the VVOL datastore is added by using the **New Datastore** wizard and by selecting the **VVOL** datastore type, as shown in the following screenshot:



Creating a VVOL datastore

When VVOL is selected as the type in the **New Datastore** wizard, the available storage containers will be displayed. Enter a **Datastore name** and select the **Backing Storage Container**, as shown in the following screenshot:



Naming a VVOL datastore

Once complete, the datastore will be created, presented to the hosts in the environment, and available for virtual machine storage.

Incorporating storage policies into a design

Storage policies are configured to simplify the provisioning of virtual machines on storage. Storage policies ensure that service levels are met for storage performance, protection, and availability. For **Software Defined Storage** (**SDS**), including VSAN and VVOL, these policies are a key component that are required for determining virtual machine placement during provisioning and throughout a virtual machine's life cycle.

How to do it…

When incorporating storage policies into a vSphere design, perform the following steps:

1. Determine the storage services and capabilities that are required by virtual machine workloads:
    - What data protection service may be required for virtual machines?

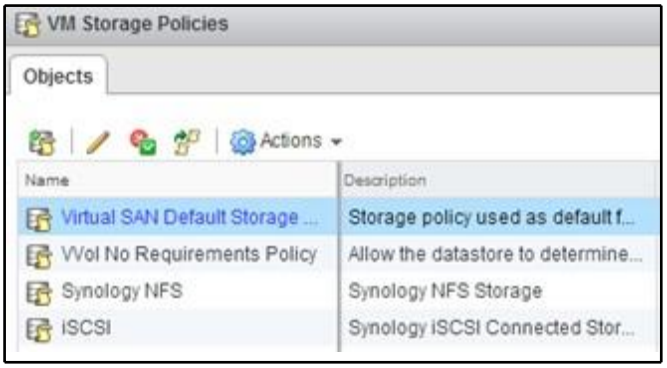
- Are capabilities such as encryption at rest, deduplication, or compression required?
- Are different tiers of storage required?

2. Identify how storage array capabilities will be discovered:
    - Is a VASA provider available to provide awareness of storage capabilities?
    - Will tags be used to manually tag datastores based on capabilities?

3. Create policies mapping storage capabilities to virtual machine requirements.
4. Assign storage policies to virtual machines and virtual machine disks.

How it works…

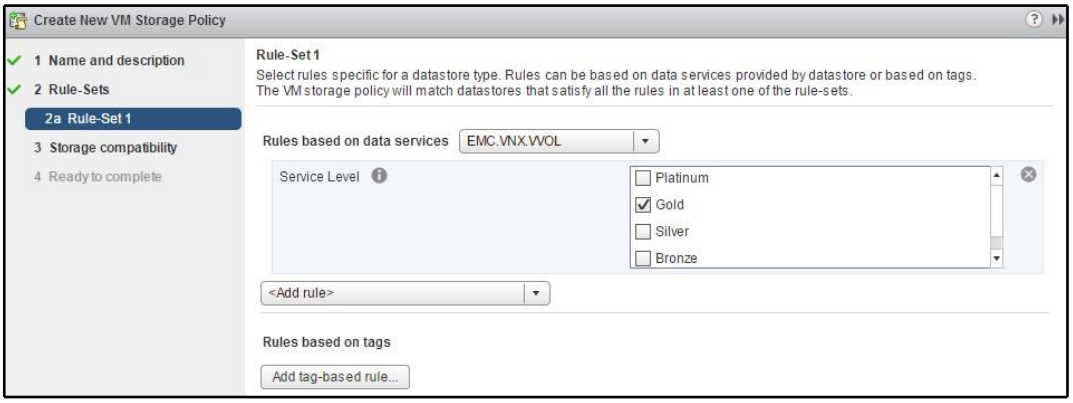
VM storage policies are created and managed through the vSphere Web Client, as shown in the following screenshot:



VM Storage Policies in the vSphere Web Client

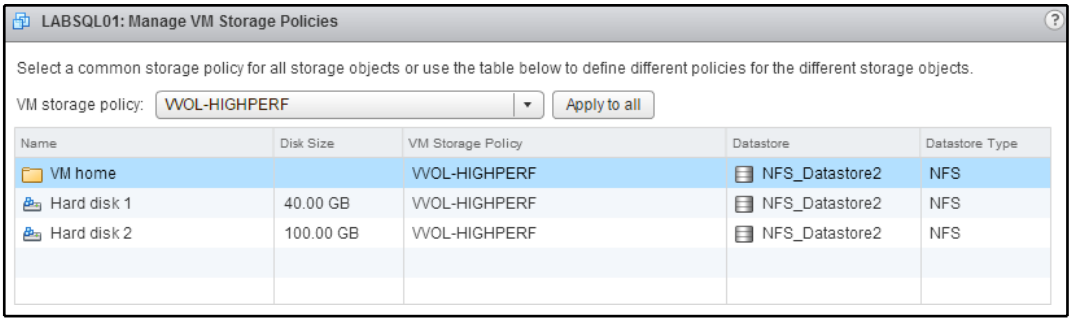
VM storage policies contain a rule or a set of rules. These can be based on tags or on data services. Tag-based rules are created from tags that the administrator creates for storage capabilities and then manually assigns them to datastores. Data services based rules are created from capabilities that are discovered from a data service provider (for example, for a VASA provider on a VVOL-enabled array).

A rule set can be based on data services, as shown in the following screenshot:



Creating rule-sets for a VM Storage Policy

VM storage policies can be assigned to a virtual machine or to individual virtual machine disks. The **Manage VM Storage Policies** dialog for a virtual machine is displayed in the following screenshot:



Assigning a VM Storage Policy to a VM

This is based on the requirements for the virtual machine. For example, a virtual machine running SQL may require different storage capabilities for the disks containing the OS, the logs, the tempDB, and the databases. Policies can be assigned to each virtual disc to ensure correct placement of the disks and ensuring compliance through the virtual machine's life cycle.

NFS version 4.1 capabilities and limits

vSphere 6 added support for NFS version 4.1. NFS clients for both NFS version 3 and NFS version 4.1 are included as part of ESXi. Using NFS version 4.1 provides additional features and functionality over NFS version 3, but there are some significant caveats and limitations that must be accounted for when using NFS version 4.1 in a vSphere design.

How to do it…

To determine how NFS version 4.1 can be incorporated into a vSphere 6 design, you must do the following:

- Identify the capabilities of NFS version 4.1
- Determine what design requirements will NFS version 4.1 satisfy
- Determine the limitations of NFS version 4.1
- Identify the requirements for configuring a NFS version 4.1 datastore

How it works…

NFS version 4.1 in vSphere 6 provides the following capabilities:

- Multipathing support for NFS datastores
- Non-root user access when using Kerberos
- Stateful server-side locking
- Better error recovery

These features provide enhancement to performance, security, and availability. There are still a number of limitations that will have an impact on using NFS version 4.1 in a vSphere design. These limitations include the following:

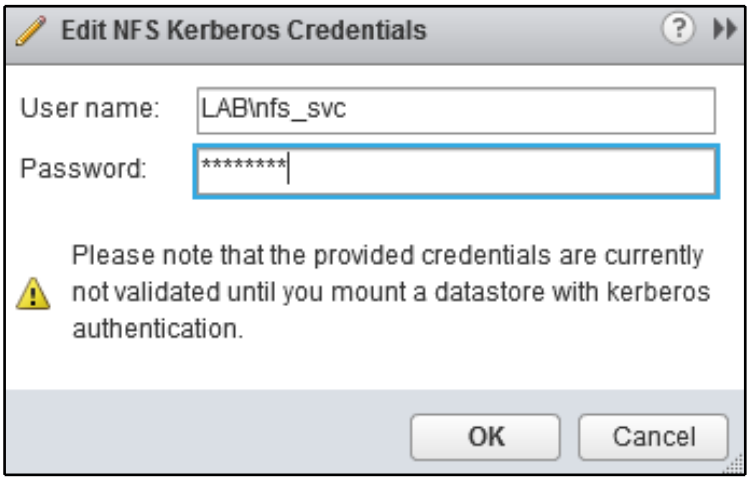
- No support for vSphere FT, VMware SRM, VVOL, or SIOC
- IPv6 is only supported for non-Kerberos datastores
- NFS version 3 datastores cannot be upgraded to NFS version 4.1
- No VAAI-NAS hardware acceleration

Features including vSphere High Availability (DRS), vSphere vMotion, and vSphere **Distributed Resource Scheduler** (**DRS**) are all supported when using NFS version 4.1 datastores.

Presenting both NFS version 3 and NFS version 4.1 datastores is supported. ESXi includes separate NFS clients to support each version. However, a single NFS share should not be accessible by both protocol versions, as this will likely result in data corruption.

The NFS server providing storage must support NFS version 4.1 and, as with all IP-connected storage, VMkernel interfaces are required on each ESXi host to support the storage connectivity.

NFS Kerberos Credentials provides a significant security improvement. It allows non-root access to the NFS export. NFS Kerberos Credentials are configured for each host. Only a single credential can be created. **NFS Kerberos Credentials** is created and managed from the **Authentication Services** settings on a host, as displayed in the following screenshot:



Entering NFS Kerberos credentials

NFS version 4.1 datastores are mounted to ESXi hosts using the **New Datastore** wizard. A datastore name and the folder location of the NFS share are required, just like NFS version 3. Multiple NFS servers can be added to provide multiple paths to the NFS version 4 share.

Using persistent memory to maximize VM performance

Persistent memory (**PMEM**) is a new technology that adds a storage layer between **Solid State Drives** (**SSDs**) and **Dynamic Random Access Memory** (**DRAM**), and takes the best of both technologies. Recall that SSDs are devices that store data in dense, non-volatile, Flash memory and are much faster than spinning hard drives that store data on magnetic disks. DRAM is typical server memory, which is very fast but is volatile, meaning that data is only stored when power is applied. When power is lost, the data in DRAM is lost as well. PMEM, also referred to as a **Non-Volatile Dual Inline Memory Module** (**NVDIMM**), is a technology that is as fast as DRAM, hundreds of times faster than SSDs, but non-volatile, like SSDs, and provides a useful function in a vSphere environment. VMs don't even have to be PMEM-aware. Legacy applications and operating systems can utilize PMEM technology in vSphere 6.7. PMEM-based storage can be presented as a virtual disk to legacy operating systems or directly to PMEM-aware operating systems.

How to do it…

Follow these steps to use PMEM storage in vSphere:

1. Install a supported PMEM device in your ESXi host
2. Present the PMEM device as a datastore
3. Create and attach virtual disks (vPEMDisk) to legacy VMs, ensuring that the VM storage policy is set to **Host-local PMem Default Storage Policy**
4. For VMs that have PMEM-aware operating systems and applications, add a new NVDIMM device from **Edit Settings….**

How it works…

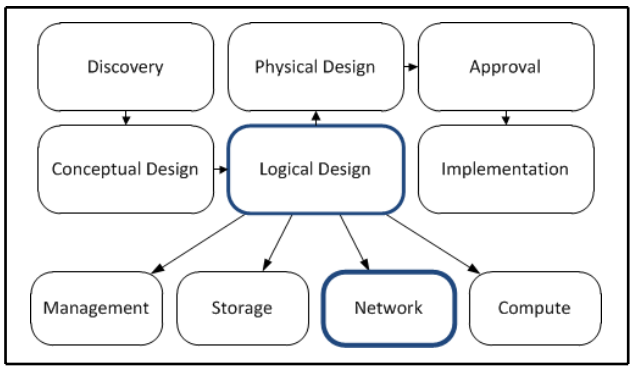
Hardware vendors with early support for PMEM include Dell EMC and HPE. Their solutions are a combination of battery-backed DRAM, some with NAND-Flash modules, and logical PMEM implementations that use DIMMs and NVMe drives.

The virtualization overhead that PMEM uses is nominal at 3 percent, according to VMware, and the benefits include up to eight times increased bandwidth and latency of less than 1 microsecond. The best performance can be achieved when applications are modified to use the PMEM device in a byte-addressable manner.

6
vSphere Network Design

To effectively design a virtual network infrastructure, a design architect must understand the virtual network architecture, including which features are available and how they are configured. This chapter will contain recipes that a design architect can use to design a virtual network architecture that provides the capacity and availability required to support the virtual infrastructure.

The logical network design includes calculating the network capacity (or bandwidth) required to support the virtual machines and determining the capacity that's required to support VMware technologies, such as vMotion and Fault Tolerance. If IP-based storage connectivity is required, the design must account for the networking that's necessary to support storage traffic, as well:



Network design in the vSphere design workflow

In this chapter, we will discuss the different virtual network switch technologies that are available in vSphere, and the different features that are available in each. This chapter will also cover how load balancing and teaming are used to improve network utilization efficiency and to increase availability. Network capacity resource management with traffic shaping, jumbo frames, and network I/O control will also be covered.

In this chapter, we will cover the following recipes:

- Determining network bandwidth requirements
- Standard or distributed virtual switches
- Providing network availability
- Network resource management
- Using private VLANs
- IP storage network design considerations
- Enabling jumbo frames
- Designing for VMkernel services
- Creating custom TCP/IP stacks
- vMotion network design considerations
- Using 10 GbE Converged Network Adapters
- IPv6 in a vSphere design
- Remote Direct Memory Access options

Determining network bandwidth requirements

Bandwidth refers to the capacity of the network, and it is measured in either **Gigabits per second** (**Gbps**) or **Megabits per second** (**Mbps**). The bandwidth that's required is based on the amount of data transferred or the throughput required by the virtual machines. Most modern networks are capable of transferring data at 1 Gbps or 10 Gbps. Network adapters that support 40 and 100 Gbps have recently become available.

The number of physical network adapters in each host that are required to support a solution is dependent on the amount of bandwidth required to support virtual machine network traffic, the number of virtual switches required, and the network redundancy requirements.

The following information from the case example in *Chapter 3*, *The Design Factors*, is used to help calculate the network bandwidth requirements:

- Cisco switches are used for network connectivity. Separate VLANs exist for management connectivity and production application connectivity

- No more than 20 application servers (or 200 customers) should be affected by hardware failure
- Currently, each physical server contains a single gigabyte network interface card. Peak network usage is 10 Mbps

How to do it…

Refer the following steps to determine bandwidth requirements.

1. Calculate the total amount of bandwidth required to support virtual machine network traffic using the following formula:

Total Number or Virtual Machines x Bandwidth per Virtual Machine (Mbps) = Total Bandwidth Requirement (Mbps)

2. Calculate the amount of bandwidth required per host. This is dependent on the maximum number of virtual machines that can be run on a single host.
3. Determine the network requirements for other vSphere services and features, such as vMotion, iSCSI, NFS, and Fault Tolerance.
4. Select the type and number of network adapters to provide the network connectivity that's required to support the design requirements.

How it works…

The physical network infrastructure must be capable of supporting the total throughput requirement of the environment. The total throughput requirement is calculated by multiplying the number of virtual machines by the throughput that's required by a single virtual machine:

100 Virtual Machines x 10 Mbps = 1,000 Mbps Total

The throughput that's required for a single host is calculated by multiplying the number of virtual machines that will run on a host by the throughput that's required by a single virtual machine:

20 Virtual Machines x 10 Mbps = 200 Mbps per Host

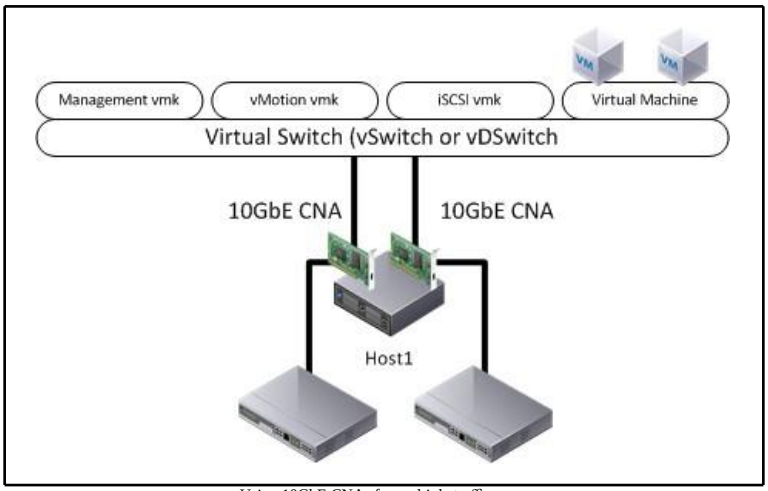
Network adapters are generally capable of delivering throughput equal to approximately 80 percent of the adapter speed, for example, 800 Mbps for a 1 Gbps network adapter. A single gigabit Ethernet connection would provide sufficient bandwidth to support the virtual machine throughput requirements that were calculated previously. An additional network adapter would be required to support failovers.

There are also network bandwidth requirements to support VMkernel interface network connectivity for management, vMotion, IP storage, and Fault Tolerance. The minimum bandwidth requirements for VMkernel network connectivity are as follows:

- Management: 100 MB
- vMotion: 1 GB
- IP storage: This is dependent on the amount of storage throughput required, but is limited to the bandwidth of a single path
- Fault Tolerance: 1 GB (10 GB required for multi-vCPU FT)

Sufficient physical network connectivity and bandwidth must be included in the design to support these services.

Network connectivity can be provided by using multiple 1 GB network adapters, or 10 GbE Converged Network Adapters (CNAs) can be used to carry multiple network traffic types, including virtual machine network traffic and VMkernel (management, vMotion, FT, and IP storage) network traffic on a single 10 GbE network adapter, as shown in the following diagram:



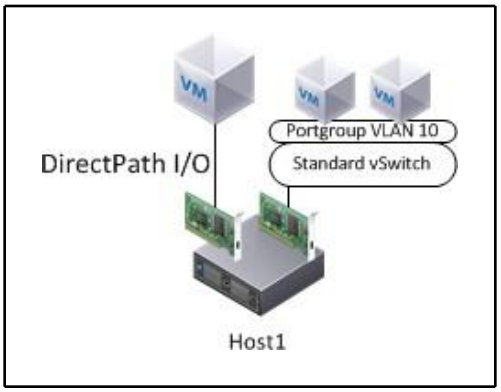
Using 10GbE CNAs for multiple traffic types

When CNAs are used to provide physical uplink connectivity to virtual switches, traffic shaping or **Network I/O Control** (**NIOC**) may be configured to ensure that sufficient network bandwidth is available to each traffic type serviced by the CNAs.

There's more…

For a virtual machine with very high network I/O requirements, DirectPath I/O allows a physical network adapter to be passed through directly to the virtual machine.

The following screenshot shows how a virtual machine is provided direct access to a physical network card with DirectPath I/O:



DirectPath I/O for a VM

When DirectPath I/O is used, the network adapter is only made available to the virtual machine that it is passed to, and cannot be shared with other virtual machines. The full bandwidth capacity of the network adapter is available to the virtual machine. Because a virtual machine with DirectPath I/O configured is dependent on the physical network card in the host, it cannot be moved to other hosts using vMotion, nor can it be protected using VMware HA.

Standard or distributed virtual switches

The connectivity of the virtual network to the physical network in a vSphere environment is accomplished by using one of two virtual switch technologies: the standard virtual Switch (vSwitch) or the virtual Distributed Switch (vDSwitch). VMware technologies like VMware HA, VMware DRS, and Fault Tolerance require that virtual switch configurations be consistent across all ESXi hosts in a cluster.

How to do it…

To determine whether to use standard or distributed virtual switches, follow these steps.

1. Identify the features and capabilities of virtual standard switches and distributed virtual switches
2. Based on the design requirements, determine which virtual switch technology should be selected to support them

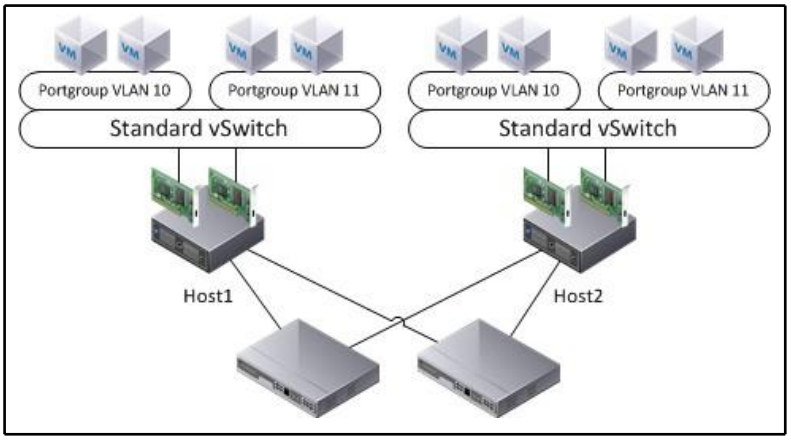
How it works…

The virtual switch technology that's chosen is dependent on the connectivity, availability, and manageability requirements, and the features that are available on the virtual switch.

A vSwitch is configured and managed independently on each ESXi host and supports up to 1,024 virtual switch ports per vSwitch. Because vSwitches are configured on each individual host, it increases the administrative overhead required to support large environments. Advanced network features, such as port mirroring, NetFlow integration, and private VLANs, are not available when using standard virtual switches. vSwitches are available at all vSphere license levels.

Several networks can use the same vSwitch, or the networks can be separated across multiple vSwitches. Multiple physical uplinks can be associated with a vSwitch to provide redundancy and load balancing. vSwitches can be created with no physical uplinks to keep virtual machine network traffic isolated on a single host.

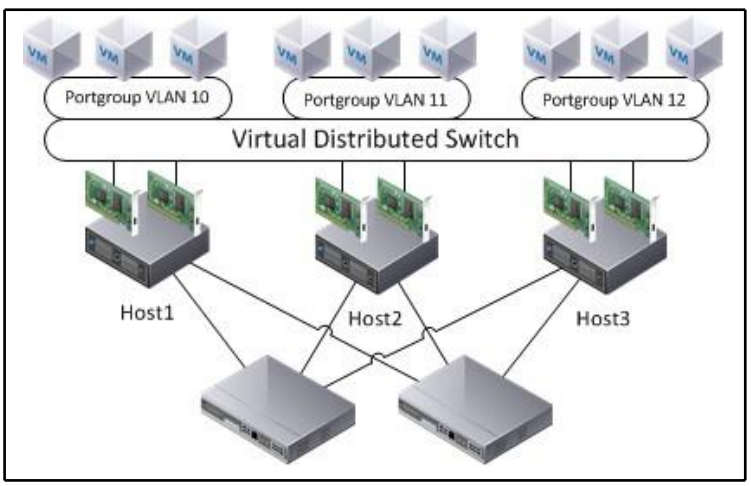
The following diagram depicts a common logical network design using standard virtual switches to provide virtual machine network connectivity:



A typical virtual standard switch network design

vDSwitches are configured and managed by vCenter. A vDSwitch guarantees consistent network policy configurations and PortGroup configurations across all hosts, with uplinks that are connected to it. vSphere Enterprise Plus Licensing is required to use vDSwitches.

Multiple physical uplinks from each host can be associated with a vDSwitch, to provide redundancy and load balancing. A vDSwitch will not be available for use by a host without any physical uplinks associated with it. The following diagram depicts a logical virtual network design using a virtual distributed switch:



A typical virtual distributed switch network design

vCenter is required to manage vDSwitches. vCenter controls the configuration state and keeps track of virtual machine connection information for the vDSwitch. If the vCenter Server managing the vDSwitch is unavailable, new connections and modifications to the vDSwitch will not be possible.

The features that are available when using a vDSwitch are as follows:

- The central management of the virtual switch and virtual machine PortGroups
- Link Aggregation Control Protocol (LACP)
- Ingress and egress traffic shaping
- Load balancing based on the physical NIC load
- NetFlow integration
- Port mirroring
- Third-party virtual switches (Cisco Nexus 1000v)
- Private VLANs (PVLAN)
- Network I/O Control

There's more…

Third-party virtual switches, such as the Cisco Nexus 1000v, can be used to extend the functionality of a vDSwitch. They provide an interface for provisioning, monitoring, securing, and configuring the virtual network, using standard vendor network management tools.

Providing network availability

Network availability is obtained by minimizing Single Points of Failure (SPOF) and providing sufficient capacity. Multiple network ports, network adapters, and physical switches can be used to minimize single points of failure, and link aggregation can be used to provide load balancing across multiple network adapters.

vSphere virtual network configurations offer multiple NIC teaming and load balancing options. The options that are used are dependent on the number of network adapters available, the number of virtual machines connected, the physical network's topology, and the amount of bandwidth required.

How to do it…

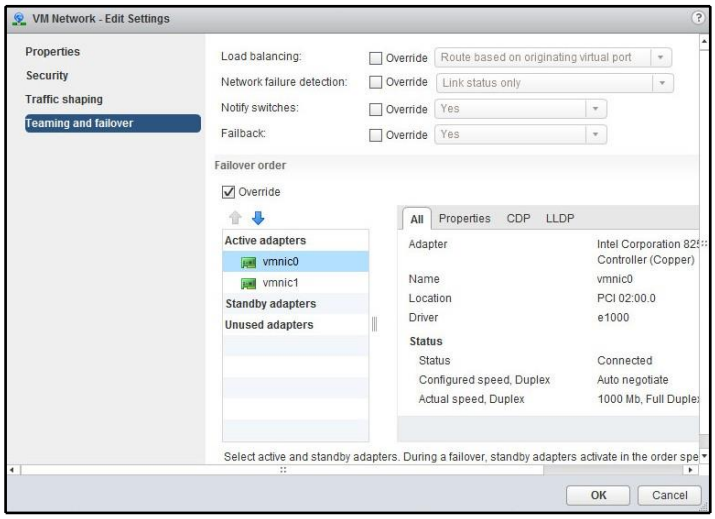
Follow these steps to design for network availability.

1. Identify the availability options that are available on virtual switches and virtual switch PortGroups
2. Determine the load balancing policies to provide availability based on design requirements
3. Determine the network adapter teaming policies to provide availability based on design requirements

How it works…

Load balancing distributes the network load across multiple available adapters. Load balancing policies are configured as a part of the NIC Teaming and failover options on virtual switches, virtual machine PortGroups, and VMkernel interfaces.

The following screenshot illustrates the Edit Settings dialog for configuring the Teaming and failover options on the virtual machine PortGroup on a standard virtual switch:



Configuring Teaming and Failover options for a portgroup

The following load balancing policies can be applied to virtual switches or virtual PortGroups:

- Route based on originating virtual port: This is the default load balancing policy. When it is being used, the load is balanced based on the number of physical NICs and the number of virtual switch ports in use. Virtual port connections are distributed across the physical NICs available to the virtual switch. A virtual machine connected to the virtual port will always use the same physical NIC, unless the NIC becomes unavailable. This is usually the best load balancing method to use.
- Route based on IP hash: This load balancing policy uses a hashing algorithm that determines the physical path based on the source and destination IP addresses of the virtual machine traffic. A virtual machine's network traffic can use multiple available NICs. This policy is used when using either EtherChannel or the LACP link aggregation.
- Route based on source MAC hash: This load balancing policy is similar to the Route based on originating virtual port policy, except that the physical NIC used for virtual machine traffic is based on the virtual network adapter's MAC address and not the virtual port connection.
- Use explicit failover order: This policy is not really a load balancing policy, because network traffic always uses the physical NIC uplink that is configured as the highest ordered active physical uplink available.
- Route based on physical NIC load: This is an additional load balancing option offered by vDSwitches that is not available to vSwitches. It is the most efficient because it distributes the load across active uplinks, based on the actual workload of the physical NICs.

Redundancy in the virtual network is provided by configuring the Failover order. These configurations define the physical uplinks that are actively used to pass network traffic, and those that are available stand in the event of an active uplink failing.

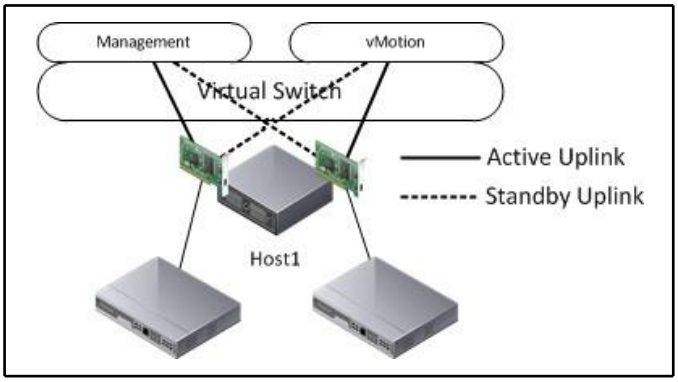
The available adapters are as follows:

- **Active adapters**: These are adapters that are available for use by the virtual switch, a virtual machine network PortGroup, or a VMkernel interface.
- **Standby adapters**: These are adapters that only become active in the event that an active adapter becomes unavailable.
- **Unused adapters**: These adapters are unused. They will never be used by the virtual switch, a virtual machine PortGroup, or a VMkernel interface.

The following screenshot shows the adapters that are configured in Active and Standby. If the Active Adapter (vmnic0) fails, the Standby Adapter (vmnic1) will become active:

Active / Standby configuration of network adapters

The following diagram shows an example of an active/standby network configuration that's commonly used in small environments to provide connectivity and redundancy for the host management and vMotion networks:



Logical view of an active / standby network configuration

Network Failover Detection and Failback are settings that control how a network failure is detected, and what happens when an active adapter is returned to service after a failure.

How network failure is detected is configured by using the Network Failover Detection option. Two failure detection options are available: Link Status Only and Beacon Probing. The Link Status Only option uses the link state (up or down) of the physical NIC to determine whether the uplink is available. The Beacon Probing option detects network failures by sending and receiving beacon probes out to all physical uplinks on the virtual switch, and it can detect link state and switch failures. At least three active uplinks are needed for beacon probing to work effectively. The VMware Knowledge Base article located at `http://kb.vmware.com/kb/1005577` provides more information on how beacon probing works with virtual switches.

The Failback setting defines whether an active adapter is returned to service if the adapter becomes available after a failure, based on the value set for Network Failover Detection. If a physical switch fails and Failback is enabled, with Link Status Only being used for Failover Detection, the adapter may become active and will be returned to service before the physical switch is available to pass traffic.

Network resource management

In a vSphere environment, physical network resources are shared across multiple virtual machines and services. The ability to ensure that sufficient capacity is available across shared resources is therefore important. If a single virtual machine or a VMkernel network service, such as vMotion or Fault Tolerance, saturates the available network capacity, other virtual machines and services, including host management services, may be adversely impacted.

How to do it…

Follow these steps to design a resource management scheme:

1. Identify the traffic shaping and network resource controls that are available on the virtual network switches.
2. Determine the network resources required for different traffic types: management, IP storage, vMotion, and virtual machine traffic.
3. Design traffic shaping, Network I/O Control policies, and Network Resource Pools to guarantee or limit network resources for the network traffic types, based on design requirements.

How it works…

Traffic shaping is used to limit the amount of bandwidth that's available to virtual switch ports. NIOC is used to apply limits and to guarantee traffic to different virtual network service types.

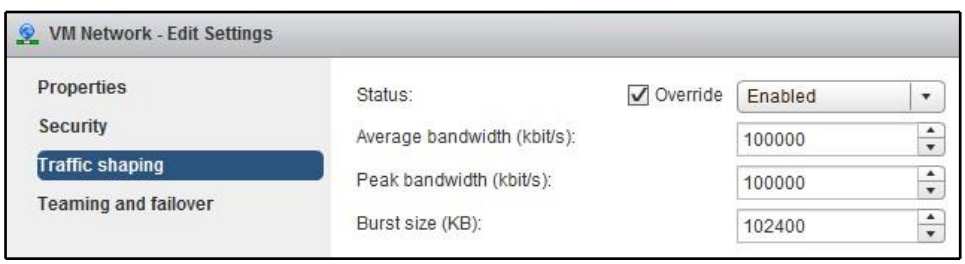
Traffic shaping can be configured on vSwitches, vDSwitch uplinks, VMkernel interfaces, and PortGroups to restrict the network bandwidth available to the network ports on the virtual switch. Traffic shaping is applied at all times, regardless of the amount of network capacity available. This means that if traffic shaping is enabled and configured on a virtual switch or PortGroup to limit the peak bandwidth to 1,048,576 Kbps (1 Gbps), only 1,048,576 Kbps of bandwidth will be used, even if more bandwidth is available.

The following bandwidth characteristics can be applied to the traffic shaping policy:

- Average bandwidth: This is the permitted average load, measured in kilobits/sec (Kbps)
- Peak bandwidth: This is the maximum allowed load, measured in kilobits/sec (Kbps)
- Burst size: This is the maximum number of bytes, measured in kilobytes, that can be burst over the specified average bandwidth

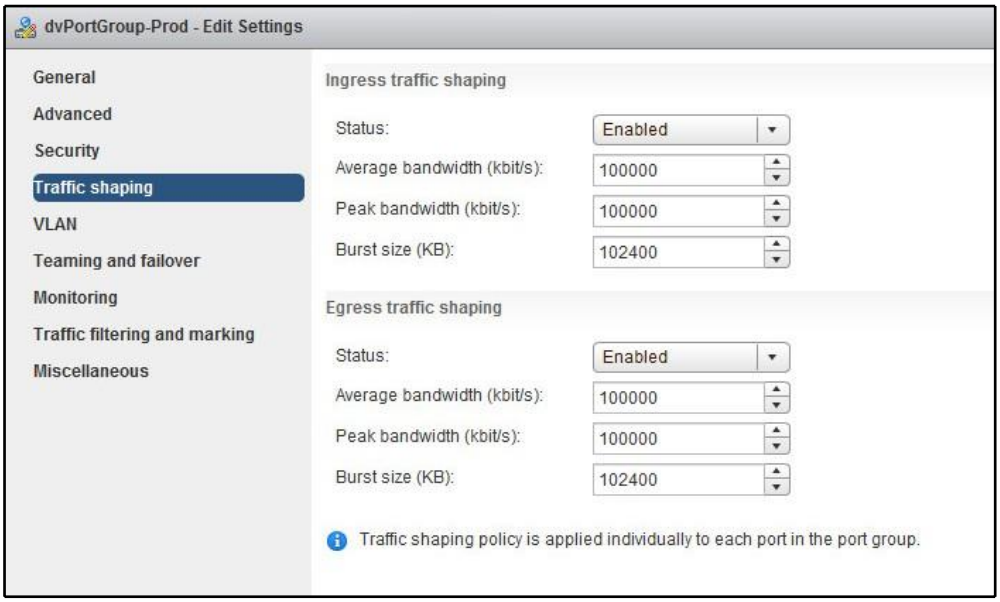
The traffic shaping policies on a vSwitch only apply to egress or outbound traffic. vDSwitch traffic shaping policies can be configured for both ingress (inbound) and egress (outbound) traffic.

The following is a screenshot of the configuration screen for applying Traffic shaping to a virtual machine PortGroup on a standard virtual switch:



Configuring traffic shaping on a VSS portgroup

The following screenshot shows the Ingress traffic shaping and Egress traffic shaping settings that can be applied to a virtual machine PortGroup on a vDSwitch:



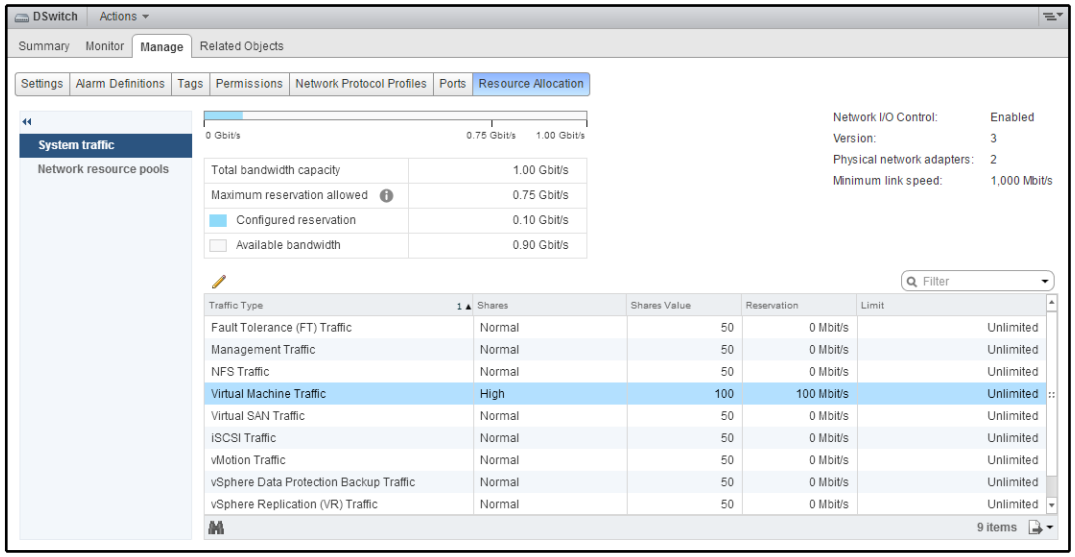
Configuring traffic shaping on a vDS portgroup

Unlike traffic shaping, Network I/O Control only provides control over the network bandwidth for specific network protocols during times of network contention. NIOC can only be enabled on vDSwitches.

Shares, limits, and reservations can be applied to network traffic types to limit and guarantee bandwidth to the different network traffic types, including the following:

- Management traffic
- vMotion traffic
- IP storage traffic (NFS/iSCSI)
- Virtual SAN traffic
- Fault Tolerance traffic
- vSphere replication traffic
- vSphere Data Protection backup traffic
- Virtual machine traffic

The System traffic screen in the vSphere Web Client provides information about the bandwidth capacity and the allocation of network resources across traffic types. The following screenshot illustrates the NIOC configurations of a vDSwitch:



NIOC configuration on a vDS

Shares define the share of the available bandwidth on a physical NIC that's attached to the vDSwitch during a time of network bandwidth contention that a specific traffic type will receive. The Reservation is the Mbps guaranteed to a specific traffic type. The Limit is the Mbps limit that's applied to all hosts connected to the vDSwitch.

The percentage of bandwidth that a traffic type receives is based on the total number of shares available; for example, in the default configuration, the virtual machine traffic receives *100* shares of the *400 (50 + 50 + 50 + 50 + 100 + 50 + 50)* shares available. The formula for this is as follows:

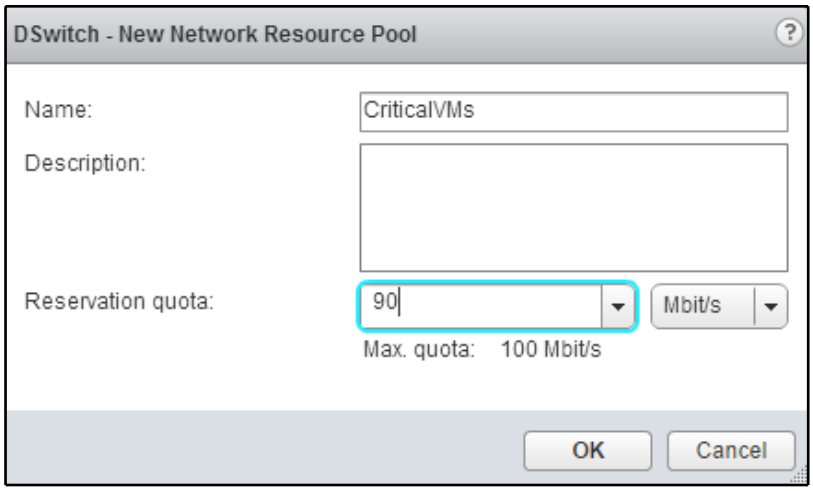
Shares Value / Total Available Shares = Percentage of Physical Network Bandwidth

Therefore, 100 / 400 = 25%

The amount of bandwidth that's available to vMotion would be calculated based on the 50 shares allocated to the vMotion traffic, as follows:

50 / 400 = 12.5%

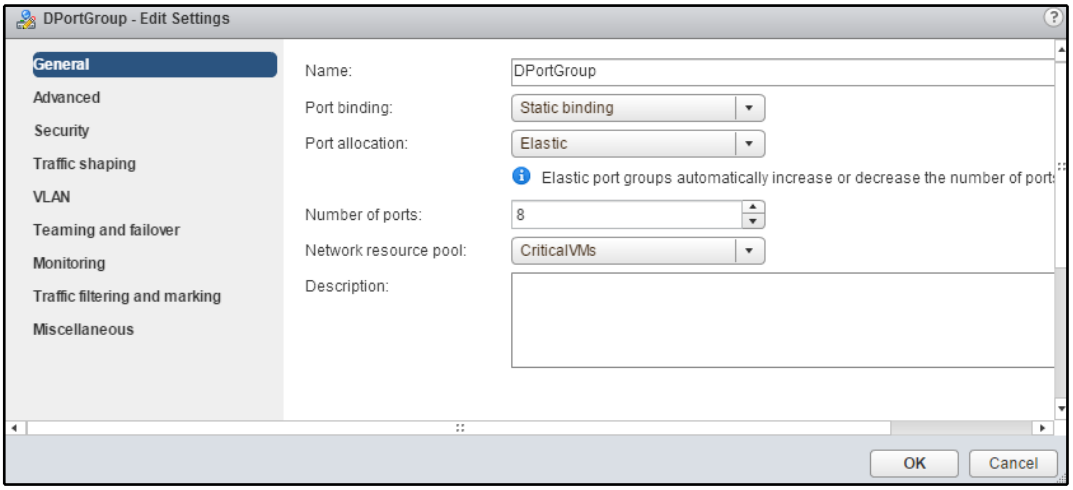
Network Resource Pools can be created to allocate virtual machine traffic reservations across distributed PortGroups. For example, if 100 Mbps is reserved for virtual machine traffic, this reservation can be applied across different PortGroups. In the following screenshot, a New Network Resource Pool is created, allocating 90 Mbps of the 100 Mbps reservation to a pool named CriticalVMs:



Configuring a network resource pool

Notice that the Reservation quota is the bandwidth that the Network Resource Pool will be guaranteed out of the overall reservation. The Reservation quota cannot be set to a value higher than the total reservation that's allocated to virtual machine traffic.

The Network resource pool is then assigned to a PortGroup on the virtual distributed switch, as shown in the following screenshot:



Assigning a network resource pool to a vDS portgroup

This allocates the reservation from the CriticalVMs network resource pool to the DPortGroup port group.

Using private VLANs

Private VLANs are an extension of the VLAN standard. PVLANs can be configured on virtual distributed switches to isolate traffic between virtual machines in the same VLAN.

How to do it…

Refer the following steps to design a private VLAN scheme:

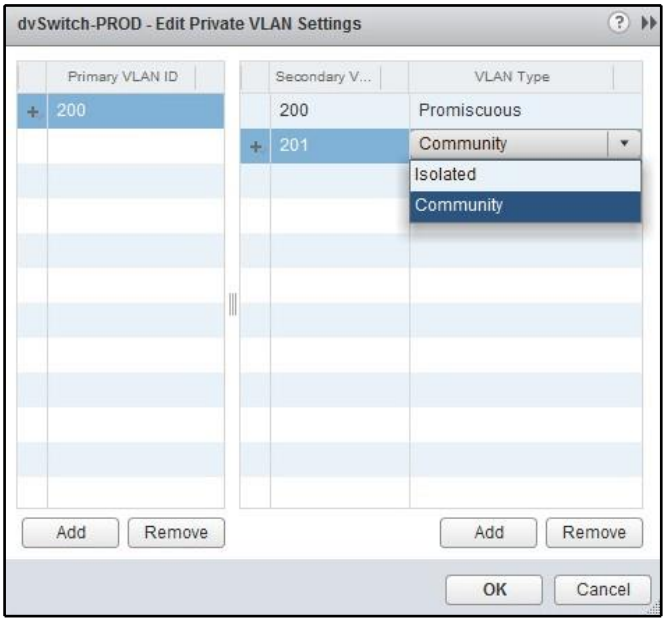
1. Identify the types of private VLANs that are available and the functionality of each
2. Determine the use cases for the PVLANs and identify whether the PVLANs can be used to satisfy the design requirements
3. Design the PVLANs to meet the design requirements

How it works…

A primary PVLAN is created on a vDSwitch, and secondary PVLANs are associated with the primary PVLAN. There are three types of secondary PVLANs: Promiscuous, Community, and Isolated:

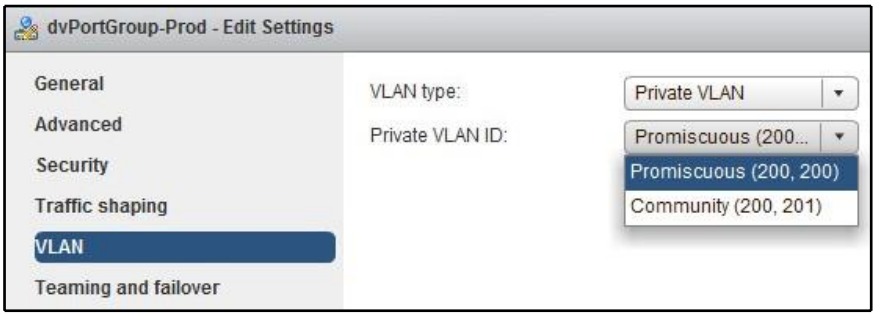
- The virtual machine connections in a Promiscuous PVLAN can communicate with all of the virtual machine connections in the same primary PVLAN. When a primary PVLAN is created, a Promiscuous PVLAN is created with the same PVLAN ID as the primary PVLAN.
- Virtual machine connections in a Community PVLAN can communicate with other virtual machine connections in the same Community PVLAN and virtual machine connections in the Promiscuous PVLAN. Multiple Community PVLANs can be associated with a single primary PVLAN.
- The virtual machine connections in an Isolated PVLAN can only communicate with the virtual machine connections in the Promiscuous PVLAN. Only one Isolated PVLAN can exist per primary PVLAN.

Private VLANs are created by editing the settings of a vDSwitch, as follows:



Editing a private VLAN

Once the PVLAN has been configured on the vDSwitch, a virtual machine network PortGroup is created with the PVLAN type and ID assigned, as follows:



Configuring a private VLAN on a vDS portgroup

There's more…

For PVLAN traffic to be passed between ESXi hosts connected to a vDSwitch, the physical switch must be PVLAN-aware and properly configured to support PVLANs. The process to configure the PVLANs on a physical switch will vary from vendor to vendor. The following process shows the steps that are necessary to configure PVLANs on a Cisco IOS switch:

1. Enter the Cisco switch configuration mode:

```
switch# configure terminal
```

2. Enable the PVLAN feature on the switch:

```
switch(config)# feature private-vlan
```

3. Create the PVLAN on the switch and set the PVLAN type:

```
switch(config)# vlan <vlan-id>
switch(config-vlan)# private-vlan primary
```

4. Associate the secondary PVLANs with a primary VLAN:

```
switch(config-vlan)# private-vlan association <secondary
pvlan>
```

5. The switch ports that are connected to the vDSwitch uplinks need to be configured to allow for the PVLAN traffic:

```
switch(config)# interface GigabitEthernet1/1
switch(config-if)# switchport mode trunk
switch(config-if)# switchport trunk allowed vlan
<vlan/pvlan ids>
```

IP storage network design considerations

iSCSI, NFS, and **Fiber Channel over Ethernet** (**FCoE**) are IP-based storage protocols that are supported in a vSphere environment. This recipe will cover the design considerations for designing the IP networks that will be used for storage traffic.

How to do it…

Refer the following steps to design an IP storage network:

1. Identify the network connectivity and virtual switch configurations that are required for IP-connected storage
2. Determine the best practices for providing connectivity for IP-connected storage
3. Design the IP storage connectivity to meet design requirements

How it works…

IP storage traffic should be separated from other IP traffic. This separation can be provided by either using physically separate hardware (network adapters and physical switches) or by using separate VLANs for IP storage traffic. The networks associated with IP storage should be directly connected and non-routable.

Multiple network paths to storage should be configured to provide redundancy and load balancing. Single points of failure should be minimized so that the loss of a single network path does not result in the loss of storage connectivity.

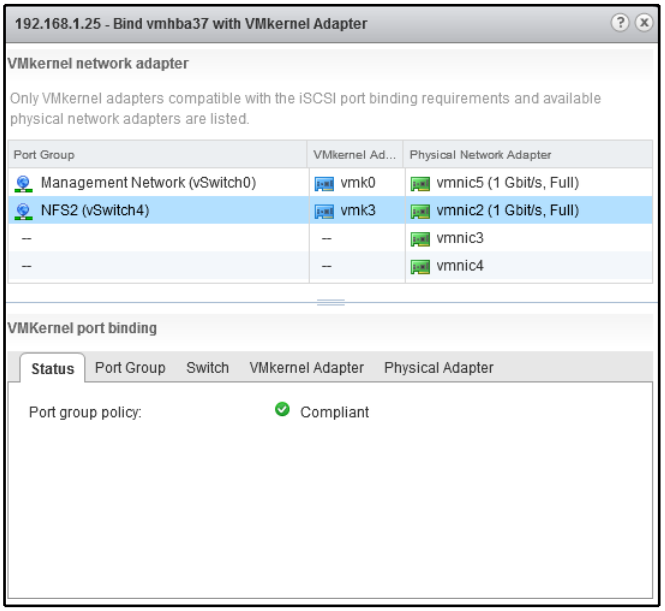
Software iSCSI, NFS, and Software FCoE each require a VMkernel interface to be created on a virtual switch. The VMkernel interfaces that are used for iSCSI and FCoE must be bound to a single physical active adapter.

Having more than one active adapter, or a standby adapter, is not supported with Software iSCSI or Software FCoE.

NFS v3 over TCP does not provide support for multipathing. Using link aggregation will only provide failover, and not load balancing. NFS will always use a single physical network path, even if multiple VMkernels are configured. To manually load balance NFS traffic, create separate VMkernel ports that are connected to separate networks and mount separate NFS v3 datastores.

NFS v4.1 supports multipathing for NFS servers, which support session trunking. Multiple VMkernel ports can be configured to provide access to a single NFS volume that's been configured with multiple IP addresses. This provides load balancing and resiliency for NFS v4.1.

The VMkernel port binding for Software iSCSI is configured in the properties of the Software iSCSI adapter. Only VMkernel ports that are compliant will be available for binding, as shown in the following screenshot:



Port-binding compliant VMkernel ports

To enable the Software FCoE adapter, an NIC that supports FCoE offloads must be installed on the host. If a supported NIC is not installed, the ability to add the software FCoE adapter will not be available.

Physical network binding for FCoE is done when enabling a Software FCoE adapter. Compliant and supported physical adapters are available when adding the Software FCoE adapter. Separate FCoE adapters should be enabled and connected to each storage network fabric. A single ESXi host can support up to four software FCoE adapters. Each software FCoE adapter requires a dedicated VMkernel port bound to a dedicated physical adapter.

Using jumbo frames

Enabling jumbo frames on the networks that's used for vMotion or IP storage can increase performance and throughput. When jumbo frames are configured, iSCSI or NFS packets can be transferred over the network in a single frame; there is no fragmentation. This decreases the amount of CPU overhead that's necessary to encapsulate (and de-encapsulate) IP storage packets.

How to do it…

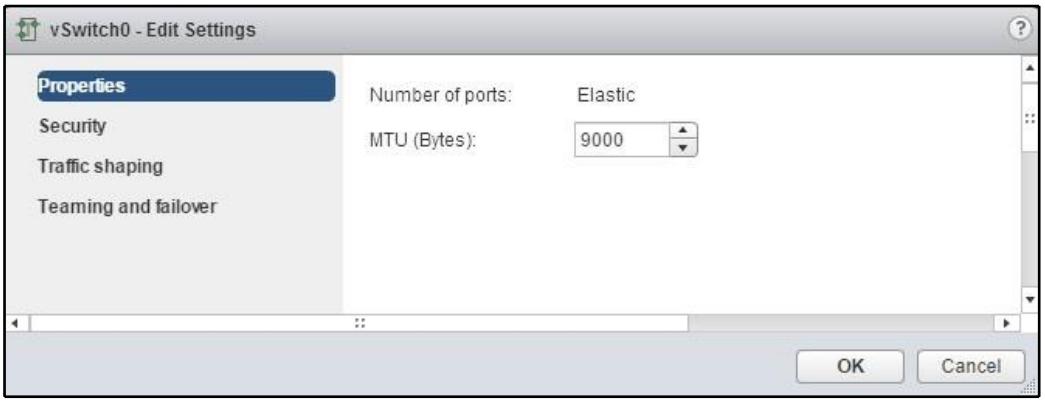
Refer the following steps to design and implement jumbo frames:

1. Determine the use cases for enabling jumbo frames
2. Configure the jumbo frames on virtual switches
3. Configure the jumbo frames on VMkernel ports
4. Ensure that jumbo frames are configured end to end on the physical network, that is, physical switches and array network interfaces
5. Test the network for proper end-to-end jumbo frame configuration

How it works…

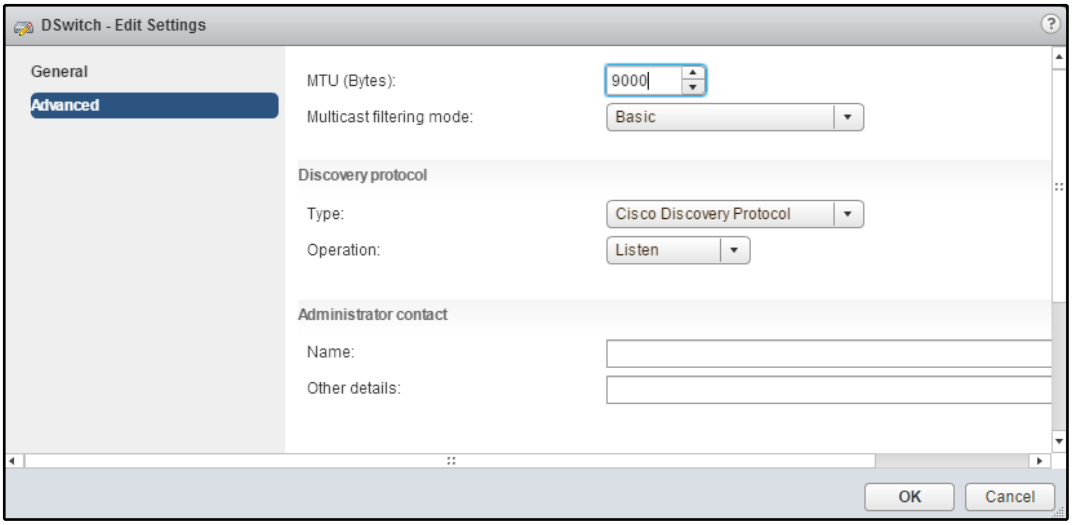
Jumbo frames must be supported and enabled on the network from end to end; this includes the physical network infrastructure as well. In vSphere, jumbo frames are enabled either in the vSwitch configuration or on the vDSwitch uplinks by setting the value **Maximum Transmission Unit** (**MTU**) to 9000. Jumbo frames must also be enabled on VMkernel interfaces by setting the value of MTU for the PortGroup to 9000.

To enable jumbo frames, set the value of MTU (Bytes) on the vSwitch to 9000, as shown in the following screenshot:



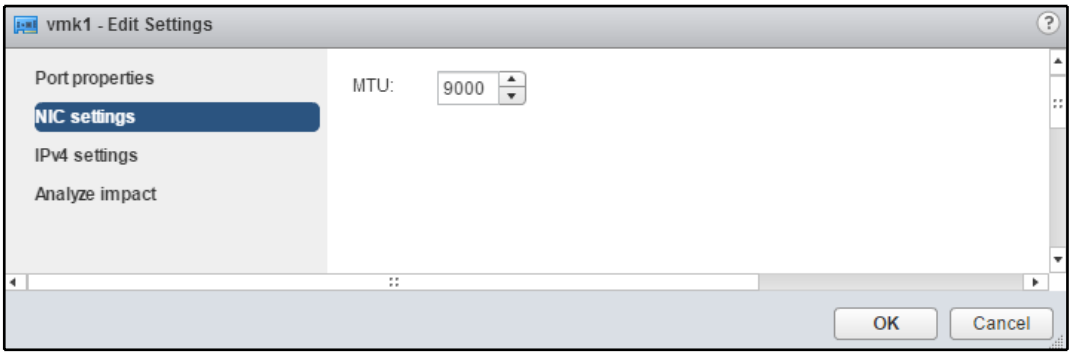
Editing the MTU of a VSS

If you are using a vDSwitch, the MTU (Bytes) is set to 9000 in the Advanced settings to enable jumbo frames:



Editing the MTU of a vDS

The MTU setting must also be changed to 9000 on a VMkernel interface on the vSwitch or vDSwitch to enable jumbo frames, as shown in the following screenshot:



Editing the MTU of a VMkernel port

When you are using jumbo frames, the physical switch must also be configured to support the MTU. This will vary, depending on the switch vendor and version. To enable jumbo frames on a Cisco Catalyst Series switch, use the following command:

```
Switch(config)# system mtu jumbo 9000
```

In this case, the switch must be reloaded for the setting to take effect. Other switches may allow (or require) per-port MTU configuration.

If you are using jumbo frames for the storage network, the jumbo frames will need to be enabled on the network interfaces of the array. The process for this will vary greatly between array vendors. If the array interfaces are not configured correctly, the traffic may not pass, or the performance will be significantly impacted.

The jumbo frame configuration can be tested from the ESXi shell by using the vmkping command, with the **Data Fragment** (**DF**) bit (`-d`) and size (`-s`) options set, as follows:

```
ESX1 # vmkping -d -s 8972 <IP_Address_of_IP_Storage_Array>
```

If jumbo frames are not configured correctly, the `vmkping` will fail.

Creating custom TCP/IP stacks

TCP/IP stacks provide flexibility in the VMkernel interface design by allowing you to apply specific DNS and default gateway configurations to a VMkernel interface on a host.

There are three preconfigured TCP/IP stacks, as follows:

- Default TCP/IP stack: Supports management traffic
- vMotion TCP/IP stack: Supports the live migration, vMotion, of virtual machines
- Provisioning TCP/IP stack: Supports the cold migration, cloning, and snapshot creation of virtual machines

Custom TCP/IP stacks can be used to handle the network traffic of other applications and services, which may require separate DNS and default gateway configurations.

How to do it…

Refer the following steps to implement custom a TCP/IP stack:

1. Create a custom TCP/IP Stack on an ESXi host
2. Configure DNS, default gateway, and advanced settings on the TCP/IP Stack
3. Assign the TCP/IP Stack to a VMkernel adapter

How it works…

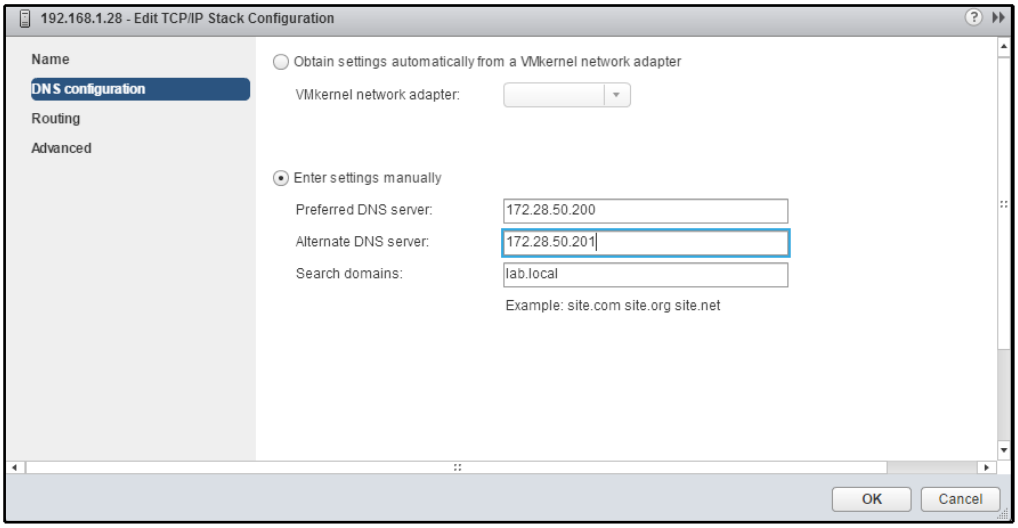
Using TCP/IP stacks for VMkernel network traffic provides the following benefits:

- It separates VMkernel routing tables
- It provides a separate set of buffers and sockets
- It isolates traffic types to improve performance and security

Currently, a custom TCP/IP stack cannot be created in the Web Client interface. A custom TCP/IP stack is created by using esxcli on the ESX host, as follows:

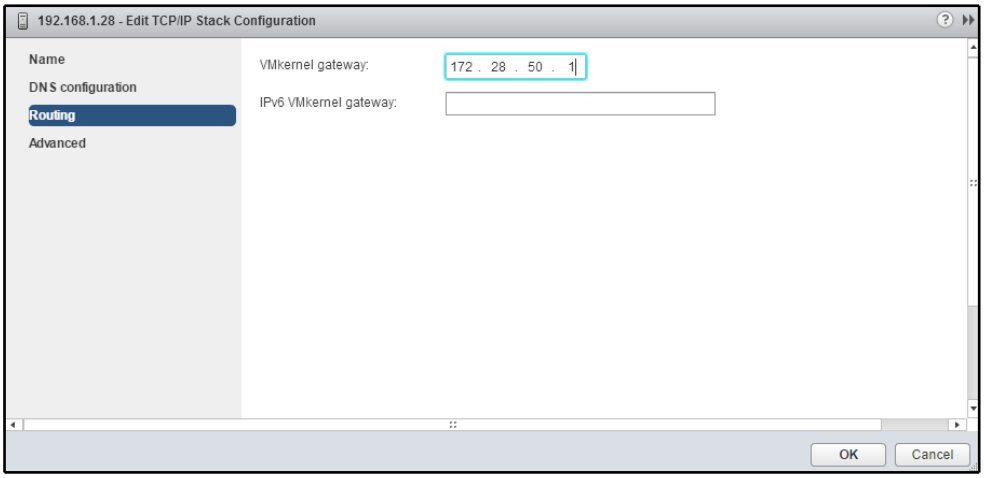
```
ESX1 # esxcli network ip netstack add -N "Name_of_Stack"
```

The DNS configuration associated with the TCP/IP stack can then be configured. This can automatically be obtained from a VMkernel adapter by using DHCP, or it can be set manually, as shown in the following screenshot:



Editing the DNS servers of a TCP/IP stack

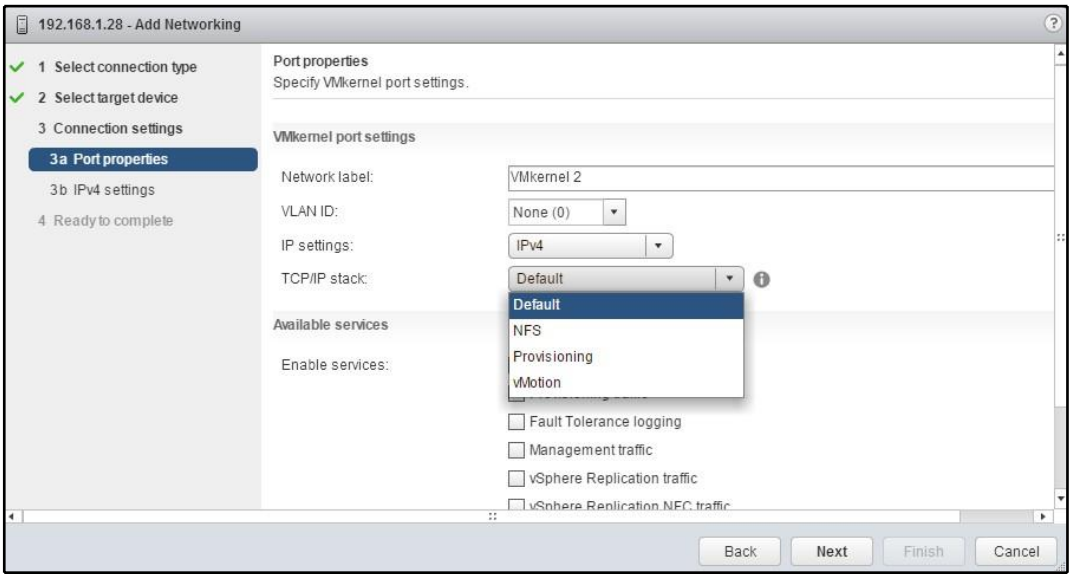
A VMkernel gateway can be assigned to the TCP/IP stack, as shown in the following screenshot:



Editing the gateway of a TCP/IP stack

Advanced TCP/IP stack settings include the maximum number of connections and the congestion control algorithm to use for the stack.

The TCP/IP stack is assigned to a VMkernel adapter when it is created, as shown in the following screenshot:



Assigning a TCP/IP stack to a VMkernel adapter

Designing for VMkernel services

VMkernel interfaces are configured to provide network connectivity for services in the vSphere environment. The VMkernels provide network paths for service connectivity. Multiple VMkernel interfaces can be created to provide a physical or logical separation for these services.

How to do it…

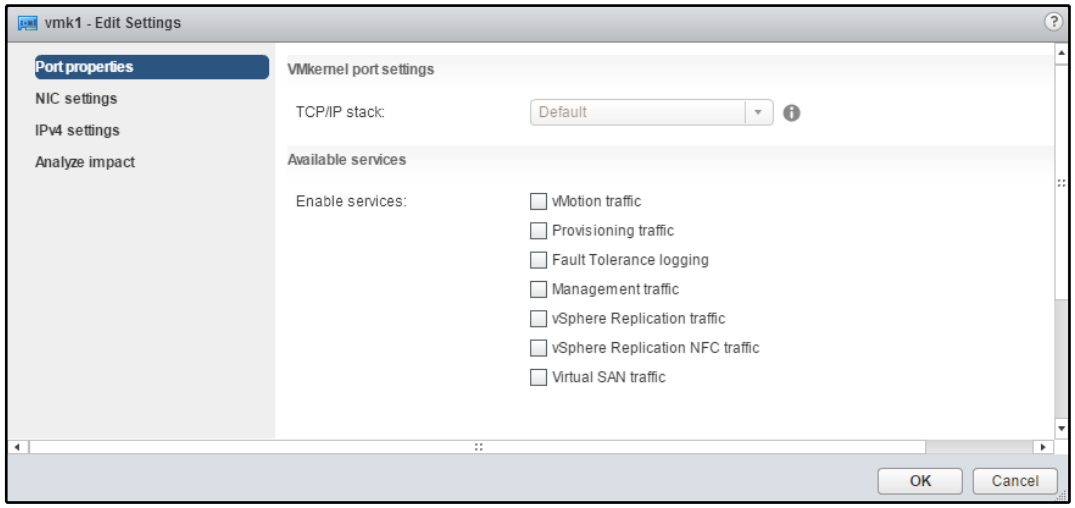
Refer the following steps to design and implement VMkernel services:

1. Identify services that require a VMkernel interface
2. Create a VMkernel interface to support the service
3. Enable services on the VMkernel interface

How it works…

Most vSphere services require a VMkernel interface to provide network connectivity. These services include the following:

- ESXi Management
- vMotion
- Fault Tolerance
- Virtual SAN
- vSphere Replication
- IP Storage (NFS, iSCSI, FCoE)
- Multiple services can share a single VMkernel port, or the services can be separated across multiple VMkernel ports for performance, management, and security. Services can be enabled on VMkernel interfaces at the time of creation, or by editing the Port properties, as shown in the following screenshot:



Services supported for VMkernel ports

Once the services have been enabled, the VMkernel interface will provide connectivity for the services that are selected. As we discussed in the previous recipe, TCP/IP stacks can be used to configure specific DNS settings and a default gateway for a service.

vMotion network design considerations

vMotion allows for the running state of a virtual machine to be transferred from one ESXi host to another. The network traffic that's required for the migration uses the VMkernel interfaces that have been enabled for vMotion. vMotion connectivity between ESXi hosts is required when using a **Distributed Resource Scheduler** (**DRS**) to balance the virtual machine load across hosts in a DRS-enabled cluster.

How to do it…

The following steps to design networking for vMotion:

1. Identify the vMotion network requirements
2. Determine the best practices for configuring the network connectivity that's required to support vMotion
3. Identify the benefits of keeping virtual machines together on the same host to minimize the network traffic that must transverse the physical uplinks
4. Design the vMotion network connectivity to support design requirements
5. Design DRS rules to support design requirements

How it works…

vMotion requires, at a minimum, a single, active, 1 GB network adapter. A second standby adapter should be configured to provide redundancy for the vMotion network.

A vMotion migration can consume all of the available network bandwidth. If the vMotion network is shared with other network traffic, traffic shaping or NIOC should be enabled to prevent vMotion from impacting other virtual network traffic. If possible, vMotion should be configured on a separate physical network or separate VLAN.

vSphere 5 introduced the ability to configure multiple adapters for use with vMotion. Multiple-NIC vMotion allows for the bandwidth of multiple physical NICs to be leveraged by vMotion to speed up the migration of virtual machines between hosts.

To configure Multiple-NIC vMotion, create multiple VMkernel interfaces with vMotion enabled. Configure each VMkernel interface to use a single active adapter, and configure other available adapters as standby adapters. When a virtual machine is vMotioned, either manually or by DRS, all available links will be used for the vMotion traffic. More information on Multiple-NIC vMotion can be found in the VMware Knowledge Base article at `http://kb.vmware.com/kb/2007467`.

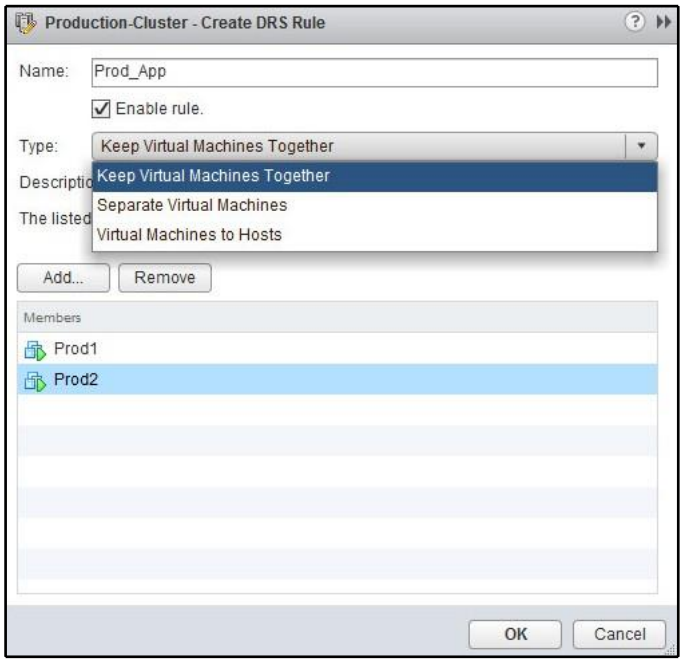
There's more…

Network communications between virtual machines that are connected to the same virtual network on the same ESXi host will not use the physical network. All of the network traffic between the virtual machines will remain on the host.

Keeping virtual machines that communicate with each other together on the same host will reduce the load on the physical network; for example, keeping a web frontend server, application server, and database server together on the same host will keep the traffic between the servers internal to the host.

If the VMware DRS is configured on a vSphere Cluster, DRS rules can be configured on the cluster to keep virtual machines together on the same host.

In the following screenshot, a Virtual Machine Affinity Rule has been created to keep two virtual machines together on the same host:



Creating a DRS rule

With DRS enabled, the virtual machines assigned to the DRS rule will be vMotioned to run on the same host.

Virtual machine anti-affinity rules (Separate Virtual Machines) can also be configured to ensure that virtual machines run on separate hosts. This can be used when service redundancy is provided by multiple virtual machines, such as with multiple virtual domain controllers. Keeping virtual machines separate will ensure that a host failure does not impact service redundancy.

Using 10 GbE converged network adapters

Computing has historically bounced between performance bottlenecks. When one technology evolves to perform better, it leaves an older technology to become the system's worst performing characteristic. Between memory caches, SSDs, high bandwidth networks, and software, the bottleneck constantly moves. With **10 Gb Ethernet** (**10 GbE**) networks today, we find it's usually not the result of too little bandwidth. But how can we ensure that we design our virtual data center with the highest performance when using 10 GbE **Converged Network Adapters** (**CNAs**)?

CNAs allow both Ethernet **Local Area Network** (**LAN**) traffic and **Fibre Channel over Ethernet** (**FCoE**) traffic to use the same physical network adapter, as opposed to using dedicated Ethernet and Fibre Channel adapters. Similar to how virtualization allowed for the maximum use of server hardware resources and reduced wasted resources, CNAs allow for the maximum use of bandwidth by pushing storage traffic over the same physical cable as LAN traffic. It's important to consider, however, how each type of traffic is managed on the physical link so that congestion is handled appropriately.

How to do it…

The best way to manage congestion in vSphere networking is likely Network I/O Control. Follow these steps to design a virtual data center with 10 GbE CNAs:

1. Identify all of your IP traffic types; for example, management, vMotion, Fault Tolerance, VM traffic, FCoE, and so on
2. Assign NIOC shares for each traffic type

How it works…

Choosing NIOC shares that ensure that storage traffic is not impacted is critical, because poor storage performance can ripple through an environment quickly, causing ill effects. Traffic types like vMotion and Fault Tolerance are important too, but their weight is less than that of storage or VM traffic. Their NIOC shares should reflect this.

IPv6 in a vSphere design

Internet Protocol version 6 (**IPv6**) was developed to replace **IP version 4** (**IPv4**). IPv6 addresses are 128-bit IP addresses, compared to the 32-bit addresses in IPv4. IPv6 is becoming more common in data center network environments, and vSphere has included support for IPv6 since vSphere 5.x.

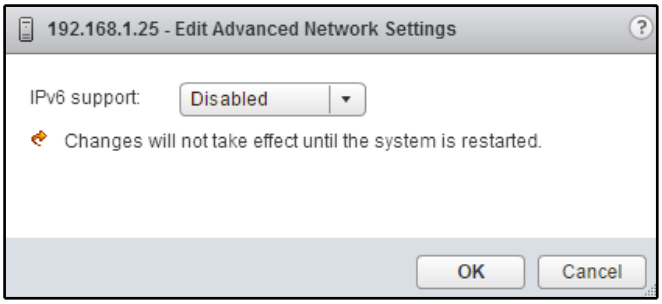
How to do it…

Refer the following steps to design and implement IPv6:

1. Enable IPv6 on the ESXi host
2. Determine the vSphere features and services with IPv6 support
3. Configure the VMkernel interfaces to use IPv6

How it works…

By default, IPv6 support is enabled on ESXi hosts. If the IPv6 support is changed, disabled, or enabled, a host reboot is required. Enabling or disabling IPv6 is done on each ESXi host by editing the Advanced Network Settings from the Networking management tab for the host, as shown in the following screenshot:



Enabling IPv6 support

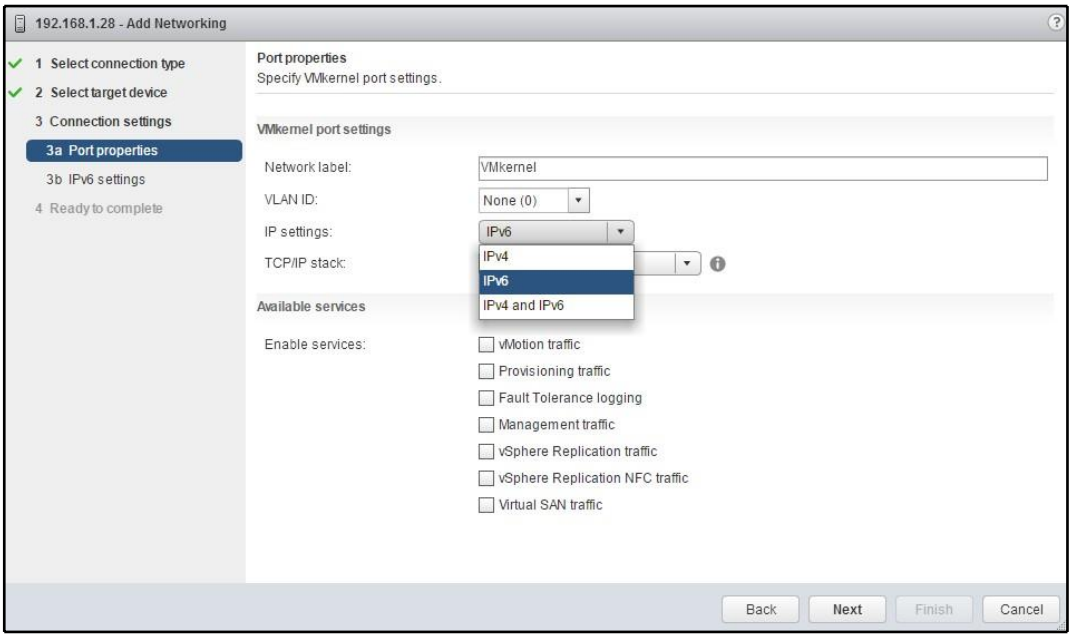
Once enabled, IPv6 can be configured for supported vSphere features and services. The following vSphere features and services support IPv6:

- ESXi and vCenter management
- vMotion and vSphere DRS
- Fault Tolerance
- vSphere HA
- NFS v3 storage
- iSCSI (software or hardware)

IPv6 is not currently supported with the following vSphere features:

- Auto deploy
- DPM with IPMI/iLO
- Virtual volumes
- Virtual SAN
- Authentication proxy
- NFS v4.1

When IPv6 is enabled on an ESXi host, VMkernel interfaces can be created with IPv4, IPv6, or both IPv4 and IPv6 settings, as shown in the following screenshot:



Choosing IP versions for a VMkernel port

IPv6 addresses can be configured automatically by using DHCP or router advertisement, or the IPv6 address can be a static address that is manually assigned.

Remote direct memory access options

Since vSphere 6.5, Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) has been supported. RoCE provides extremely low latency and high throughput communication over an Ethernet network, and allows one VM to access the memory contents of another VM directly, without the involvement of the hosts' CPU. This is usually reserved for network-intensive applications. Like FCoE, it requires a lossless network. RoCE requires hardware support in the form of Host Channel Adapters (HCAs) when VMs communicate across different ESXi hosts. RDMA is built into ESXi, and therefore, HCAs are not required when VMs reside on the same ESXi host. As of vSphere 6.7, only some Linux distributions support RoCE, such as guest operating systems running virtual hardware version 13 or later. The RoCE support in vSphere 6.5 (and later) is named **Paravirtual RDMA** (**PVRDMA**).

How to do it…

Refer the following steps to implement RDMA in vSphere:

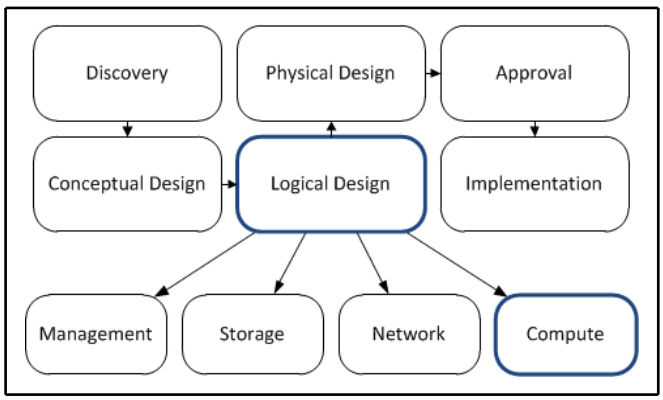
1. Install a supported Host Channel Adapter
2. Configure PVRDMA on each host with an HCA
3. Assign a PVRDMA network adapter to a VM

How it works…

When configured with PVRDMA adapters, VMs with network-intensive applications can communicate over RDMA. When two VMs need to communicate across ESXi hosts and each host has an HCA installed, PVRDMA communication is enabled. If two VMs with PVRDMA adapters need to communicate on the same ESXi host, the HCA, if installed, is not used, and traffic is kept within the VMkernel itself. If a VM with RDMA enabled needs to communicate with a VM that does not support RDMA, the communication falls back to TCP/IP.

7
vSphere Compute Design

This chapter will cover logical compute design. **Compute** refers to the processor and memory resources required to support the virtual machines running in the vSphere environment. Calculating the required CPU and memory resources is an important part of the design process and ensures that the environment will be able to support the virtual machine workloads. Design decisions like scaling up, scaling out, and clustering hosts will be covered. The following diagram displays how the compute design is integrated into the design process:



Compute design in the vSphere design workflow

In a physical environment where a single operating system or a single application is installed on a dedicated physical hardware, compute utilization usually averages as 10-20 percent of the available resources. The majority of the memory and CPU resources are idle and wasted. In a virtual environment, the resources that are available are utilized by multiple operating systems and applications. It is not uncommon to see a usage of 65-80 percent of the available resources.

We will take a look at the clustering hosts' resources to take advantage of the advanced VMware features: vSphere **High Availability** (**HA**), vSphere **Distributed Resource Scheduler** (**DRS**), and vSphere **Fault Tolerance** (**FT**). Ensuring that significant resources are available for failover and provide vMotion compatibility are key factors of cluster design. Methods to reserve or limit resources and for providing flash-based caching will also be covered.

In this chapter, we will cover the following recipes:

- Calculating CPU resource requirements
- Calculating memory resource requirements
- Transparent page sharing
- Scaling up or scaling out
- Determining the vCPU-to-core ratio
- Clustering compute resources
- Reserving HA resources to support failover
- Using distributed resource scheduling to balance cluster resources
- Ensuring cluster vMotion compatibility
- Using resource pools
- Providing Fault Tolerance protection
- Leveraging host flash

Calculating CPU resource requirements

There are several factors that must be considered when calculating CPU resource requirements, such as the amount of CPU resources that are required to support the current workloads, the amount of CPU resources required to support future growth, and the maximum CPU utilization threshold.

The following information from *Chapter 3*, *The Design Factors*, will be used to calculate the CPU requirements:

- Currently, there are 100 physical servers, each hosting a single application. Each application services 10 customers.
- The business expects to add 50 new customers over the next year.

- Support growth over the next five years.
- Each application server is configured with two dual-core 2.7 GHz processors. The peak usage of a single application server is approximately 10 percent of the total, or approximately 1 GHz.

How to do it…

Refer the following steps to calculate CPU resource requirements:

1. Determine the amount of CPU resources required to support the current workloads:

Number of Workloads x CPU Speed in MHz or GHz = Current CPU Resources Required

2. Determine the maximum utilization threshold. This is the maximum percentage of available resources that should be consumed.
3. Determine the amount of growth in CPU resources that the environment should support.
4. Calculate the amount of CPU resources required:

Current Workload CPU Resources + Future Growth + Maximum Utilization Threshold = Total CPU Resources Required

How it works…

Calculating the required CPU resources that are necessary to support the current workload is straightforward and uses the following formula:

Number of Workloads x CPU Speed in MHz or GHz = Current CPU Resources Required

100 x 1 GHz = 100 GHz

To determine the total CPU resources required, the amount required to support future growth must also be calculated. The amount of CPU resources required for future growth will be determined by the design requirements. Based on the requirements identified in *Chapter 3, The Design Factors*, the environment should be designed to support a growth of 25 additional virtual machines over the next five years.

A maximum utilization threshold must also be determined and accounted for in CPU resource requirements. This threshold determines the maximum percentage of total CPU resources that will be consumed. It is unlikely that the environment would be configured to consume 100 percent of the CPU resources available. If the maximum utilization threshold is 75 percent, an additional 25 percent of CPU resources will be added to calculate the total CPU resources required:

Current CPU Resources Required + Future Growth = Total CPU Resources Required

100 GHz + (25 x 1 GHz) = 125 GHz

When the maximum utilization threshold is 75%, the calculation will be as follows:

*125 GHz * (100/75) = ~167 GHz*

The environment must be designed to support the 167 GHz of CPU resources that are, in turn, required to support the current workloads and future workloads, and to provide for a maximum utilization threshold of 75 percent.

Calculating memory resource requirements

There are several factors that must be considered when calculating memory requirements; these factors include the amount of memory required to support the current workloads, the amount of memory required to support future growth, the amount of memory required for virtual machine memory overhead, and the maximum memory utilization threshold.

Like the CPU requirements, information from *Chapter 3*, *The Design Factors*, will also be used to calculate the memory requirements:

- Currently, there are 100 physical servers, each hosting a single application. Each application services 10 customers.
- The business expects to add 50 new customers over the next year.
- Support growth over the next five years.
- Each application server is configured with 8 GB of memory. The peak usage of a single application server is approximately 65 percent, or around 5.2 GB.

How to do it…

Refer the following steps to calculate memory resource requirements:

1. Determine the amount of memory resources required to support the current workloads:

Number of Workloads x Memory Usage = Current Memory Required

2. Determine the memory overhead required.
3. Determine the maximum utilization threshold. This is the maximum percentage of available resources that should be consumed.
4. Determine the amount of growth in the memory resources that the environment should support.
5. Calculate the amount of memory resources required:

Current Workload Memory Usage + Memory Overhead + Future Growth + Maximum Utilization Threshold - TPS Savings = Total Memory Resources Required

How it works…

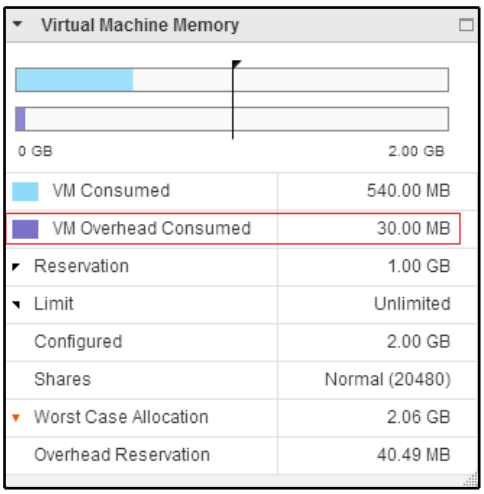
To calculate the amount of memory required to support the current workloads, the following formula is used:

Number of Workloads x Memory Usage = Current Memory Required

100 x 5.2 GB = 520 GB

The memory overhead of a virtual machine must also be accounted for when calculating memory requirements. The amount of memory required for an overhead depends on the configuration of the virtual machine.

The number of vCPUs allocated to the virtual machine, the amount of memory allocated to the virtual machine, and the virtual hardware configured for the virtual machine, will all have an impact on the amount of memory required for overhead:



Memory overhead example

Typically, the memory overhead required for a virtual machine is between 20 MB and 150 MB. Memory overhead estimations based on the amount of RAM and the number of vCPUs can be found in the vSphere documentation, as follows:

- **vSphere 6.0**: `https://pubs.vmware.com/vsphere-60/topic/com.vmware.vsphere.resmgmt.doc/GUID-B42C72C1-F8D5-40DC-93D1-FB31849B1114.html`
- **vSphere 6.5/6.7**: `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.resmgmt.doc/GUID-4954A03F-E1F4-46C7-A3E7-947D30269E34.html?hWord=N4IghgNiBcILYFM4HsBOBPABMgbg1AFgmACYgC+QA`

This may seem like a small amount of memory, but over dozens (or even hundreds) of virtual machines, it can have a significant impact on the amount of total memory required:

(Number of Workloads x Memory Usage) + (Number of Workloads x Memory Overhead) = Current Memory Required

(100 x 5.2 GB) + (100 x 50 MB) = 525 GB

To calculate the total memory required, future growth must be considered. When memory is calculated for growth, the memory overhead that's required to support the additional virtual machines must also be considered.

The maximum utilization threshold must also be determined for memory resources. This threshold defines the maximum percentage of the total memory resources that will be consumed. If the maximum utilization threshold is 75%, an additional 25% of memory resources will need to be added to calculate the total memory resources required:

*Current Memory Required + Future Growth * (100/Maximum Threshold%) = Total Memory Resources Required*

*525 GB + [(25 x 5.2 GB) + (25 x 50 MB)] * (100/75) = ~875 GB*

875 GB of memory is required to support the current workloads, the future growth, and a maximum utilization threshold of 75%.

Transparent page sharing

Transparent Page Sharing (**TPS**) is a memory saving technology used by vSphere that allows for duplicate memory pages to be shared between virtual machines. To address security concerns about sharing memory between virtual machines across security domains, TPS can be disabled, enabled across groups of virtual machines with the same salt settings, or enabled across all virtual machines in the environment.

How to do it…

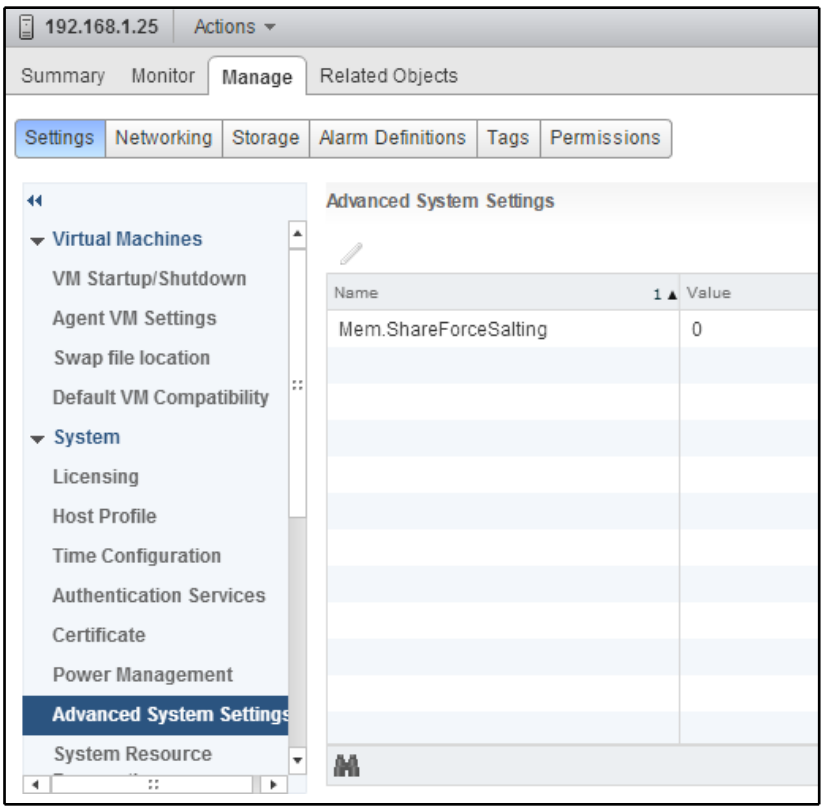
Refer the following steps to design and implement TPS:

1. Identify the different options for sharing duplicate memory pages within a virtual machine and across groups of virtual machines
2. Configure TPS to meet requirements for the security and performance of the environment
3. Configure salt values on virtual machines to enable or disable page sharing between virtual machines

How it works…

TPS de-duplicates pages of memory both within a virtual machine (Intra-VM) and across virtual machines (Inter-VM). By default, Inter-VM TPS is disabled due to security concerns about sharing memory pages between virtual machines that cross security boundaries (for example, virtual machine guests within the DMZ and virtual machine guests in the production environment). TPS can be configured to allow for Inter-VM sharing between all virtual machines, or only across certain groups of virtual machines, by adding a salt value to the virtual machines. The VMware KB article at `http://kb.vmware.com/kb/2097593` provides more information about changes and enhancements to TPS.

The host advanced configuration option, `Mem.ShareForceSalting`, can be set to configure how TPS will be used. This setting is configured per ESXi host, as shown in the following screenshot:



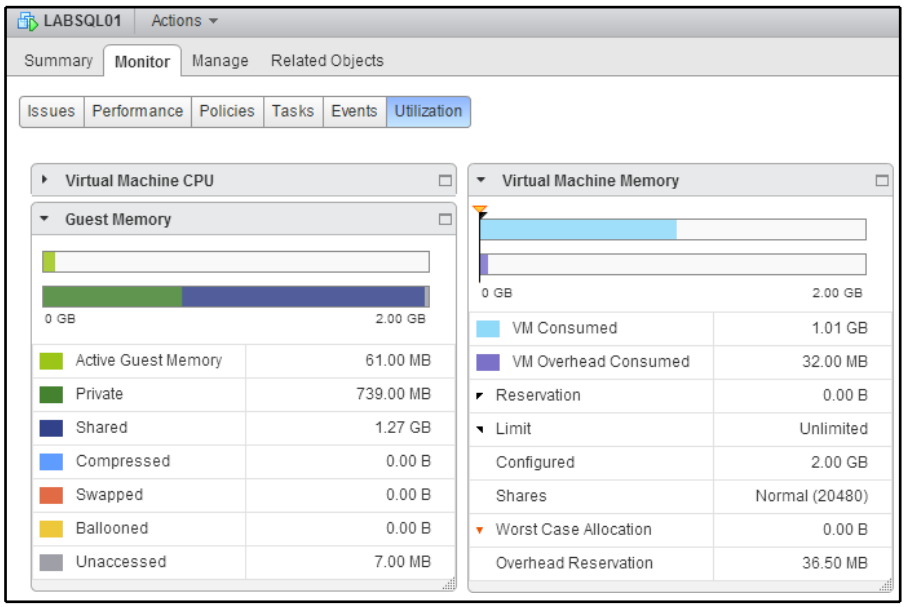
Enabling advanced settings for TPS

`Mem.ShareForceSalting` can be set to a value of `0`, `1`, or `2`. Add the configuration parameter `sched.mem.pshare.salt` to a virtual machine to set the salt value. Page sharing can be configured to only share pages between virtual machines with the same salt values.

The following table outlines how Intra-VM and Inter-VM page sharing is impacted based on the `Mem.ShareForceSalting` setting:

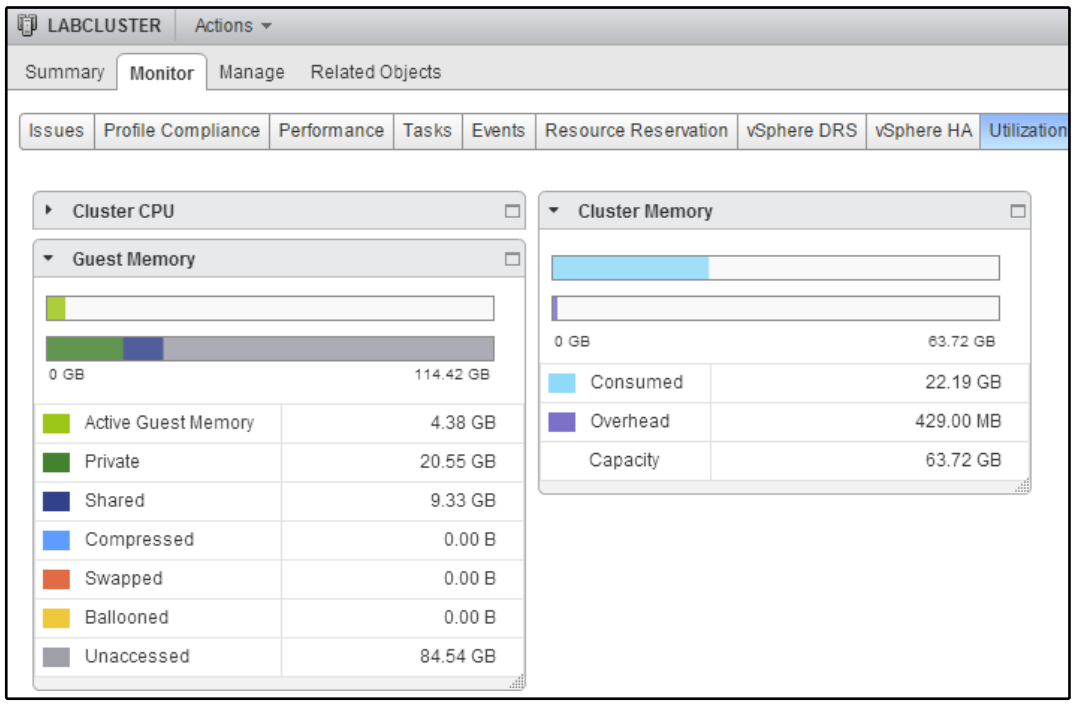
`Mem.ShareForceSalting` settings	Inter-VM sharing	Intra-VM sharing
`0`	Yes, between all virtual machines on the host. Virtual machine salt value is ignored.	Yes
`1`	Sharing between virtual machines with the same `sched.mem.pshare.salt` setting. Sharing between virtual machines where `sched.mem.pshare.salt` is not present.	Yes
`2` (default)	Only among virtual machines with the same `sched.mem.pshare.salt` setting. The virtual machine `vc.uuid` is used as the salt value by default.	Yes

The following screenshot displays the memory utilization for a specific virtual machine, showing the amount of shared memory:



VM memory utilization and shared memory

Notice the savings that the shared memory provides compared to the VM consumed memory, which is the amount of physical memory consumed on the host. The following screenshot shows the TPS savings across a vSphere cluster:



TPS savings on a vSphere cluster

There's more…

When large memory pages are used, TPS only provides a benefit when there is memory pressure on the host. When the memory utilization on a host reaches 95%, large pages are broken down into small pages to enable TPS. Large memory pages can be disabled on the ESXi hosts. This is done by setting `Mem.AllocGuestLargePage` to `0`. This configuration must be done on each host. Disabling large pages will increase sharing and decrease the amount of physical memory required, but it can have a performance impact, especially with memory-intensive workloads.

Scaling up or scaling out

Once the total CPU and memory resource requirements have been calculated, the amount of resources per host must be determined. Host resources can be designed based on two resource-scaling methodologies: scaling up or scaling out.

When scaling up, fewer, larger hosts are used to satisfy the resource requirements. More virtual machines run on a single host; because of this, more virtual machines are also affected by a host failure.

When scaling out, many smaller hosts are used to satisfy the resource requirements. Fewer virtual machines run on a single host, and fewer virtual machines will be affected by a host failure.

How to do it…

Refer the following steps to design for scaling up or scaling out:

1. Determine whether the host in the environment should scale up or scale out.
2. Determine the number of virtual machine workloads per host.
3. Based on the number of virtual machines per host, calculate the number of hosts that are required. This should also include the number of hosts required to support growth and failover:

(Number of Workloads / Number of Workloads per Host) + (Number of Future Workloads / Number of Workloads per Host) + Number of Failover Hosts = Number of Physical Hosts Required

4. Using the identified CPU requirements, calculate the CPU resources required per host:

Total CPU Resources Required / (Number of Physical Hosts Required - Failover Hosts) = CPU Resources per Host

5. Using the identified memory requirements, calculate the memory resources that are required per host:

Total Memory Resources Required / (Number of Physical Hosts Required - Failover Hosts) = Memory Resources per Host

How it works…

Many CPU and memory resources were calculated in the earlier recipes in this chapter, and are as follows:

- The total number of CPU resources required is 167 GHz
- The total number of memory resources required, taking into account a 25% savings for transparent page sharing, is 657 GB

Based on the design factors, the determination can be made on whether a host should be designed to scale up or scale out. In this case, the following design information provides what is needed to size the individual host resources:

- Currently, there are 100 physical servers, each hosting a single application. Each application services 10 customers
- No more than 20 application servers (or 200 customers) should be affected by a hardware failure
- The business expects to add 50 new customers over the next year
- Support growth over the next five years

Based on the requirements, the total number of hosts that are required to support the current workloads, the future workloads, and the redundancy requirements can be calculated as follows:

Total Hosts Required = (100 physical servers / 20 virtual servers per host) + [((50 new customers x 5 years) / 10) / 20] + 2 failover hosts = 8.25 = 9 Physical Hosts Required

Use the following equation to determine the number of CPU resources that are required per host (the failover hosts are not included here because these resources are effectively reserved for failover):

167 GHz / 7 = 23.8 GHz CPU per Host

Use the following equation to determine the number of memory resources that are required per host (as with CPU resources, the failover hosts are not included in the calculation):

657 GB / 7 = ~94 GB Memory per Host

Each physical host will need to be sized to support 20 virtual machines, and will require 23.8 GHz of CPU resources and 94 GB of memory resources.

There's more…

The requirements from *Chapter 3*, *The Design Factors*, are very specific about the maximum number of virtual machines that can be run on a host. This simplifies the scale-up or scale-out design decision. The following are a couple of other possible design requirements to work through to demonstrate the impact that scaling up and scaling out will have on resources:

- What if a requirement was to virtualize the environment using three hosts? What resources would be required for each host? If there are 100 virtual machines, how many will be impacted during a host hardware failure?
- What if the requirement was that each host should be configured with resources to support no more than 10 virtual machines? How will that change the number of resources required for each host? If there are 100 virtual machines, how many will be impacted during a host hardware failure?

Determining the vCPU-to-core ratio

The number of virtual machine vCPUs allocated compared to the number of physical CPU cores available is the vCPU-to-core ratio. Determining this ratio will depend on the CPU utilization of the workloads.

If the workloads are CPU-intensive, the vCPU-to-core ratio will need to be smaller; if the workloads are not CPU-intensive, the vCPU-to-core ratio can be larger. A typical vCPU-to-core ratio for server workloads is about 4:1—four vCPUs allocated for each available physical core. However, this can be much higher if workloads are not CPU-intensive.

A vCPU-to-core ratio that is too large can result in high CPU ready times—the percentage of time that a virtual machine is ready but is unable to be scheduled to run on the physical CPU—which will have a negative impact on the virtual machine's performance.

How to do it…

Refer the following steps to determine the vCPU-to-core ratio:

1. Determine the number of vCPUs that are required, as follows:

vCPUs per Workload x Number of Workloads Per Host = Number of vCPUs Required

2. Determine the vCPU-to-core ratio based on the CPU utilization of the workloads. If the workloads are CPU-intensive, the vCPU-to-CPU-core ratio will be lower; for less CPU-intensive workloads, the ratio will be higher. The ratio of 4:1 is generally a good starting point for server workloads.
3. Calculate the number of CPU cores that are required to support the vCPU-to-CPU-core ratio:

Number of vCPUs / vCPU-to-core ratio = Number of Cores Required

How it works…

The vCPU-to-core ratio is calculated based on the number of vCPUs allocated and the number of physical CPU cores available. For example, if two vCPUs are allocated to each virtual machine, the following applies:

2 vCPUs allocated to each virtual machine x 20 virtual machines = 40 vCPUs

In a design with 40 vCPUs that requires a 4:1 vCPU-to-core ratio, a minimum of 10 physical cores would be required.

If dual 8-core processors are used, the vCPU-to-core ratio can be calculated as follows:

2 x 8 Cores = 16 Total Cores

40 vCPUs and Physical 16 Cores = 2.5 vCPUs to each physical core, or a 2.5:1 vCPU-to-core Ratio

Clustering compute resources

A **vSphere cluster** is a group of ESXi hosts. The CPU, memory, storage, and network resources of each host are combined to form a logical set of cluster resources. A vSphere cluster is required to facilitate the use of features such as vSphere HA, vSphere DRS, and Fault Tolerance.

A single vSphere 5.x cluster can contain up to 32 hosts. For vSphere features such as vSphere HA and DRS to work correctly, the configurations must be consistent across all hosts in the cluster. The consistency of shared storage and network configurations is a necessity.

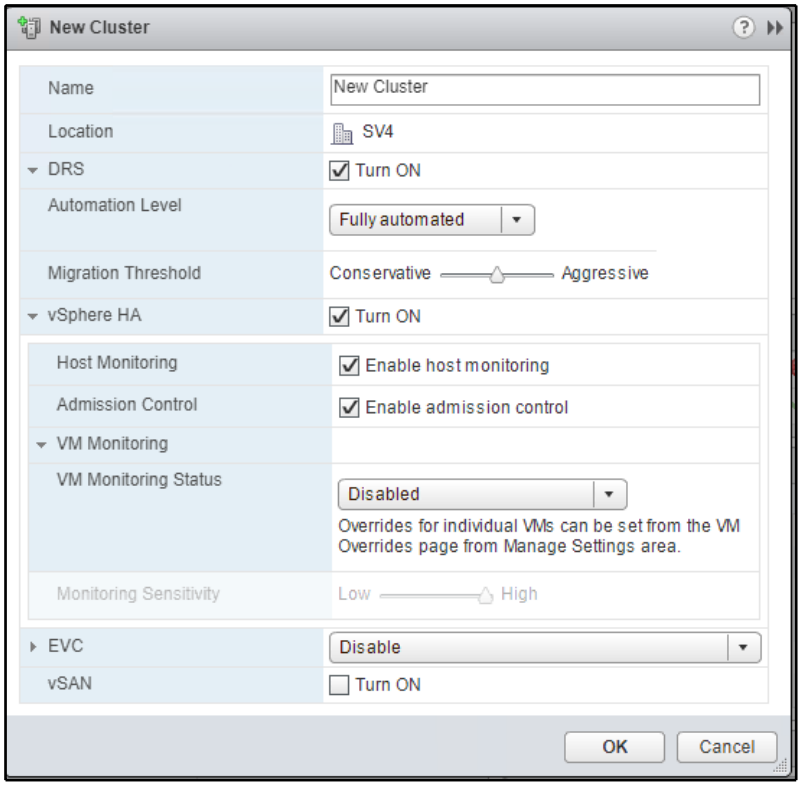
How to do it…

Refer the following steps to create a vSphere cluster:

1. Using the vSphere Web Client, create a new vSphere cluster
2. Enable vSphere High Availability on the cluster
3. Enable vSphere Distributed Resource Scheduling on the cluster

How it works…

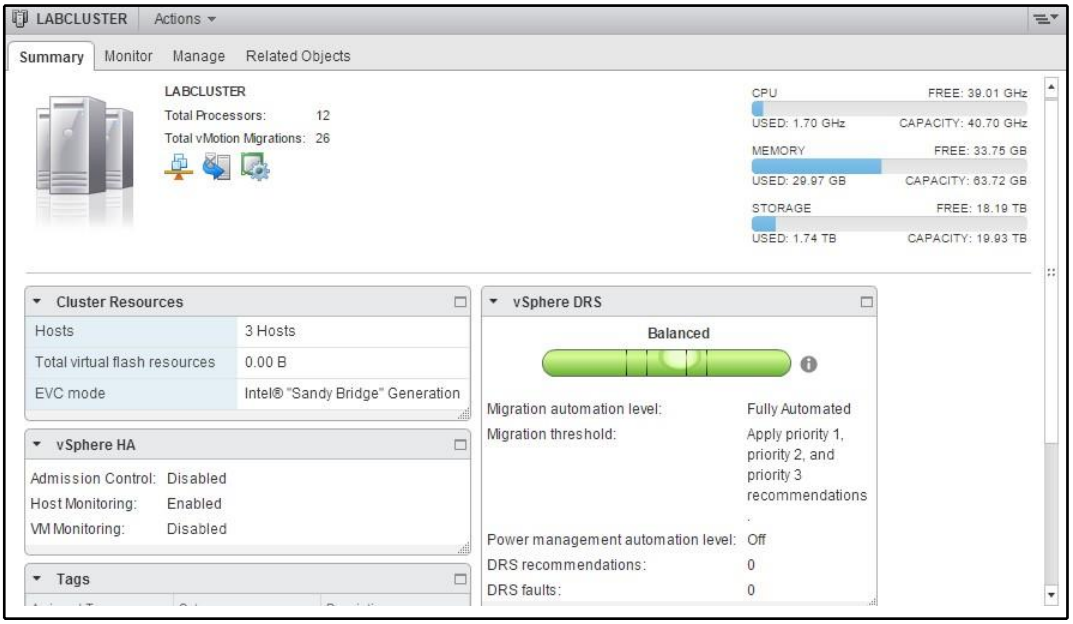
A new cluster is created by using either the vSphere Web Client or the vSphere Client. Right-click on the data center object in which you want the cluster to be created and select **New Cluster**. The **New Cluster** dialog will open, as shown in the following screenshot:



Creating a new cluster

A name must be provided for the cluster. Other cluster options, such as enabling DRS and vSphere HA, can be configured during the new cluster creation, or can be configured at a later time by editing the properties of the cluster.

Once the cluster has been created, hosts can be added to the cluster. New hosts are added to the cluster with the **Add Host** wizard if you right-click on the cluster and select **Add Host**. Existing hosts can be added to the cluster by dragging and dropping the host inventory object into the new cluster. The cluster's **Summary** tab displays the available cluster resources, the cluster resource usage, and details about the vSphere DRS and vSphere HA configurations:



Cluster summary tab

Hosts within a cluster should be configured with similar compute resources. In a cluster where the VMware DRS is enabled, processor compatibility is required. Checking for processor compatibility will be covered later in this chapter.

Reserving HA resources to support failover

When vSphere **High Availability** (**HA**) has been enabled on a vSphere cluster, the virtual machines running on the cluster are protected from a host hardware failure or virtual machine guest operating system crash.

In the event that a host suffers a hardware failure, or if ESXi crashes, the virtual machines are restarted on the surviving hosts in the cluster. Resources must be reserved in the cluster to guarantee that the necessary resources are available to restart the virtual machines.

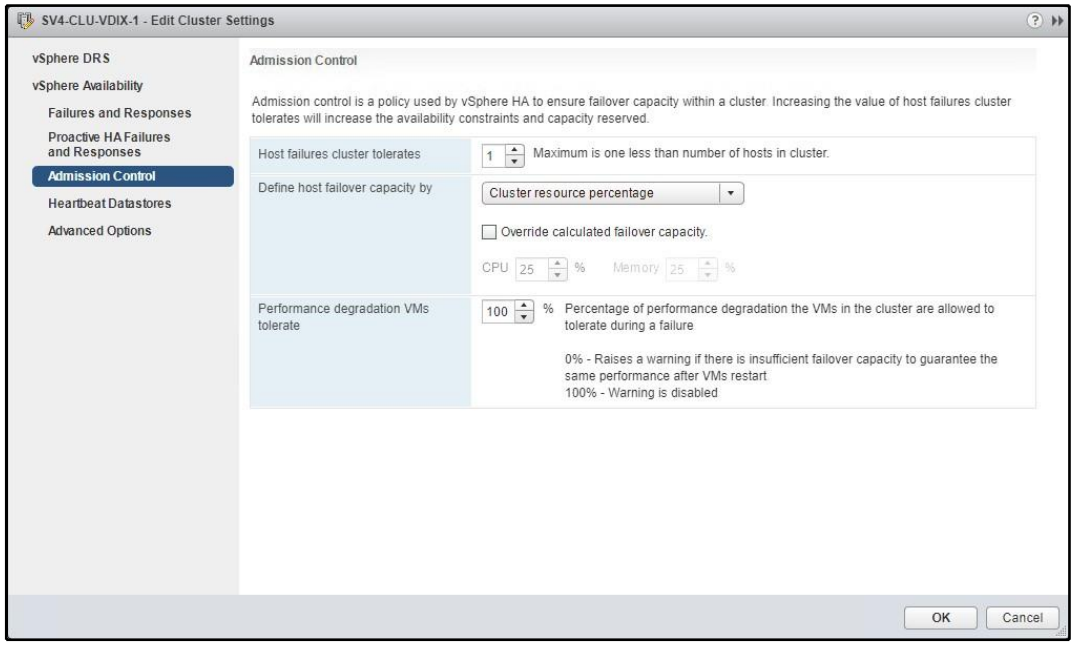
How to do it…

Refer the following steps to reserve HA resources to support failover:

1. Edit the settings of the vSphere cluster to enable high availability
2. Enable the HA Admission Control policy
3. Select the HA Admission Control policy that should be applied to the cluster
4. Define the failover settings that are required, based on the HA Admission Control policy that's selected

How it works…

VMware HA Admission Control ensures that enough physical resources are available to meet the CPU and memory reservation requirements needed to restart the virtual machines on surviving hosts, in case there is a host failure:



Configuring cluster admission control settings

When HA Admission Control is enabled, virtual machines cannot be powered on if there are insufficient resources to meet the reservation requirements for the virtual machines that are protected in the HA cluster. The resource requirements are calculated based on the HA Admission Control policy that's selected.

In vSphere 6.7, there are three HA admission control policies:

- Define failover capacity by static number of hosts
- Define failover capacity by reserving a percentage of the cluster resources
- Use dedicated failover hosts

The **Define failover capacity by static number of hosts** policy (for vSphere client: **Host failures cluster tolerates**) reserves failover resources based on the slot size. The slot size is determined by the largest CPU and memory reservation for a virtual machine that has been powered on. The number of slots available in the cluster and the number of slots to be reserved based on the failover capacity selection are calculated by HA.

A single virtual machine with a large memory or CPU reservation will have an impact on the number of slots available. The value of **Fixed slot size**, which can be configured using the vSphere Web Client, defines the amount of CPU and memory resources that make up a slot. In the versions of vSphere prior to 5.1, the slot size could be configured with the vSphere Client by setting the HA advanced options as `das.slotCPUInMHz` for CPU resources and as `das.slotMeminMB` for memory resources.

Using the **Define failover capacity by reserving a percentage of the cluster resources** policy (for vSphere Client: **Percentage of cluster resources reserved as failover spare capacity**) allows for a percentage of the memory and CPU resources to be reserved to accommodate a host failure. This reservation is distributed across all hosts in the cluster. To guarantee resource availability in the event of a host failure, the percentage should be set to reserve the CPU and memory resources equal to a single host in a cluster; for example, for a five-host cluster, 20 percent of cluster resources should be reserved. This will guarantee that enough resources are available to support a single host failure.

The **Use dedicated failover hosts** policy (for vSphere Client: **Specify failover hosts**) reserves a configured host to be available for failover. The hosts that are specified as failover hosts will not provide resources to virtual machines during normal operations. The host is a hot spare, and virtual machines will only be started on the hosts.

If HA Admission Control is disabled, virtual machines can be powered on, even if there are not enough resources available to ensure failover capacity. If the surviving hosts are not able to provide the resources with the necessary reservations to start the virtual machines, the virtual machines will not be restarted when a host fails.

Using distributed resource scheduling to balance cluster resources

The vSphere DRS determines the initial placement of virtual machines and balances resources across available host resources in a vSphere cluster. Virtual machine resources can be guaranteed or limited. Rules can be applied to keep virtual machines together on the same host, or to ensure that virtual machines run on separate hosts.

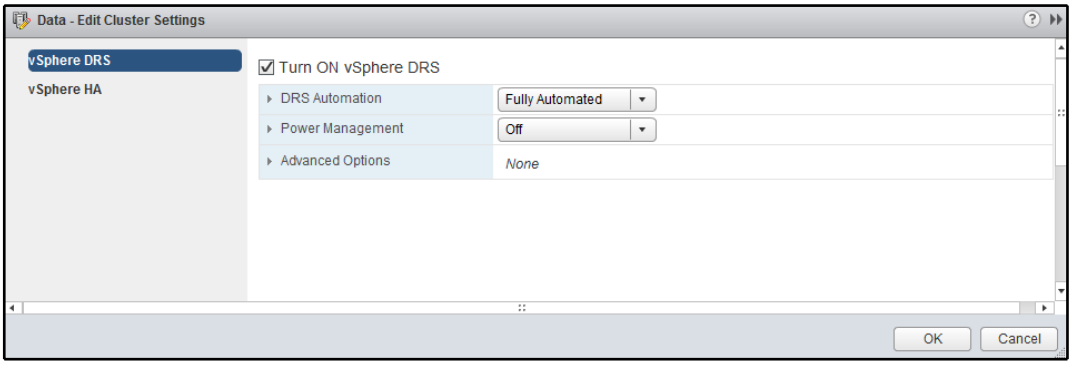
How to do it…

Refer the following steps to implement and configure DRS:

1. Edit the settings of the vSphere cluster to enable vSphere DRS
2. Select a value for the **DRS Automation Level** that should be applied to the DRS-enabled cluster
3. Select a value for the **Migration Threshold** that should be applied to the DRS-enabled cluster

How it works…

vSphere DRS can be enabled when creating a new vSphere cluster, or by editing the settings of an existing cluster:



Enabling vSphere DRS

When DRS is enabled, the **DRS Automation Level** and **Migration Threshold** are set to determine how DRS will place and migrate virtual machines between hosts in the cluster to balance the resources across all hosts in the cluster.

If the **Automation Level** is set to **Manual**, vCenter will make suggestions for initial virtual machine placement and virtual machine migrations. When a virtual machine is powered on, DRS makes a suggestion for the initial placement of the virtual machine based on the balance of cluster resources, but this must be acknowledged by (or can be changed by) the administrator. Migrations will not be performed unless they are acknowledged by an administrator.

Setting the **Automation Level** to **Partially Automated** will make vCenter automatically select a cluster host to place the virtual machine at power-on, but it will only make recommendations for virtual machine migrations. Migrations are not performed unless they are acknowledged by an administrator.

When the **Automation Level** is set to **Fully Automated**, it allows vCenter to automatically determine the initial placement of virtual machines. This setting also causes vCenter to automatically migrate virtual machines between the hosts in the cluster, in order to balance resource usage across all cluster hosts. When the **Automation Level** is set to **Fully Automated**, virtual machines will also automatically be migrated to other hosts in the cluster when a host is placed in the maintenance mode.

The default DRS migration threshold will typically provide the best balance for most clusters. If the cluster resources are not balanced or if too many DRS migrations are being invoked, the migration threshold can be adjusted to be either more conservative or more aggressive.

The **Migration Threshold** determines how the cluster will be balanced when the **Automation Level** is set to **Fully Automated,** or how DRS recommendations will be generated when the **Automation Level** is set to **Manual** or **Partially Automated**.

A conservative migration threshold setting will only cause virtual machines to migrate if the migration will result in a significant improvement in the balance of resources. Setting the migration threshold to be more aggressive will cause the virtual machines to migrate if any benefit can be realized from the migration. Setting the migration threshold to be too aggressive can result in unnecessary virtual machine migrations, or virtual machines constantly migrating in an attempt to aggressively balance the resources.

Ensuring cluster vMotion compatibility

vMotion allows running virtual machines to be migrated between vSphere hosts. To facilitate live vMotion, the processors between hosts must contain the same CPU features and present the same instruction sets. **Enhanced vMotion Compatibility** (**EVC**) masks compatibility issues between the hosts in a cluster.

Enabling EVC on a cluster ensures that hosts that are added to the cluster in the future will not have vMotion compatibility issues.

Processors must be from the same manufacturer; EVC does not provide vMotion compatibility between Intel and AMD processors. EVC is not required to support HA across different processor types and only supports live vMotion between hosts.

How to do it…

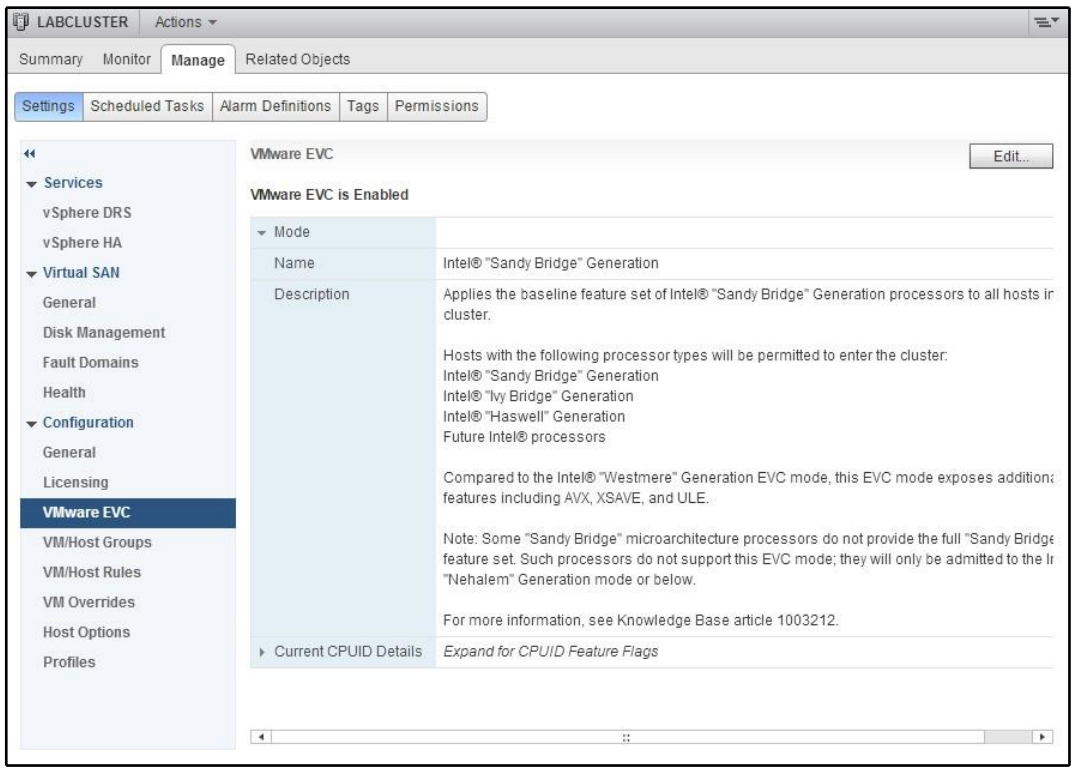
Refer the following steps to enable EVC:

1. Edit the settings of the vSphere cluster
2. Change the value of **EVC Mode** to **Enable EVC** and select an EVC mode baseline

How it works…

The EVC mode is enabled on the cluster when the cluster is created, or by editing the properties of the cluster. The EVC baseline is selected based on the processor manufacturer (EVC for AMD hosts or EVC for Intel hosts). The selected baseline compatibility is validated against all hosts in the cluster.

The following screenshot shows the EVC mode enabled for Intel hosts and the mode set to **Intel "Sandy Bridge" Generation**:



Example EVC configuration

The EVC baseline configuration and the processor supported for each EVC baseline can be found in the *VMware Knowledge Base* at `http://kb.vmware.com/kb/1003212`.

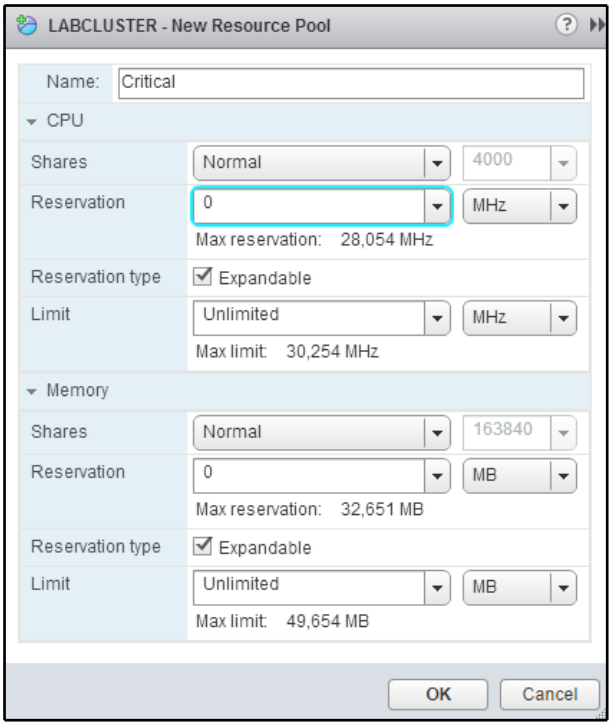
Using resource pools

Resource pools are logical abstractions of resources that can be grouped into hierarchies to reserve or limit CPU and memory resources to virtual machines and subordinate resource pools. Shares, limits, and reservations can be applied to a pool, and can be expanded from child pools to parent pools.

How to do it…

Refer the following steps to configure resource pools:

1. Understand how resource pool shares, reservations, and limits are applied.
2. Create and configure resource pools to reserve or limit resources to virtual machines. The following screenshot shows how a resource pool is created and configured with **Shares**, **Reservations**, and a **Limit** for **CPU** and **Memory** resources:



Example resource pool configuration

How it works…

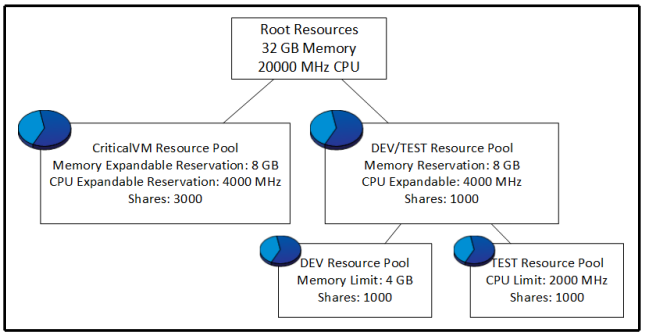
Resource pools are used to define how CPU and memory resources are shared between virtual machines during times of contention, to guarantee CPU and memory resources to a group of virtual machines, and to limit the amount of resources available to virtual machines.

When you are creating a resource pool, the following resource allocations can be applied:

- **Share**: This is used during the time of CPU or memory contention to determine how virtual machines will be scheduled against available CPU and memory resources. Access to the resources is relative to the number of shares allocated. Each virtual machine in a resource pool receives a percentage of the shares that are available to the pool.
- **Reservation**: The CPU and memory resources that are guaranteed to the resource pool. Reservations can be set to expandable, which means that if there are not enough resources in the pool to meet the reservation, it can expand into the parent. If resources are not available to meet the reservation, virtual machines cannot be powered on.
- **Limit**: The upper limit of CPU and memory resources that are available to the resource pool. If a limit is configured, the resource pool will not exceed this limit, even when additional resources are available.

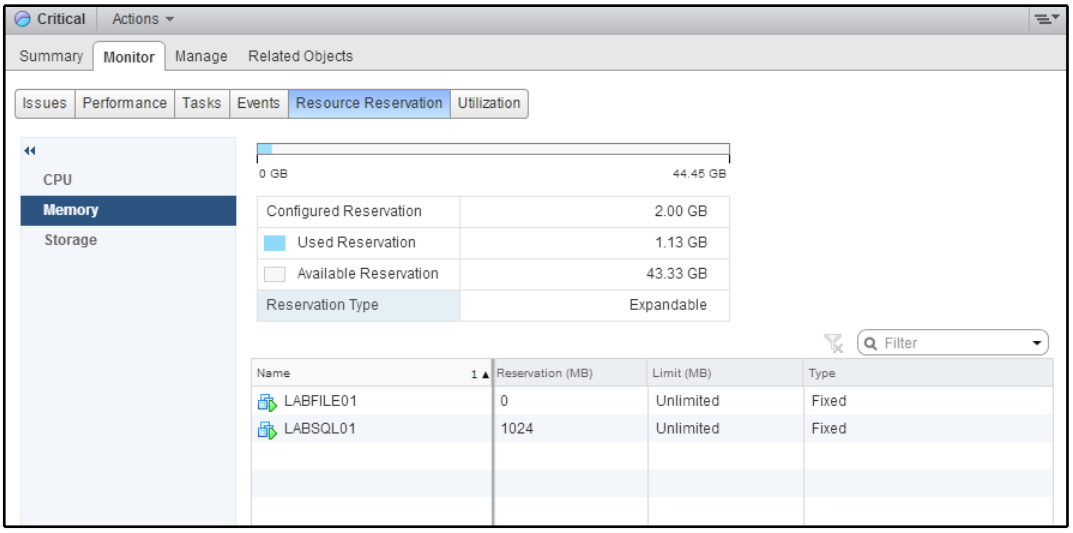
Resource pools can be configured in a hierarchy of parents and children. This can be helpful in delegating shares, reservations, and limits to multiple applications or departments.

The following diagram provides a logical example of how resource pools can be used to allocate available resources between **CriticalVM** workloads and **DEV/TEST** environments:



Resource pool shares example

The resources that are available and consumed by virtual machines in a resource pool can be viewed in vSphere Web Client, as shown in the following screenshot:



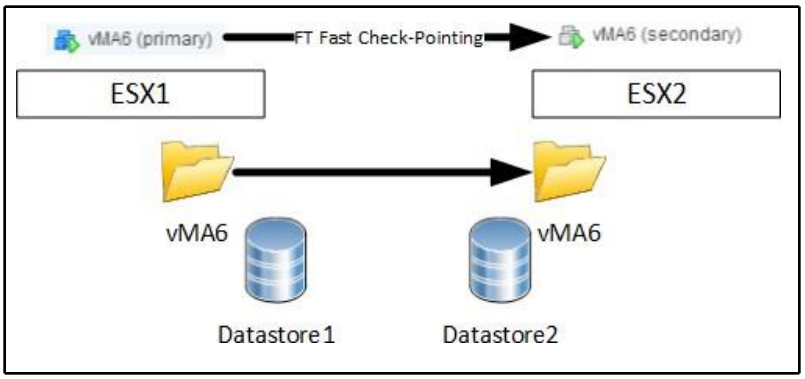
Resource pool monitoring

Notice that the **Configured Reservation** of the pool is set to **2.00 GB**, but the **Available Reservation** is **43.33 GB**. Since the **Reservation Type** is set to **Expandable**, the memory resources in the parent resource pool, in this case, the cluster, are available to this resource pool.

Resource pools add complexity to a design and should only be used if necessary. Do not use resource pools for the organization of virtual machines.

Providing Fault Tolerance protection

vSphere **Fault Tolerance** (**FT**) provides protection from a host or storage hardware failures for critical virtual machines by enabling a secondary running copy of the virtual machine, running on a separate host and stored in a different datastore. The secondary virtual machine is identical to the primary protected virtual machine, and failover is instant and transparent:



Fault Tolerance example

vSphere FT Fast Checkpointing keeps the primary and secondary virtual machines in sync to allow the secondary virtual machine to instantly take over, should the primary virtual machine be impacted by a host or storage failure.

How to do it…

To configure Fault Tolerance, refer the following steps:

1. Identify the use cases for vSphere FT
2. Identify the requirements for enabling vSphere FT
3. Enable vSphere FT for a virtual machine
4. Test vSphere FT for an enabled virtual machine

How it works…

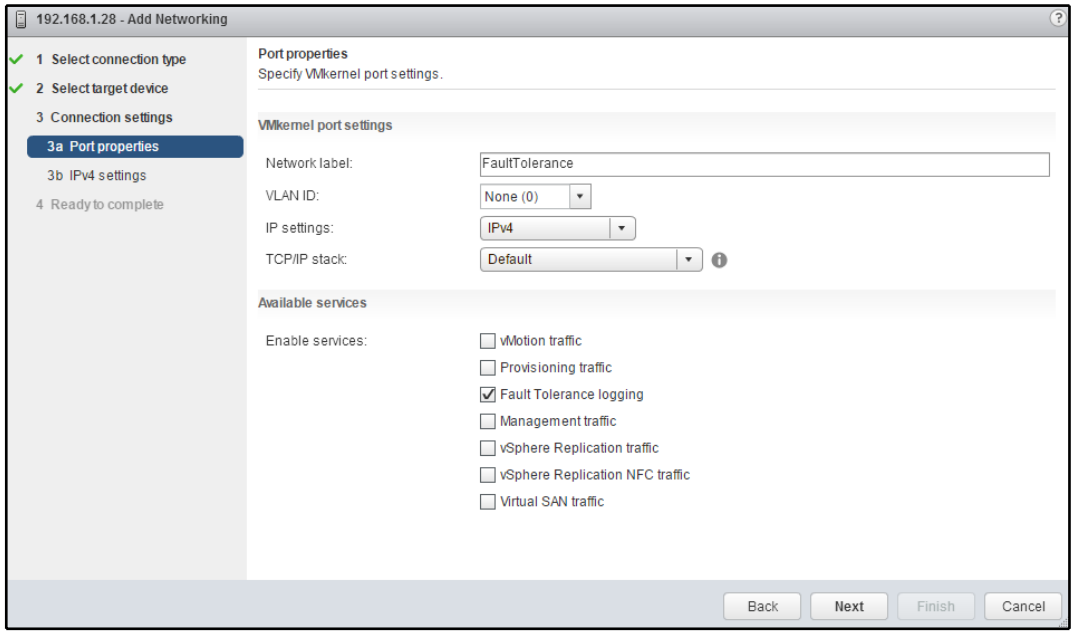
FT protects critical virtual machines against host and storage hardware failures. Prior to vSphere 6, FT only supported a single vCPU virtual machine. Support for **vSMP (Symmetric Multi-Processing)** now provides more use cases for utilizing FT to protect virtual machines, including the following:

- Protecting critical virtual machines with up to 4 vCPUs and 64 GB memory.
- Reducing the complexity of other clustering services.
- Protecting applications sensitive to the loss of TCP connections. FT failover maintains TCP connections between clients and the protected virtual machine.

The requirements for enabling FT protection for a virtual machine are as follows:

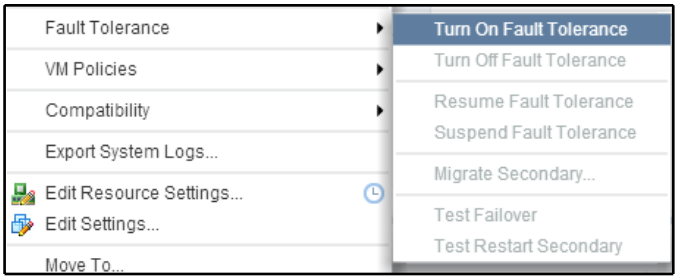
- 10 GbE network connectivity between hosts when protecting multi-vCPU virtual machines.
- vSphere 6 FT supports virtual disks provisioned as thin, thick, or eager-zeroed thick.
- Up to four FT protected (primary or secondary) virtual machines, with up to 8 vCPUs total per host. For example, across two hosts, four virtual machines, with two vCPU each, can be protected.
- All virtual machine allocated memory will be reserved for both the primary and secondary virtual machines when FT is enabled.
- NFS v3 and block storage are supported. VSAN, VVOLS, and NFS v4.1 datastores are not supported for primary or secondary FT protected virtual machines.

FT requires 10 GbE to protect virtual machines with multiple vCPUs. If it is protecting a single vCPU virtual machine, 1 GbE can be used. The Fault Tolerance logging service must be enabled on a VMkernel port, as shown in the following screenshot:



Configuring VMkernel port for Fault Tolerance networking

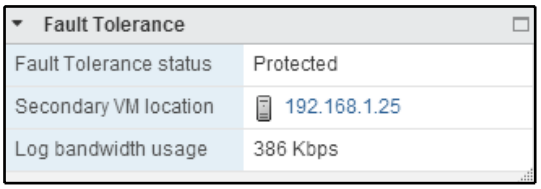
To enable FT for a virtual machine, select the virtual machine in the inventory, right-click it, and, from the **Fault Tolerance** menu, **Turn On Fault Tolerance**, as follows:



Turning on Fault Tolerance

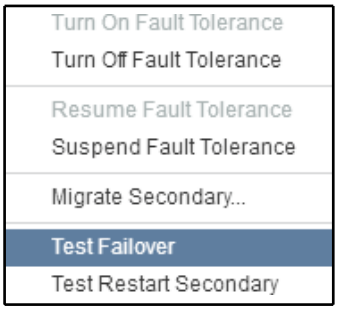
The **Turn On Fault Tolerance** wizard will prompt for a host to run the initial secondary virtual machine and the datastore to store the secondary virtual machine configuration file, VMDK files, and tie breaker file. These should be stored in a separate datastore from the primary virtual machine, but can be stored together on the same datastore; this will not provide protection against a storage outage.

When FT is enabled, a secondary virtual machine is created on a different host from the primary virtual machine and the virtual disks are copied to the selected datastore. Once enabled, the vSphere Web Client displays the **Fault Tolerance status**, **Secondary VM location**, and the **Log bandwidth usage** for the FT protected virtual machine on the virtual machine's **Summary** page, as shown in the following screenshot:



Viewing Fault Tolerance summary

Once FT has been enabled on a virtual machine, it can be tested by selecting **Test Failover** from the **Fault Tolerance** menu of the virtual machine, as follows:



Test Failover option for Fault Tolerance

When testing failover, the secondary virtual machine becomes the primary virtual machine, and a new secondary virtual machine is created.

Leveraging host flash

Virtual Flash enables the use of **Solid State Disks** (**SSD**) or PCIe-based flash storage in ESXi hosts to accelerate the performance of virtual machines by providing read caching and host swapping.

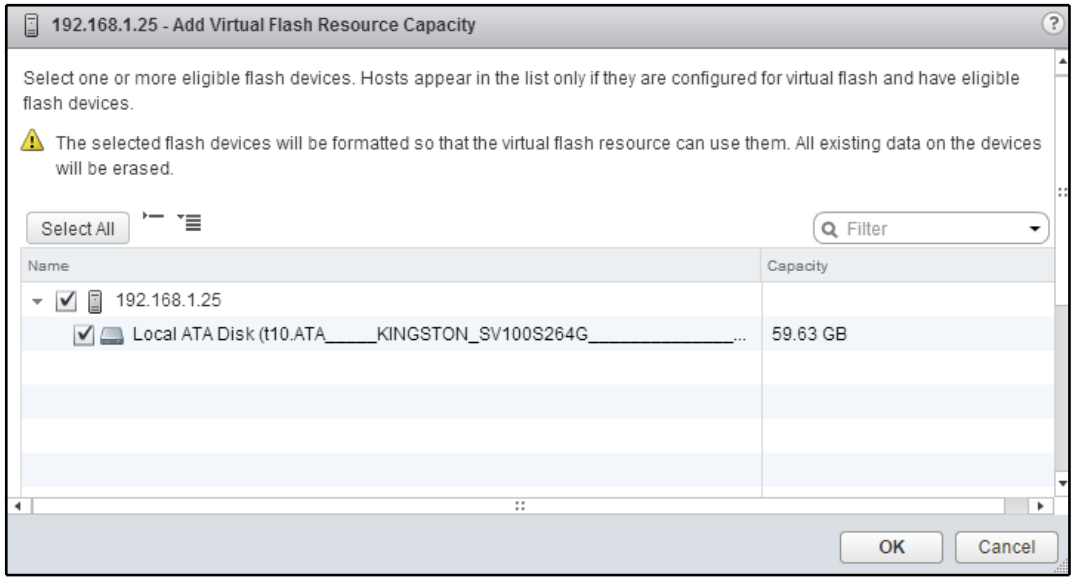
How to do it…

Refer the following steps to leverage host flash:

1. Configure local SSDs or flash devices as a Virtual Flash Resource
2. Configure **vSphere Flash Read Cache** (**vFRC**) for virtual machine disks
3. Allocate Virtual Flash capacity for the host swap cache

How it works…

Configuring local SSDs for use as a Virtual Flash Resource is done from the ESXi host settings' **Virtual Flash Resource Management** menu. Adding capacity will display the flash devices that are eligible to be used as Virtual Flash Resources, as shown in the following screenshot:

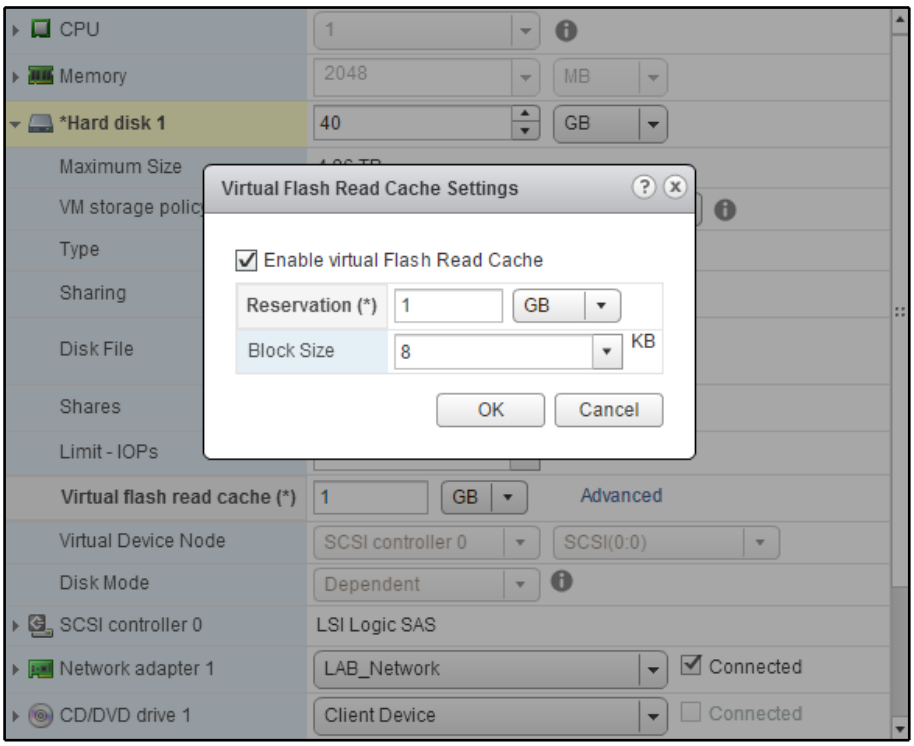


Adding a virtual flash resource

Flash devices that are selected as Virtual Flash cache are formatted with a **Virtual Flash File System** (**VFFS**), and the capacity can only be used for Virtual Flash Resource.

Once flash capacity has been added, the virtual machines are configured to consume the Virtual Flash Resource as vFRC. vFRC is configured per virtual machine disk.

Configuring the flash read cache for a virtual machine is done by using **Edit Settings**. Select the virtual machine hard disk to configure for **Virtual Flash Read Cache**, **Enable virtual Flash Read Cache**, and allocate the amount of cache to reserve and the **Block Size** for the virtual machine, as shown in the following screenshot:

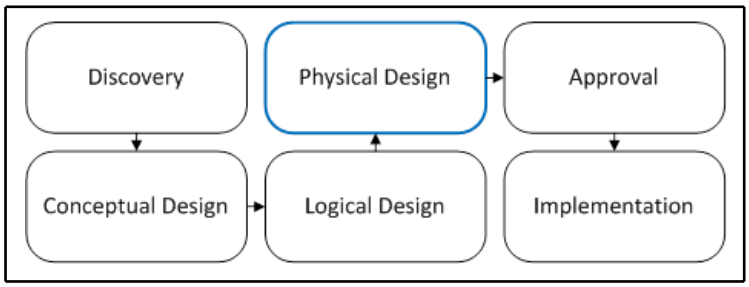


Enabling virtual Flash Read Cache

A portion of Virtual Flash Resource capacity can be reserved for the host swap cache. This cache can be shared by all virtual machines running on the host and provides low-latency caching for virtual machine swap files. Using Virtual Flash Resources for host caching will reduce the performance impact on the virtual machines, should VMkernel swapping occur.

8
vSphere Physical Design

The vSphere physical design process (as shown in the following diagram) includes choosing and configuring the physical hardware that is required to support storage, network, and compute requirements:



Physical design in the vSphere design workflow

During the physical design process, the hardware and configuration choices should map to the logical design and satisfy the functional and nonfunctional design requirements.

A design architect should answer the following questions about each design decision:

- Does the design meet the requirements of the logical design?
- Does the design satisfy the functional and nonfunctional requirements?
- Is the selected hardware supported?

There will often be more than one physical solution that will meet the design requirements. The job of the architect is to choose hardware to provide the resources that are required while meeting the design requirements and constraints.

This chapter will contain recipes for using VMware's Hardware Compatibility List, the physical design of storage, network and compute resources, creating a custom ESXi image, and the best practices for BIOS settings on a server running ESXi. This chapter will also provide an overview of methods for upgrading existing ESXi hosts.

In this chapter, we will cover the following recipes:

- Using the VMware **Hardware Compatibility List** (**HCL**)
- Understanding the physical storage design
- Understanding the physical network design
- Creating the physical compute design
- Creating a custom ESXi image
- The best practices for ESXi host BIOS settings
- Upgrading an ESXi host

Using the VMware Hardware Compatibility List

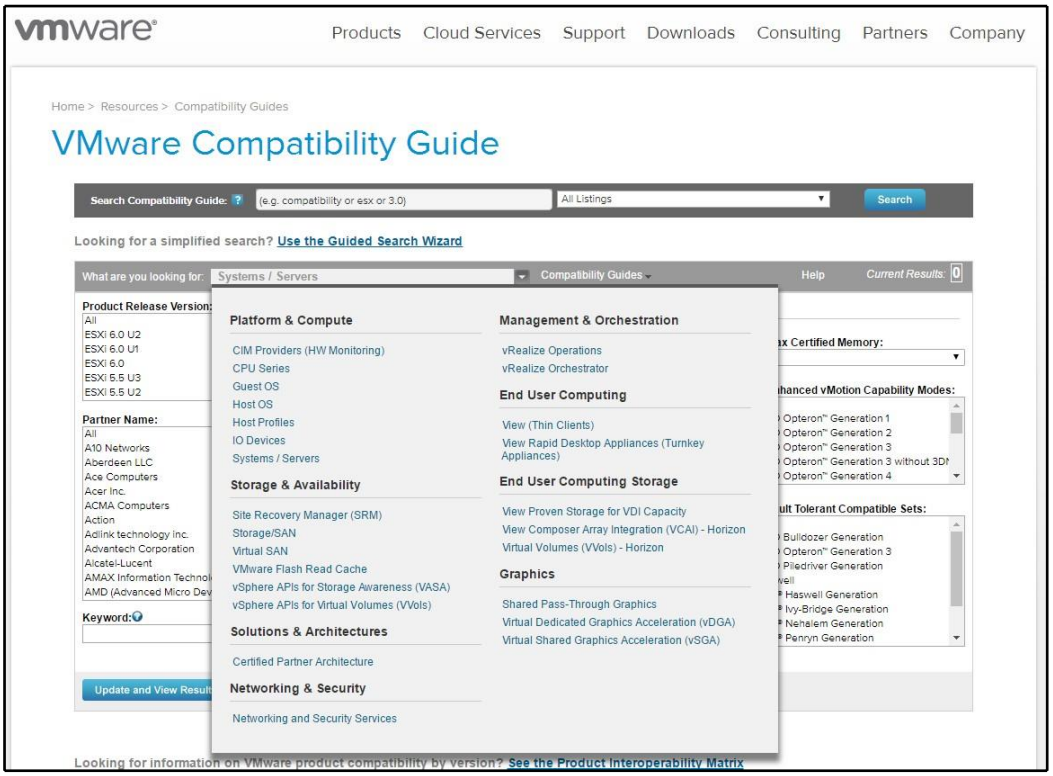
VMware's Hardware Compatibility List is a database of all of the tested and supported physical hardware. The physical hardware that's chosen to support the created design must be checked against the HCL to ensure that it will be supported. This includes storage devices, I/O devices, and servers. It is important to ensure not only that the hardware vendor and model are supported, but also that the firmware version of the hardware is supported.

Verifying support against the HCL is important not only for new designs, but also when upgrading a design from one version to another in vSphere. Legacy hardware is often removed from the HCL when new versions of vSphere are released.

How to do it…

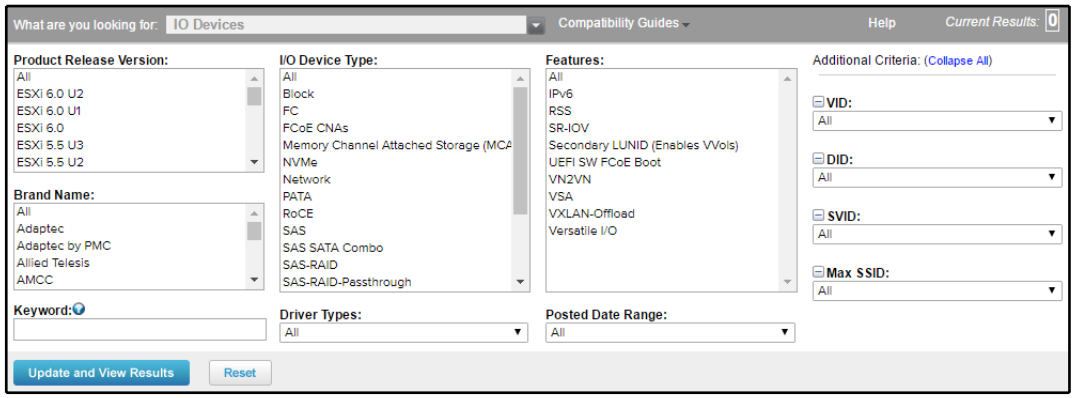
To verify whether a certain hardware device is supported in the current version of vSphere, perform the following process:

1. Visit `http://www.vmware.com/go/hcl/`.
2. Select the type or category of device to determine its compatibility by selecting it using the **What are you looking for** drop-down menu. For example, if the compatibility of a **Network Interface Card** (**NIC**) is being determined, select **IO Devices**, as shown in the following screenshot:



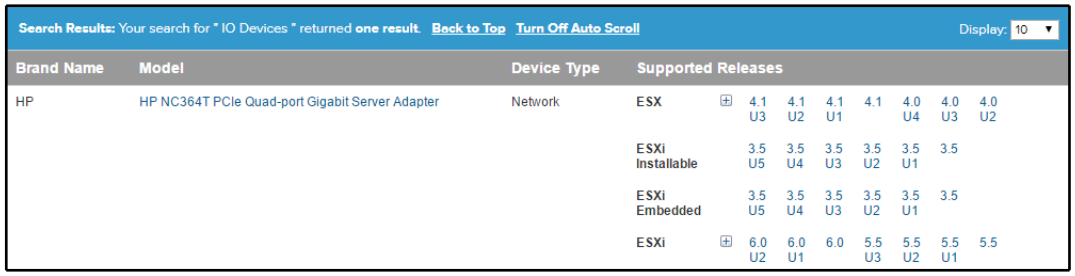
Screenshot of the VMware Hardware Compatibility List webpage

3. Select values for the **Product Release Version**, **Brand Name**, and **IO Device Type**, as shown in the following screenshot:



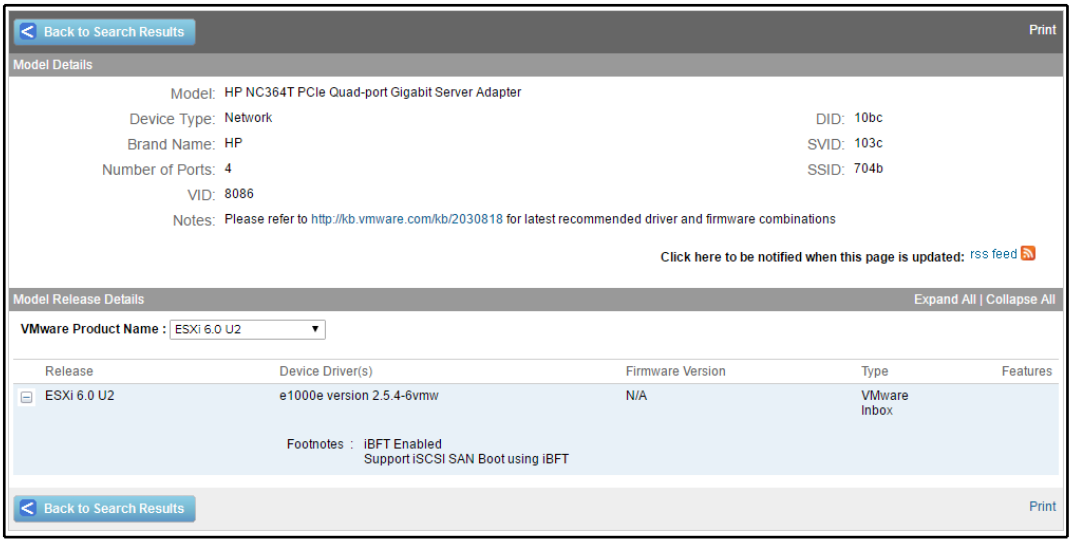
IO device options in the VMware HCL

4. Enter a value for **Keyword**, such as the model number. In this example, the search will be for `NC364T`, which is an HP quad port 1 GB server adapter, as shown in the following screenshot:



Example output of IO devices in the VMware HCL

5. Clicking on the device model will display details about the device, the firmware versions that are supported, and the device driver that's used by ESXi, as shown in the following screenshot:



Example output after choosing a device model in the VMware HCL

How it works…

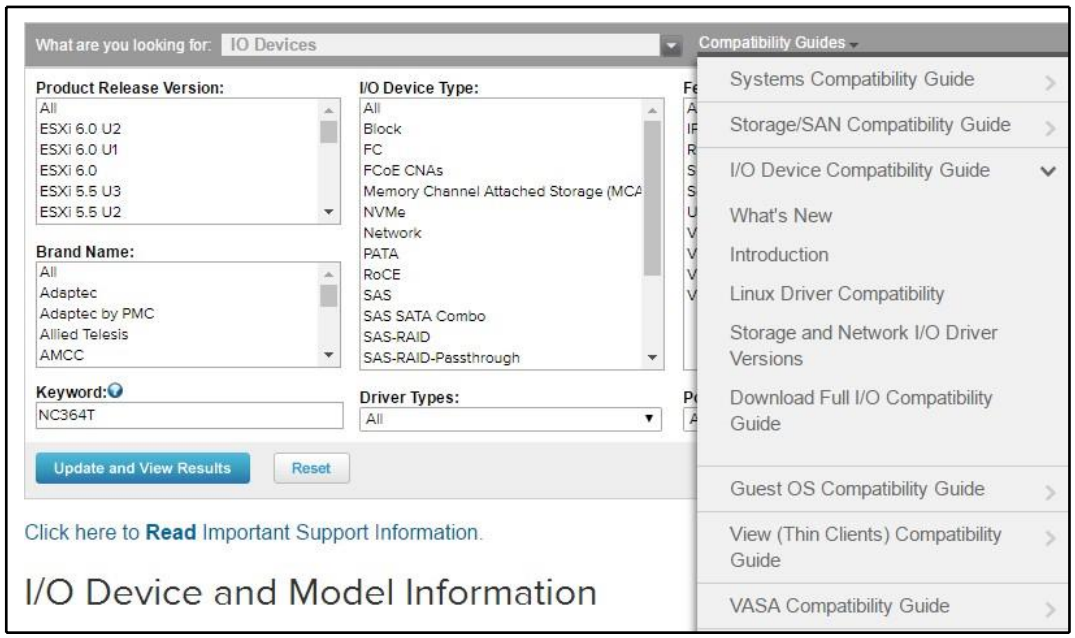
The VMware HCL provides an easy-to-search online database of hardware that has been tested and is supported in a vSphere environment.

Hardware that is not listed on VMware's Hardware Compatibility List may still work with vSphere. However, if the hardware is not listed on the HCL, it may cause issues in regards to obtaining support from VMware in the event of issues with the environment. Only hardware found on the HCL should be used in vSphere production environments.

Selecting the hardware type, VMware product and version, hardware vendor, device type, and supported features allows a design architect to quickly view and select supported hardware. Details about the supported firmware or BIOS versions, the availability of native drivers, and the requirements for third-party drivers can also be viewed quickly for hardware information on the HCL.

There's more…

VMware also provides compatibility guides. The compatibility guides can be accessed through a menu on the HCL page, located at `http://www.vmware.com/go/hcl`. These guides provide details about the features that are supported by a specific piece of supported hardware, and can be seen in the following screenshot:



Listing the compatibility guides directly from the HCL webpage

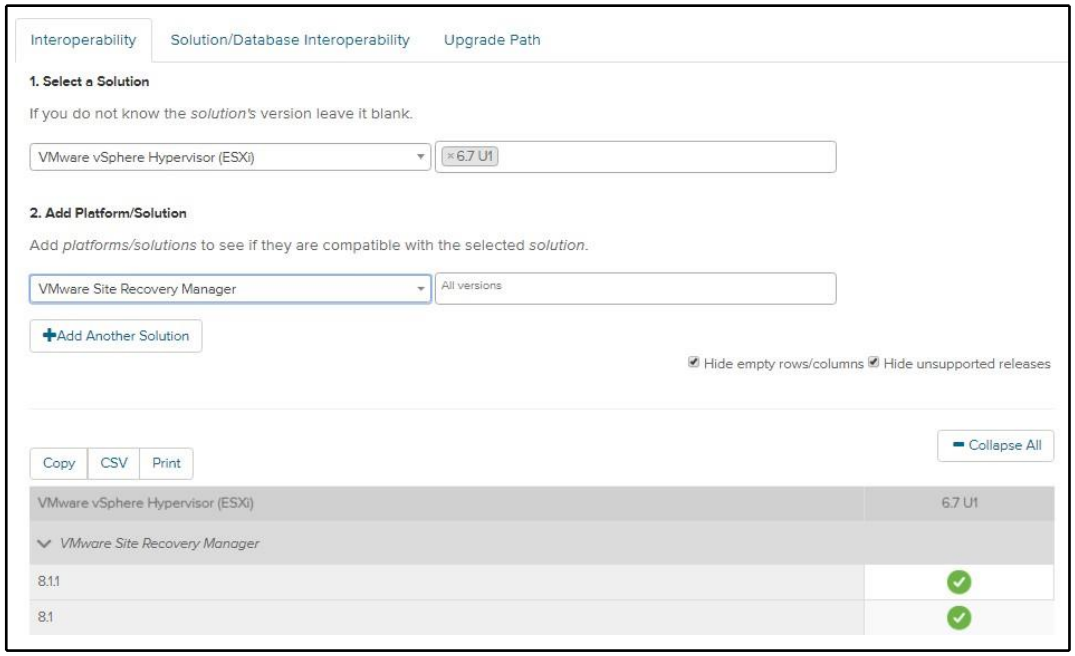
The following screenshot is an excerpt from the **Storage/SAN Compatibility Guide**, showing the support details of the EMC VNX series arrays:

EMC	VNX5100	FC	VMW_SATP_ALUA_CX / VMW_PSP_FIXED *13, 34, 74 VMW_SATP_ALUA_CX *13, 34, 74, 80, 182, 24, 148, 158, 196, 76, 168, 187, 260, 171, 201, 202, 270, 243, 172, 271, 272, 273, 274, 275	DGC
EMC	VNX5200	FC	VMW_SATP_ALUA_CX *13, 34, 74, 80, 182, 24, 148, 158, 196, 76, 168, 187, 260, 171, 201, 202, 270, 243, 172, 271, 272, 273, 274, 275	RAID5
EMC	VNX5300	FC	VMW_SATP_ALUA_CX / VMW_PSP_FIXED *13, 34, 74 VMW_SATP_ALUA_CX *13, 34, 74, 80, 182, 24, 148, 158, 196, 76, 168, 187, 260, 171, 201, 202, 270, 243, 172, 271, 272, 273, 274, 275	DGC
EMC	VNX5400	FC	VMW_SATP_ALUA_CX *13, 34, 74, 80, 182, 24, 148, 158, 196, 76, 168, 187, 260, 171, 201, 202, 270, 243, 172, 271, 272, 273, 274, 275	RAID5
EMC	VNX5500	FC	VMW_SATP_ALUA_CX *13, 34, 74, 80, 182, 24, 148, 158, 196, 76, 168, 187, 260, 171, 201, 202, 270, 243, 172, 271, 272, 273, 274, 275 VMW_SATP_ALUA_CX / VMW_PSP_FIXED *13, 34, 74, 24	DGC

Example EMC VNX support output from the HCL

Another important guide to reference is VMware's *Product Interoperability Matrix*, located at `http://partnerweb.vmware.com/comp_guide2/sim/interop_matrix.php`. This guide provides interoperability information about vSphere products, databases, and host operating systems. If multiple VMware products will be used in the storage design, their interoperability must be checked against the matrix to determine which versions are compatible with which products.

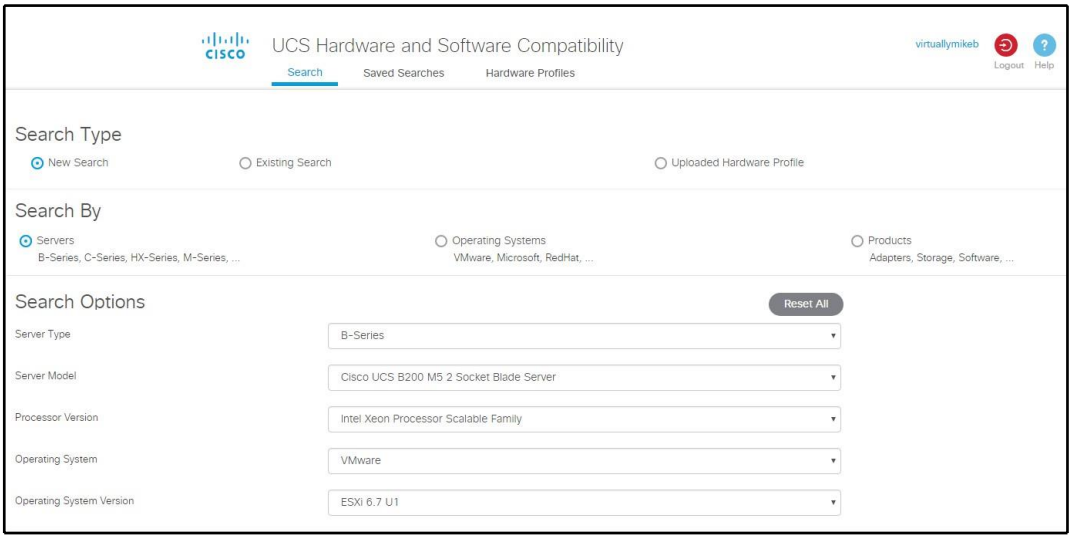
For example, if the VMware **Site Recovery Manager** (**SRM**) is going to be used in a vSphere 6.7 design, the interoperability matrix can be checked to determine which versions of SRM are compatible with that version of ESXi 6.7:



Example ESXi and SRM compatibility output from the VMware Product Interoperability Matrix

The VMware compatibility guides are also available from the **Product Interoperability Matrixes** page.

In addition to VMware interoperability pages, hardware vendors usually have their own interoperability pages that they require you to adhere to when running their hardware. For example, a common vSphere design includes VMware vSphere on Cisco UCS blade servers. If you call Cisco for technical support, one of the first things they'll do is verify on their own HCL whether they support your configuration. If not, they'll likely ask you to update your environment to a supported state before they'll put too much time into troubleshooting. The following screenshot shows Cisco's HCL:



Cisco's UCS Hardware Compatibility List

Understanding the physical storage design

Storage is the foundation of any vSphere design. Properly designed storage is key for vSphere features like **High Availability** (**HA**), **Distributed Resource Scheduling** (**DRS**), and **Fault Tolerance** (**FT**), to operate.

How to do it…

Performance, capacity, availability, and recoverability are all factors that must be taken into account when determining the hardware and the configuration of the physical storage. The physical storage design requires that you follow these steps:

1. Select a storage hardware that satisfies the logical storage design. This includes the storage array, storage host bus adapters, and any switching, fiber channel, or Ethernet that may be required to support storage connectivity.
2. Verify the compatibility of each storage hardware component by using the VMware HCL.

3. Design the storage configuration to satisfy the design factors related to availability, recoverability, performance, and capacity.

How it works…

The physical storage design must meet the capacity and performance requirements that are defined by the logical storage design, and these requirements must be mapped back to the design factors.

The logical storage design identifies the capacity, IOPS, and throughput that are required to support the vSphere design. The design factors identify the functional requirements, such as availability and recoverability, and any constraints that may be placed on the physical design, such as using an array from a specific vendor or using a specific storage protocol.

The logical storage design specifications are as follows:

- **Storage capacity**: 16 TB
- **Storage IOPS**: 6,250
- **I/O profile**: 8 k
- **I/O size**: 90% Read/10% Write
- **Total storage throughput**: 55 MB/s
- **Number of virtual machines per datastore**: 20
- **Datastore size**: 2.5 TB

The factors that influence the physical storage design are as follows:

- Shared or local storage
- Block storage or file storage
- Array specifications, such as active/active or active/passive, the number of storage processors, and the cache
- Storage protocol that uses fiber channel, iSCSI, NFS, or **Fiber Channel over Ethernet** (**FCoE**), as well as the type and number of disks and the RAID configuration
- Support for VMware integration: **vStorage APIs for Array Integration** (**VAAI**) and **vSphere APIs for Storage Awareness** (**VASA**)
- Support for advanced storage technologies: deduplication, tiering, and flash-based cache

- The **Recovery Point Objective** (**RPO**), which is the amount of data that will not be lost in the event of a disaster, and the **Recovery Time Objective** (**RTO**), which is the amount of time it takes to recover the system and data in the event of a disaster

The chief considerations when choosing a storage platform are IOPS, throughput, and support for features such as VAAI, VASA, SRM, and accelerated backups. The physical storage design should focus on both performance and capacity. The physical storage design must be able to meet the performance requirements of the design.

Meeting the design capacity requirements is typically easy to accomplish, but ensuring that the storage will meet the performance requirements takes a bit more work. The I/O profile of the workloads, the number of IOPS required, the types of disks used, and the RAID level that is selected all have an impact on the storage performance. It is good practice to first design the storage to meet the performance requirements, and then design it to meet the capacity requirements.

Understanding the physical network design

Network connectivity must be provided for both virtual machine network connectivity and VMkernel connectivity. Physical switches, uplinks, virtual switches, and virtual port groups are all components of the physical network design.

How to do it…

Performance, capacity, availability, and recoverability are all factors that must be taken into account when determining the hardware and the configuration of the physical network. The following steps are necessary to successfully complete the physical network design:

1. Select the network hardware that satisfies the logical network design, including physical network switches and network interface cards.
2. Verify whether the network I/O device hardware, such as network interface cards and **Converged Network Adapters** (**CNAs**), are compatible and supported by using the VMware HCL.

3. Design the physical network topology and virtual network configuration to satisfy the design factors related to availability, recoverability, performance, and capacity.

How it works…

The physical network design must satisfy the performance and availability requirements that are defined by the logical network design, which, in turn, must support the design factors. The logical network design identifies the capacity requirements, and the design factors define the availability and recoverability requirements.

Aside from providing virtual machine connectivity, many vSphere features, such as High Availability, vMotion, and Fault Tolerance, have specific virtual and physical network connectivity requirements that must be taken into account when designing the physical network. If IP-connected storage like iSCSI or NFS is used, the physical network connectivity for these must also be included as part of the physical network design.

The logical network design specifications are as follows:

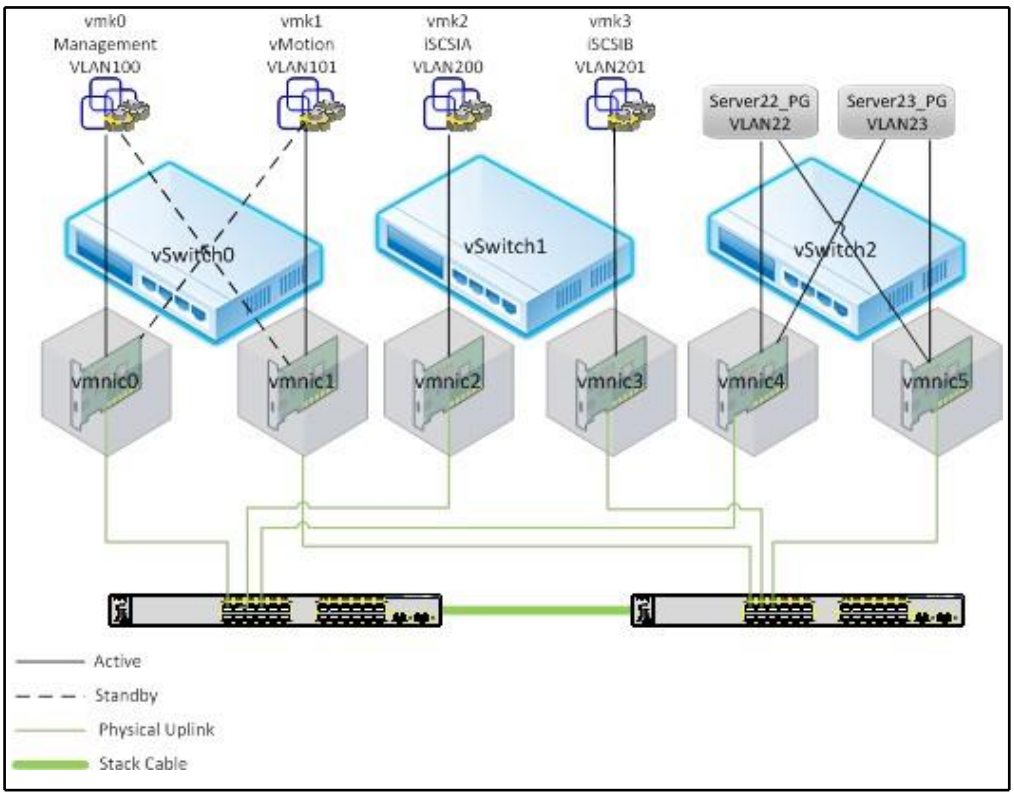
- **Total virtual machine throughput**: 1000 Mbps
- **Virtual machines per host**: 20
- **Virtual machine throughput per host**: 200 Mbps
- **IP storage**: iSCSI
- **Storage throughput**: 55 MB/s
- **vMotion/DRS**: Enabled

The factors that influence the physical network design include the following:

- The number and type of the physical switches
- The topology of the existing physical network
- Using either physically or logically separated (such as VLANs) networks
- The number of physical uplinks per host
- The physical adapter type: 1 GB or 10 GB
- Teaming and link aggregation
- Network bandwidth and throughput
- Failover and failback policies
- The quality of service and traffic shaping

- The type of virtual switches to use: virtual standard switches or virtual distributed switches
- The networks required for VMkernels to support management, vMotion, Fault Tolerance, and IP-connected storage

In the following diagram of an example virtual and physical network design, the VLANs and virtual standard switches have been configured to distinguish traffic types; single points of failure have been minimized by using multiple uplinks, and failover policies have been configured to provide redundancy and performance:



Example physical and virtual network design

Creating the physical compute design

The physical compute design selects the CPU and memory resources to meet the requirements of the design. Aside from the CPU and memory resources, the physical compute design also includes selecting the form factor to support the interface cards that are necessary to support the design.

How to do it…

Like with other parts of the physical design, the performance, capacity, availability, and recoverability are all factors to consider in the physical compute design. The following steps can be performed to create the physical compute design:

1. Select server hardware that satisfies the logical compute design
2. Verify the compatibility of each component of the compute hardware by using the VMware HCL
3. Configure compute resources to satisfy the design factors related to availability, recoverability, performance, and capacity

How it works…

The logical compute design defines the capacity and performance requirements for CPU and memory resources.

The logical compute design specifications are as follows:

- **Total CPU resources**: 167 GHz
- **Total memory resources (25% TPS savings)**: 657 GB
- **Number of virtual machines per host**: 20
- **Number of hosts required (*N*+2)**: 9
- **CPU resources per host**: 23.8 GHz
- **Memory resources per host**: 94 GB

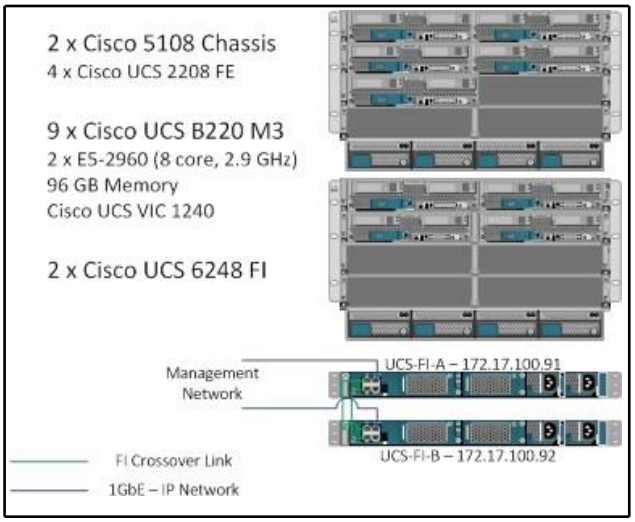
The hardware that's selected for the physical compute design must satisfy the resource requirements of the logical compute design. These resources include the CPU and memory resources. The physical hardware that's selected must also be able to support the network and storage connectivity resources that are defined in the logical network and storage design.

Along with the design requirements and constraints, the factors that influence the physical compute design are as follows:

- The required CPU resources
- The required memory resources
- The vCPU-to-CPU-core ratio

- The processor manufacturer and model
- The number of hosts required: scale up or scale out
- The host form factor: rack or blade
- The number of PCI slots
- The number and type of network uplinks
- The number and type of **Host Bus Adapters** (**HBA**)
- Power, space, and cooling requirements

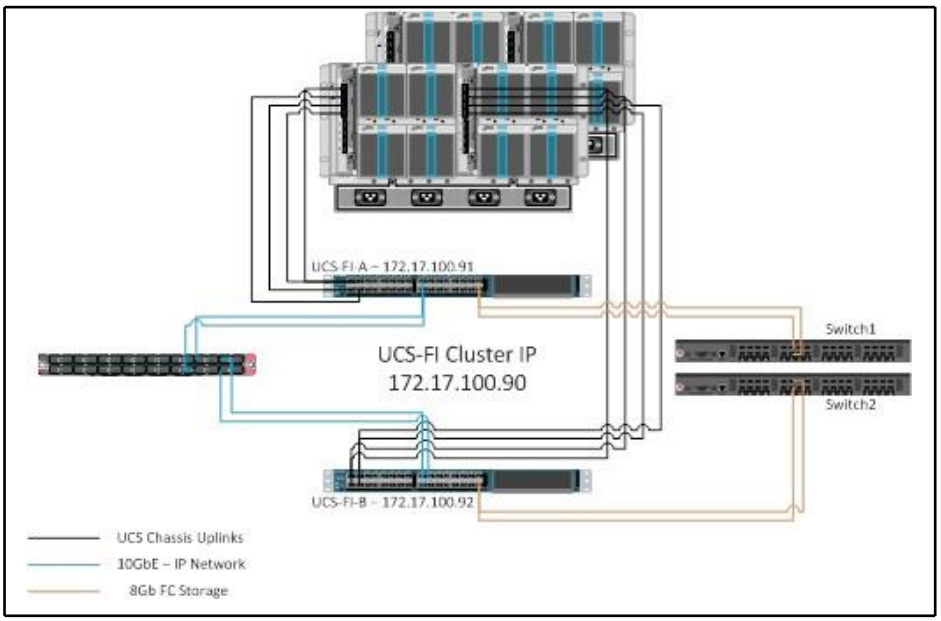
The following diagram is an example of a physical compute design that uses the Cisco UCS blade platform; the blades have been configured to support the logical requirements, and multiple chassis have been chosen to eliminate single points of failure:



Example physical compute design

The following diagram is the rear view of the Cisco UCS blade solution and shows the supporting components, including the connectivity of the chassis to the fabric interconnects.

The diagram also shows connectivity between the fabric interconnects and the network and storage. Multiple links to the chassis, network, and storage not only provide the capacity and performance that are required, but also eliminate single points of failure, as shown in the following diagram:



Example Cisco UCS physical compute solution

Creating a custom ESXi image

The drivers for some supported hardware devices are not included as part of the base ESXi image. These devices require that a driver be installed before the hardware can be used in vSphere.

How to do it…

Third-party drivers are packaged as **vSphere Installation Bundles** (**VIBs**). A VIB file is similar to a ZIP archive, in that it is a single file that includes an archive of the driver files, an XML descriptor file, and a signature file. VIB files have the `.vib` file extension.

The required drivers can be installed after ESXi has been installed using the `esxcli` command:

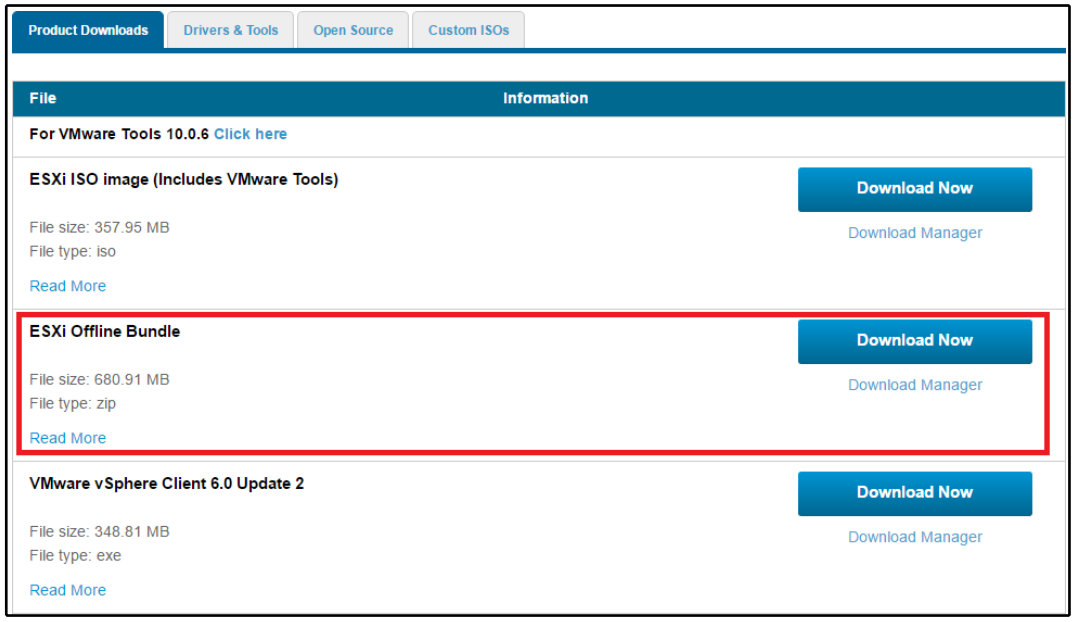
```
esxcli software vib install -v <path to vib package>
```

When installing from a bundle or ZIP file, the following `esxcli` command can be used:

```
esxcli software vib install -d <full path to vib zip bundle>
```

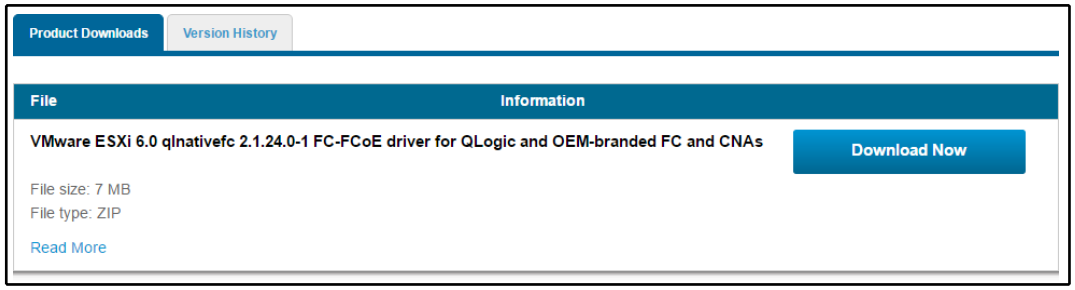
A custom ESXi image can also be created by using the Image Builder tools that are included with PowerCLI. PowerCLI can be downloaded from `https://www.vmware.com/support/developer/PowerCLI/`. Custom ESXi images can be used when deploying hosts using VMware Auto Deploy, or custom images can be exported to an ISO to be used for installation or upgrades. Perform the following steps to create a custom EXSi image:

1. Download the **ESXi Offline Bundle** from the **My VMware** portal. The following screenshot displays the **ESXi Offline Bundle** download link on the **My VMware** portal:



Download the ESXi Offline Bundle

2. Download the required third-party VIB files. This example uses the drivers of QLogic FC-FCoE, which were downloaded from the **My VMware** portal, as shown in the following screenshot:

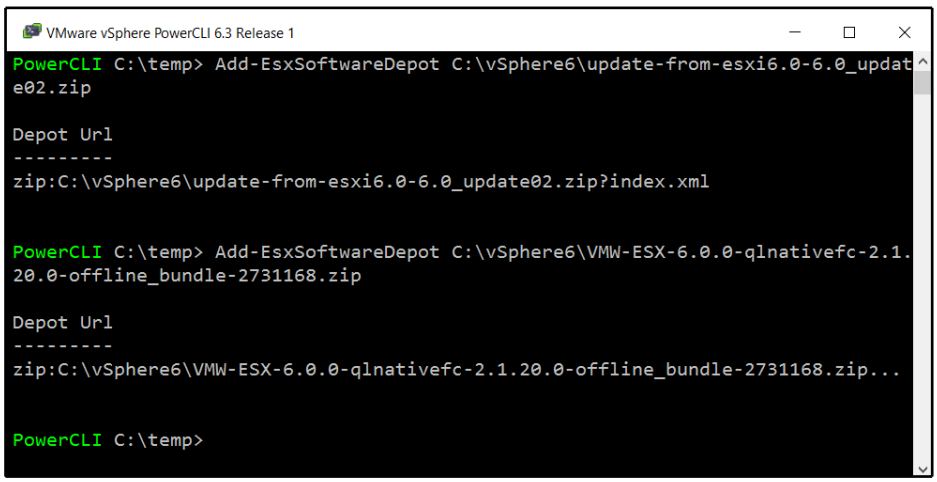


Example download of drivers to install in custom image

3. Use Image Builder PowerCLI to add the **ESXi Offline Bundle** and third-party VIB files as software depots, as follows:

```
Add-EsxSoftwareDepot <pathtoESXiOfflineBundel.zip> Add-
EsxSoftwareDepot <pathto3rdPartyVIB.zip>
```

The following screenshot illustrates adding the **Offline ESXi Bundle** and third-party software bundles by using the `Add-EsxSoftwareDepot` Image Builder PowerCLI command:

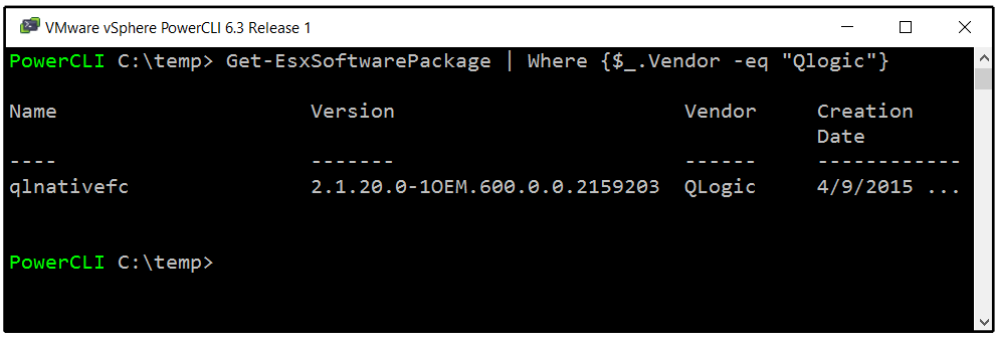


Adding the offline ESXi bundle and third-party software to software depot with PowerCLI

4. List the available software packages to locate the `Qlogic` drivers, and note the package names:

```
Get-EsxSoftwarePackage | where {$_.Vendor -eq "Qlogic"}
```

The following screenshot illustrates the use of the `Get-EsxSoftwarePackage` PowerCLI command to locate the package name of the third-party package that will be added to the new ESXi image:

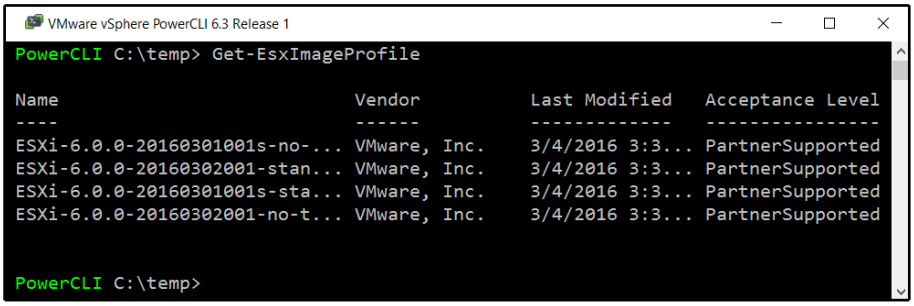


Locating the package name of third-party packages using PowerCLI

5. List the available image profiles by using the following command:

```
Get-EsxImageProfile
```

The following screenshot illustrates the output of the `Get-EsxImageProfile` PowerCLI command that lists the available profiles:

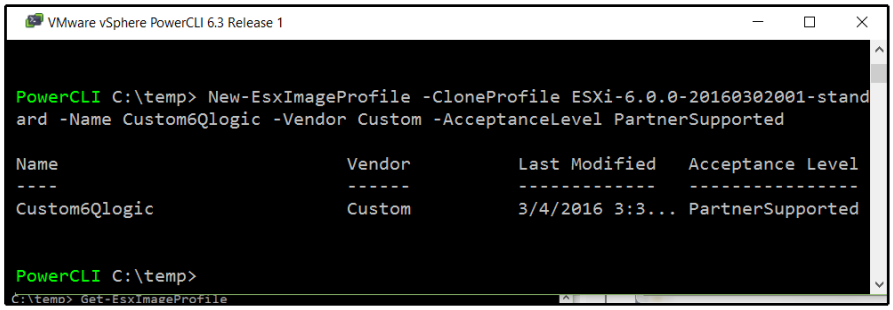


Example output of Get-EsxImageProfile in PowerCLI

6. Create a clone of an image profile to apply customization to. The clone will allow the profile to be manipulated without making changes to the original profile:

```
New-EsxImageProfile -CloneProfile <ProfiletoClone> -Name
<CustomProfileName> -Vendor Custom -AcceptanceLevel
PartnerSupported
```

The following screenshot illustrates the output of the `New-EsxImageProfile` PowerCLI command that creates a clone of an existing profile:

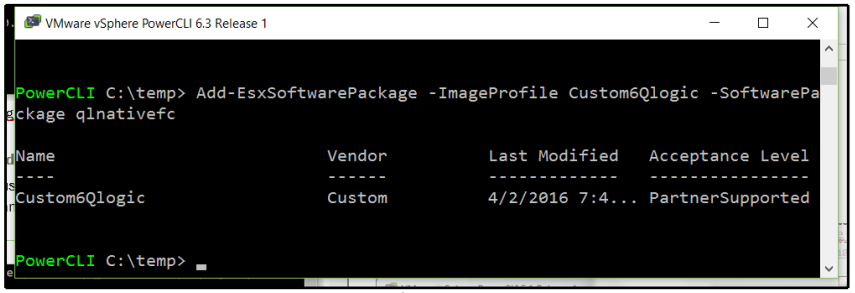


Creating a clone of an existing profile in PowerCLI

7. Add the software packages to the cloned image profile; this step is repeated for each package that needs to be added to the new image profile:

```
Add-EsxSoftwarePackage -ImageProfile <CustomProfileName> -
SoftwarePackage <SoftwarePackagetoAdd>
```

The following screenshot displays the output of the `Add-EsxSoftwarePackage` PowerCLI command when the third-party software package is added to the new ESXi image:

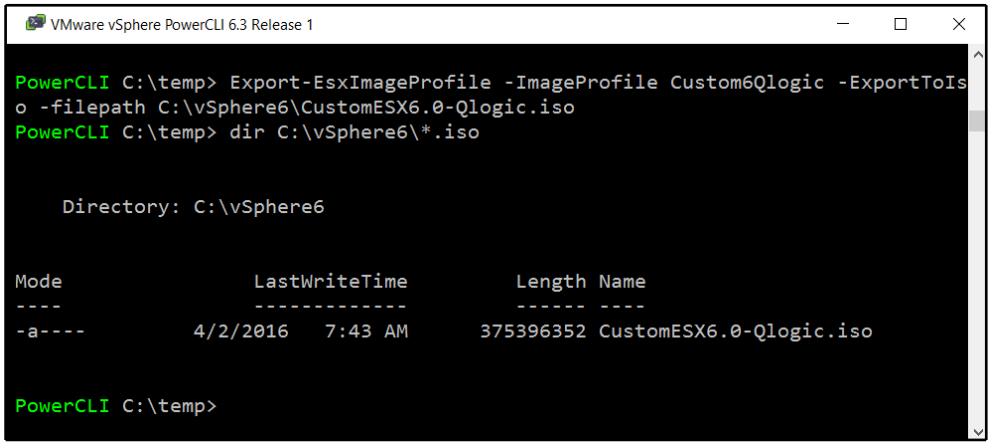


Adding a third-party software package using PowerCLI

8. Create an ISO from the cloned image profile by using the following command; the ISO will include the additional software packages:

```
Export-EsxImageProfile -ImageProfile <CustomProfileName> -
ExportToIso -filepath <Pathtonew.iso>
```

The following screenshot shows the `Export-EsxImageProfile` PowerCLI command. There will be no message output from the command if it completes successfully, but the ESXi image ISO will be created and made available in the provided path:



Exporting the new image in PowerCLI

The new ISO image includes the third-party VIB files, and it can be used to install ESXi.

How it works…

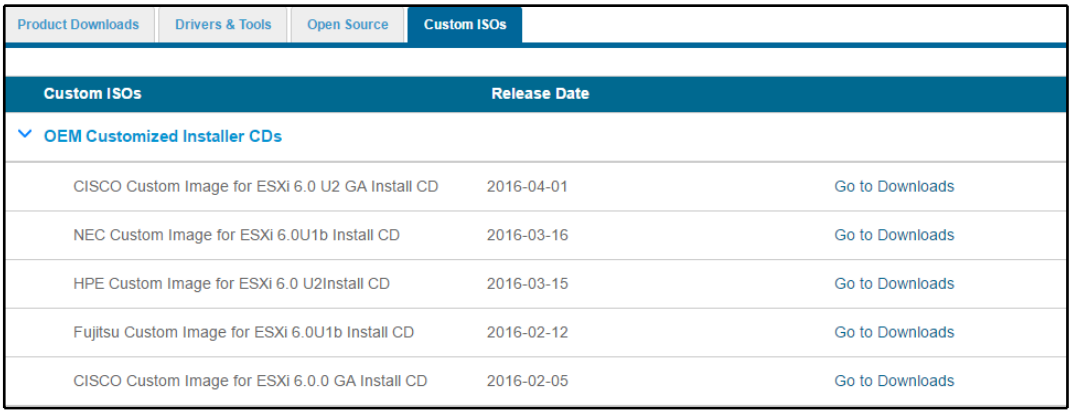
The Image Builder PowerCLI commands are used to create a custom image from the ESXi Offline Bundle and the vendor-provided bundle. To maintain the smallest footprint possible, not all supported hardware drivers are included with the native ESXi installation package.

The ESXi Offline Bundle contains multiple profiles: one that includes VMware tools (the standard profile) and one without VMware tools (the no-tools profile). A clone of the profile is created and the vendor software is added to it. When all of the necessary software has been added to the profile, it is exported to an ISO image that can be used to deploy ESXi hosts.

Custom image profiles that are created using the Image Builder PowerCLI commands are also used when using auto deploy to deploy stateless or stateful hosts. The procedure for creating a custom image profile to be used by Auto Deploy is the same, with the exception of the profile being exported to the custom ISO.

There's more…

Custom ESXi ISOs are also provided by manufacturers. Cisco, HP, Hitachi, Fujitsu, and other hardware manufacturers provide these custom ESXi ISO images, which can be downloaded from the **My VMware** portal using the **Custom ISOs** tab, as illustrated in the following screenshot:

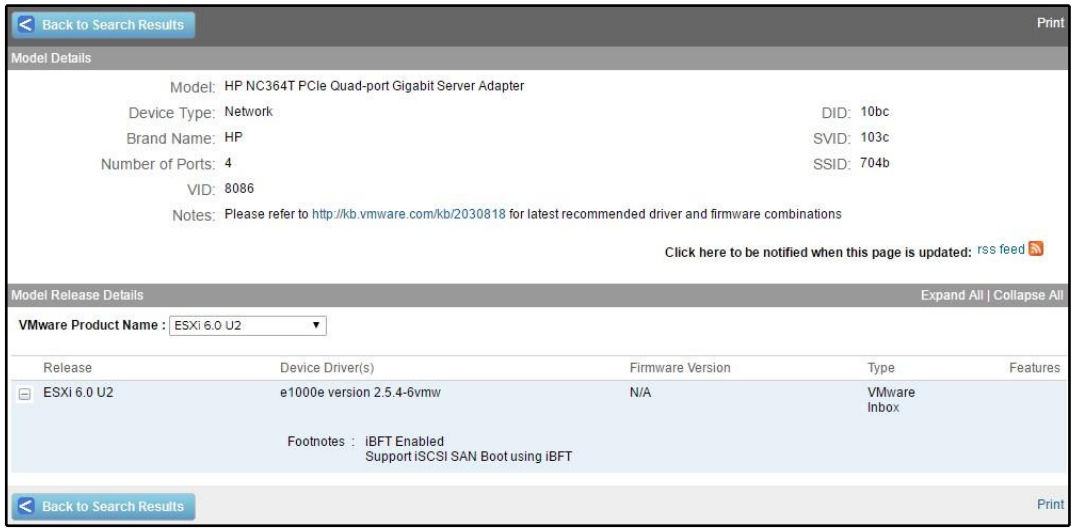


Custom ISOs are available to download

These custom ISOs are preconfigured to include the drivers that are necessary for manufacturer-specific hardware.

The best practices for ESXi host BIOS settings

The BIOS settings will vary, depending on the hardware manufacturer and the BIOS version. Supported BIOS versions should be verified on the VMware HCL for the hardware selected. The following screenshot shows the HCL details of a Dell **PowerEdge R620**, with the supported BIOS versions:



HCL details with supported BIOS versions

If the hardware is supported but the running BIOS version is not supported, the BIOS should be upgraded to a supported version.

How to do it…

The BIOS manufacturer and the BIOS version will determine the BIOS settings available for a particular server. The following settings are provided as a guideline for optimizing the BIOS for an ESXi installation. Ask the hardware vendor for recommendations on settings that are specific to the hardware and BIOS versions:

- Enable Intel VTx or AMD-V
- Enable Intel **Extended Page Tables** (**EPT**) or AMD **Rapid Virtualization Indexing** (**RVI**)
- Disable node interleaving if the system supports **Non-Uniform Memory Architecture** (**NUMA**)
- Enable Turbo Boost if the processor supports it
- Enable **Hyper-Threading** (**HT**) if it's supported by the processor
- Set Intel **Execute Disable** (**XD**) or AMD **No Execute** (**NX**) to **Yes**
- Set power saving features to **OS Control Mode**
- Enable the C1E halt state
- Disable any unnecessary hardware or features (floppy controllers, serial ports, USB controllers, and so on)

How it works…

Intel VTx, **Intel EPT**, **AMD-V**, and **AMD RVI** are hardware-based virtualization technologies that provide extensions to perform tasks that are normally handled by software to improve resource usage and enhance virtual machine performance. Enabling Intel VTx or AMD-V is required if the host will be running 64-bit guests.

If the design calls for disabling large memory pages to realize the advantages of **Transparent Page Sharing** (**TPS**) at times other than when there is memory contention, the Intel EPT or AMD RVI must be disabled. Enabling EPT or RVI will enforce large pages, even if they have been disabled in ESXi. This is not recommended for production environments, but can provide sufficient memory savings in lab or test environments.

If the system is NUMA-capable, the option to enable node interleaving, which will disable NUMA, may be available. Enabling NUMA by disabling node interleaving will provide the best performance. This ensures that memory that's accessed by a processor is local to that processor or in the same NUMA node as that processor.

Enabling Turbo Boost will increase efficiency by balancing CPU workloads over unused cores.

Intel Hyper-Threading allows for multiple threads to run on each core. When HT is enabled, the number of logical processors available is doubled. Each core is able to accept two concurrent threads of instructions.

Setting the power saving features to **OS Control Mode** will allow ESXi to manage power saving on the host. If **OS Control Mode** is not available or supported, power saving features should be disabled. Enabling the C1E halt state increases the power savings.

There's more…

If you're running Cisco UCS servers, it's useful to know that the default BIOS settings are already set to support ESXi out of the box. Therefore, important settings like hardware virtualization, no execute, NUMA, and others are already configured correctly. Other server vendors may be the same, but I encourage you to verify this. The default BIOS settings on UCS servers are a balance of performance and power management. For most workloads, you can expect these settings to perform well. If you expect to run **High-Performance Computing** (**HPC**), select workloads like massive **Online Transaction Processing** (**OLTP**), or similarly unique and performance-sensitive workloads, expect to dive deeply into the vendor documentation and white papers to tune your BIOS settings accordingly.

Upgrading an ESXi host

Many environments have already adopted some level of virtualization. A data center design will likely have to include upgrading the current infrastructure to leverage new features and functionality. Upgrading ESXi is a simple process, but it requires some planning to ensure compatibility and supportability. *Chapter 4*, *vSphere Management Design*, provides details about upgrading vCenter Server. This recipe will provide the details for planning an upgrade of ESXi hosts.

ESXi 5.x can be upgraded directly to ESXi 6.0 or 6.5. If you want to upgrade a host to 6.7 from 5.x, you'll first need an upgrade to 6.0 or 6.5, then another upgrade to 6.7.

How to do it…

When preparing to upgrade ESXi hosts, use the following steps:

1. Identify the methods that are available for upgrading ESXi
2. Verify hardware and firmware compatibility by using the VMware HCL, which can be found at `http://www.vmware.com/go/hcl`

How it works…

The following methods are available to upgrade ESXi:

- **Interactive upgrade**: An interactive upgrade can be performed from the ESXi console to upgrade ESXi from an ESXi image on a CD/DVD-ROM or USB drive. To perform an interactive upgrade boot to the image on the CD-ROM or USB drive, follow the onscreen wizard to upgrade ESXi.
- **Scripted upgrade**: ESXi upgrades can be scripted by using a kickstart file. The default kickstart file is located in `/etc/vmware/ks.cfg`. The `ks.cfg` file allows the upgrade to be performed from a CD/DVD-ROM, USB, FTP server, NFS server, or HTTP/HTTPS server. Details about creating a `ks.cfg` file and automating the ESXi upgrade can be found in the VMware vSphere documentation for installing or upgrading hosts by using a script, which is located at `http://pubs.vmware.com/vsphere-60/topic/com.vmware.vsphere.upgrade.doc/GUID-870A07BC-F8B4-47AF-9476-D542BA53F1F5.html`.
- **The command line with `esxcli`**: `esxcli` can be used to upgrade ESXi from a depot or bundle by using the following command:

```
esxcli software vib install -d <FullPathToUpgradeDepot>
```

- **vSphere Update Manager** (**VUM**): Using VUM, an ESXi image is uploaded, an upgrade baseline is created, the baseline is applied to hosts, and the hosts are remediated against the baseline. The *Designing a vSphere Update Manager deployment* recipe in *Chapter 4*, *vSphere Management Design*, provides details about installing and using VUM to upgrade ESXi hosts.
- **vSphere Auto Deploy**: Hosts that are deployed using vSphere Auto Deploy can be reprovisioned by using a new image profile. Creating an ESXi 6 image profile is covered in the *Creating a custom ESXi image* recipe, from earlier in this chapter.

Any time that an ESXi upgrade is performed, it is important to verify compatibility on the VMware HCL. The host hardware and BIOS and the installed adapters should be verified with the list to ensure the stability of the host and the support of the environment. Verifying compatibility using the HCL is covered in the *Using the VMware Hardware Compatibility List* recipe, from earlier in this chapter.

9
Virtual Machine Design

Virtual machine design is just as important as physical hardware design, and it should be part of the physical design process. Correctly designing and configuring virtual machines with proper resource allocation will help to increase consolidation in the virtual environment and ensure that a virtual machine has access to the resources that it requires to run the workloads efficiently.

A few questions that should be answered as a part of virtual machine design are as follows:

- What resources will be assigned to individual virtual machines?
- What virtual hardware will be allocated to virtual machines?
- How will new virtual machines be deployed?
- How will multiple virtual machines supporting an application be grouped based on dependencies?
- How will virtual machines be placed on host resources to ensure the efficient use of resources and availability?
- How will physical servers be converted into virtual machines?

This chapter will cover right-sizing virtual machines to ensure that they have the resources they require without over-allocating resources. We will also cover allocating virtual hardware to virtual machines and how to create a virtual machine template to quickly deploy a standardized virtual machine.

Configuring the ability to add CPU and memory resources without taking the virtual machine out of production will also be covered, along with how to group virtual machines into applications, or vApps. We'll also discuss using affinity and anti-affinity rules on a DRS cluster to reduce the demand on a physical network, or to provide application availability in the event of a host failure. Finally, in this chapter, we will demonstrate how to convert a physical server into a virtual machine.

In this chapter, we will cover the following recipes:

- Right-sizing virtual machines
- Enabling CPU hot add and memory hot plug
- Using paravirtualized VM hardware
- Creating virtual machine templates
- Installing and upgrading VMware Tools
- Upgrading VM virtual hardware
- Using vApps to organize virtualized applications
- Using VM affinity and anti-affinity rules
- Using VMs to Hosts affinity and anti-affinity rules
- Converting physical servers with vCenter Converter Standalone
- Migrating servers into vSphere

Right-sizing virtual machines

Right-sizing a virtual machine means allocating the correct amount of CPU, memory, and storage resources that are required to support a virtual machine's workload. The optimal performance of the virtual machine and the efficient use of the underlying hardware are both obtained through right-sizing virtual machine resources.

In a physical server environment, it can be difficult and time-consuming to add resources:

1. Compatible parts will need to be identified
2. The parts will need to be ordered
3. You will likely need to wait weeks for the parts to arrive
4. Someone will need to be in the data center
5. The server will need to be shut down
6. The server will need to be removed from the rack and opened up
7. The parts need to be installed

Because of this, physical servers are often configured with more resources than are actually required to ensure that there are sufficient resources available if the need for resources increases. Typically, physical servers only use a small percentage of the resources available to them; this means that a great deal of resources are constantly kept idle or are wasted. Adding resources to a physical server also typically requires that the server be powered off, and possibly even removed from the rack, which takes even more time and impacts production.

In a virtual environment, it is much easier to add CPU, memory, and disk resources to a virtual machine. This eliminates the need to over-allocate resources. Virtual machines are configured with the resources that they require, and more resources can be added quickly and easily as the demand increases. If a virtual machine has been configured to use CPU hot add and memory hot plug, additional resources can be added without taking the virtual machine out of production.

How to do it…

Perform the following steps to right-size virtual machines:

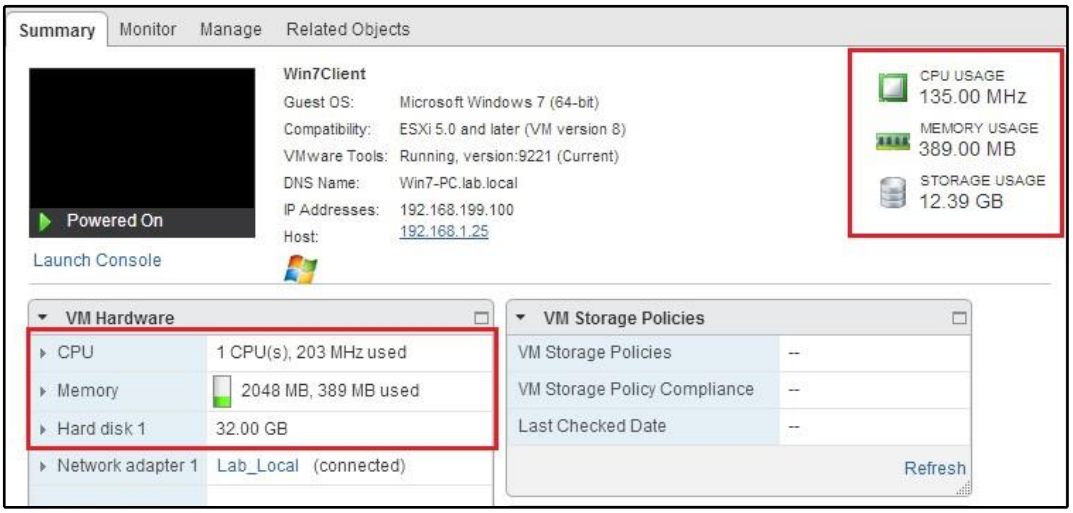
1. Determine the CPU, memory, and storage resources required by the virtual machine

When you are right-sizing virtual machine resources, start with the minimum requirements and add additional resources to the virtual machine as needed.

2. Adjust the virtual machine CPU, memory, and storage resource allocations to meet the requirements of the workload without over-allocating

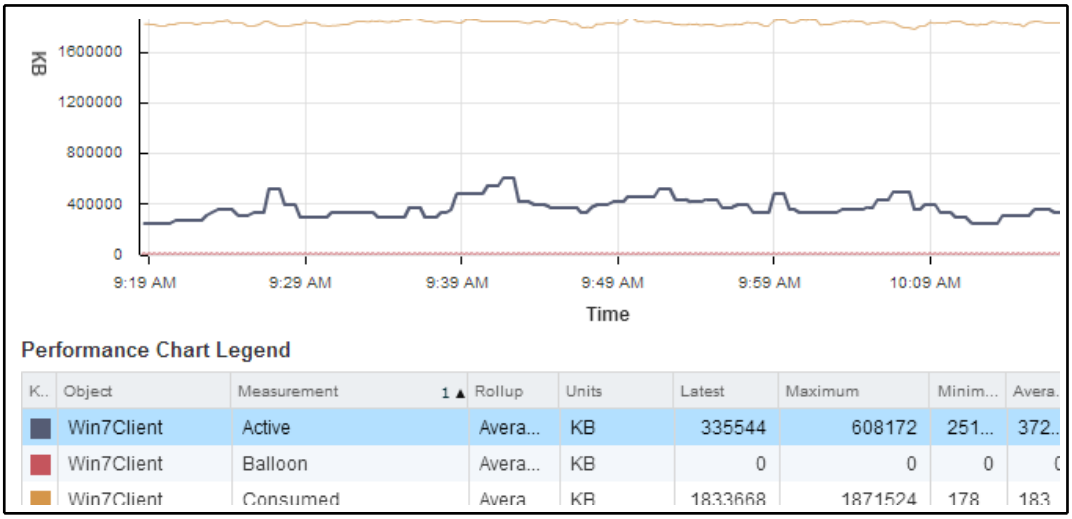
How it works…

Tools like VMware Capacity Planner or Windows perfmon can be used to determine the actual resources that are required by an application running on a physical server. The resources that are used by a virtual machine can be examined by using the vSphere Client program. From the **Summary** tab on the summary page of a virtual machine, it is easy to determine what CPU, memory, and disk resources have been allocated to the virtual machine, along with the current usage of each of these resources, as shown in the following screenshot:



Viewing resources allocated to a VM

Performance charts can also be used to provide information about CPU and memory usage over time. The real-time advanced memory performance chart that's shown in the following screenshot shows the memory metrics of the **Win7Client** virtual machine:



Viewing performance charts in the vSphere Web Client

The chart options can be adjusted to show metrics for the last day, week, month, or even year. These metrics can be used to determine whether a virtual machine has been allocated more memory than required.

Once the resource requirements have been identified, the virtual machine resources can be modified, or right-sized, to ensure that a virtual machine has not been allocated more resources than are required for the workload running on it.

vRealize Operations Manager (**vROps**) is a separate VMware product that can be used to monitor the resources that are used by a virtual machine, and it has capacity planning and efficiency monitoring that's specific to right-sizing virtual machines. More information on vROps can be found at `http://www.vmware.com/products/vcenter-operations-management/`.

Enabling CPU hot add and memory hot plug

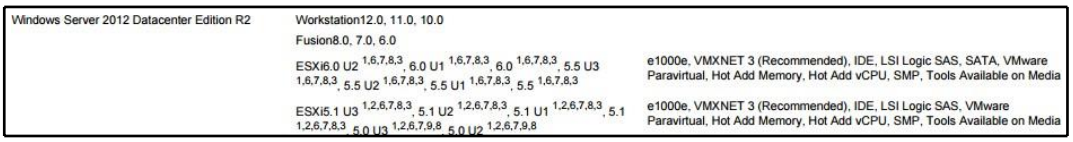
Adding CPU and memory resources to a virtual machine is a simple process. The process to add resources to a virtual machine is to power down the virtual machine, increase the number of vCPUs or the amount of memory, and power on the virtual machine again.

In vSphere 4.0, two new features, CPU hot add and memory hot plug, were introduced to allow for virtual machine vCPUs and virtual machine memory to be increased without requiring that the virtual machine be powered off. CPU hot add and memory hot plug must first be enabled on the virtual machine, which does require it to be powered off. Once it is enabled, however, CPU and memory resources may be added dynamically; powering off the virtual machine is not necessary.

How to do it…

Perform the following steps to enable CPU hot add and memory hot plug for virtual machines:

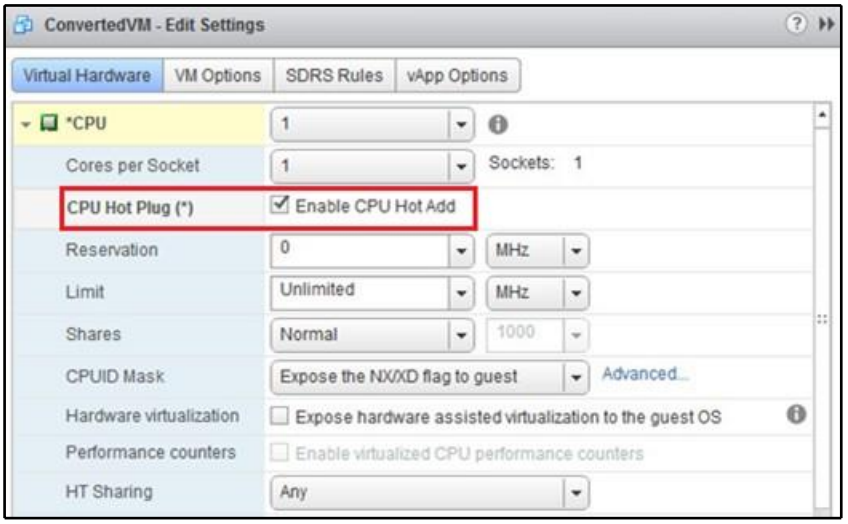
1. Check the VMware Guest OS Compatibility Guide, which can be found at `http://partnerweb.vmware.com/comp_guide2/pdf/VMware_GOS_Compatibility_Guide.pdf`, to identify whether vCPU and memory hot-adding is supported for the virtual machine guest operating system. The following screenshot is from the VMware Guest OS Compatibility Guide and shows the **Hot Add Memory** and **Hot Add vCPU** support for **Windows Server 2012 Datacenter Edition R2**:



Windows Server 2012 Datacenter Edition R2	Workstation12.0, 11.0, 10.0 Fusion8.0, 7.0, 6.0 ESXi6.0 U2 [1,6,7,8,3], 6.0 U1 [1,6,7,8,3], 6.0 [1,6,7,8,3], 5.5 U3 [1,6,7,8,3], 5.5 U2 [1,6,7,8,3], 5.5 U1 [1,6,7,8,3], 5.5 [1,6,7,8,3]	e1000e, VMXNET 3 (Recommended), IDE, LSI Logic SAS, SATA, VMware Paravirtual, Hot Add Memory, Hot Add vCPU, SMP, Tools Available on Media
	ESXi5.1 U3 [1,2,6,7,8,3], 5.1 U2 [1,2,6,7,8,3], 5.1 U1 [1,2,6,7,8,3], 5.1 [1,2,6,7,8,3], 5.0 U3 [1,2,6,7,9,8], 5.0 U2 [1,2,6,7,9,8]	e1000e, VMXNET 3 (Recommended), IDE, LSI Logic SAS, VMware Paravirtual, Hot Add Memory, Hot Add vCPU, SMP, Tools Available on Media

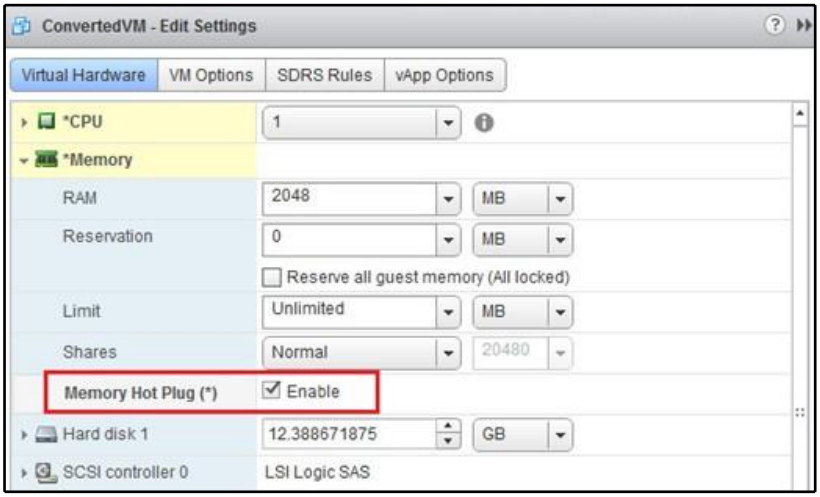
Viewing hot-add support for guest operating systems

2. To configure CPU hot add or memory hot plug for a virtual machine, it must be powered off.
3. To enable CPU hot add on the virtual machine, edit the virtual machine settings and expand the ***CPU** settings. Select the **Enable CPU Hot Add** checkbox, as shown in the following screenshot:



Enabling CPU Hot Add

4. To enable memory hot plug, expand the ***Memory** settings and select the **Enable** checkbox for **Memory Hot Plug (*)**, as shown in the following screenshot:



Enabling memory Hot Plug

5. Once CPU hot add and memory hot plug have been enabled for a virtual machine, the virtual machine can be powered back on.
6. vCPUs and memory can now be added to the running virtual machine without having to shut it down.

How it works…

Once CPU hot add and memory hot plug have been enabled on a virtual machine, it will not be necessary to power off the virtual machine to add additional vCPUs or additional memory.

Although vCPUs and memory can be added while the virtual machine is running, once Hot Add (or Hot Plug) has been enabled, some operating systems may require for the guest to be rebooted before the added vCPUs or memory are recognized by the operating system.

Enabling the CPU hot add and memory hot plug features does increase the virtual machine overhead reservation slightly. Also, remember that when a virtual machine's memory is increased, the virtual machine swap file (`.vswp`) also increases to the size of the allocated memory (minus any memory reservations). The swap file automatically grows when the virtual machine's memory is increased.

CPU resources do not have to be added in twos. If a virtual machine requires the resources that are associated with three CPUs, three vCPUs can be assigned to the virtual machine. It is also not necessary to allocate virtual machine memory in GB increments; a virtual machine can be allocated 1,256 MB of memory if that is what is necessary to meet resource requirements.

With CPU hot add and memory hot plug enabled, the removal of vCPUs and memory from a virtual machine will still require that the virtual machine be powered off or that the operating system be rebooted before the resources are removed, or before the removal of resources is recognized by the guest operating system.

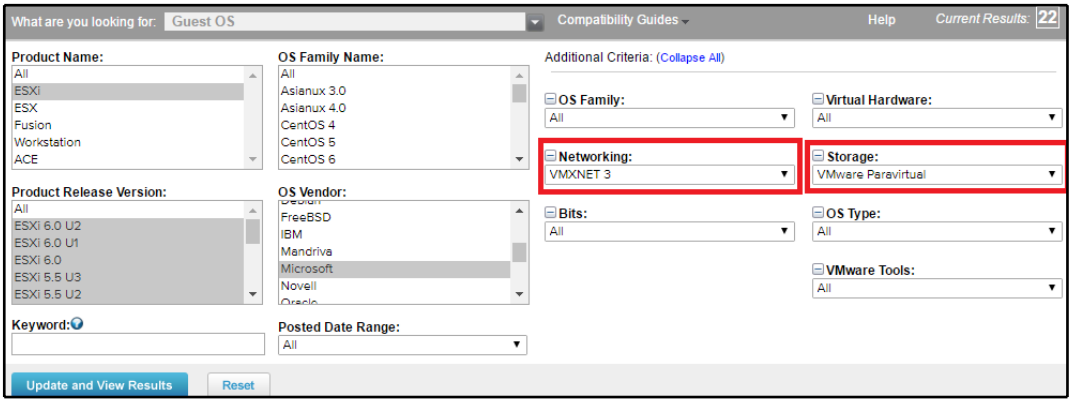
Using paravirtualized VM hardware

Paravirtualization provides a direct communication path between the guest OS within the virtual machine and the ESXi hypervisor. Paravirtualized virtual hardware and the corresponding drivers that are installed with VMware Tools are optimized to provide improved performance and efficiency. This hardware includes the VMXNET network adapter and the PVSCSI storage adapter.

How to do it…

Adding paravirtualized hardware adapters to a virtual machine is done by using the following process:

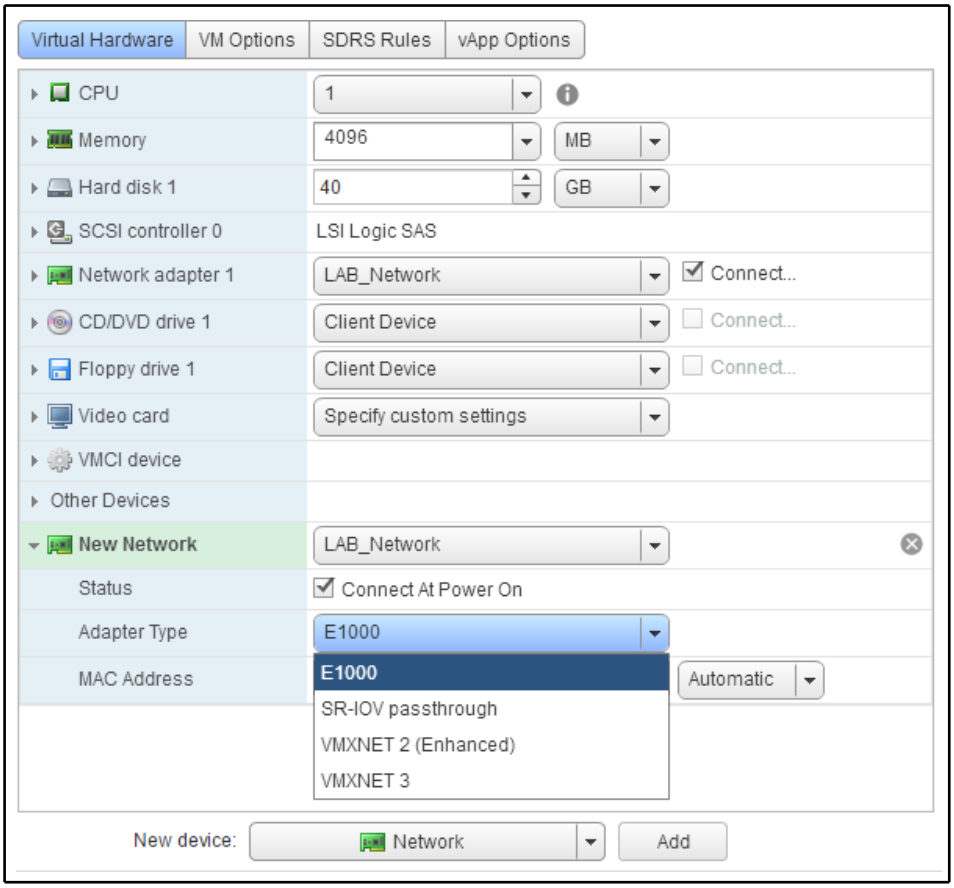
1. Access the **Guest OS** compatibility section of the VMware HCL at `http://www.vmware.com/go/hcl` to determine guest OS support for paravirtual adapters. A screenshot of the **Guest OS** compatibility HCL, along with the **Networking** and **Storage** adapters highlighted in red boxes, is as follows:



Verifying paravirtual network support for guest operating systems

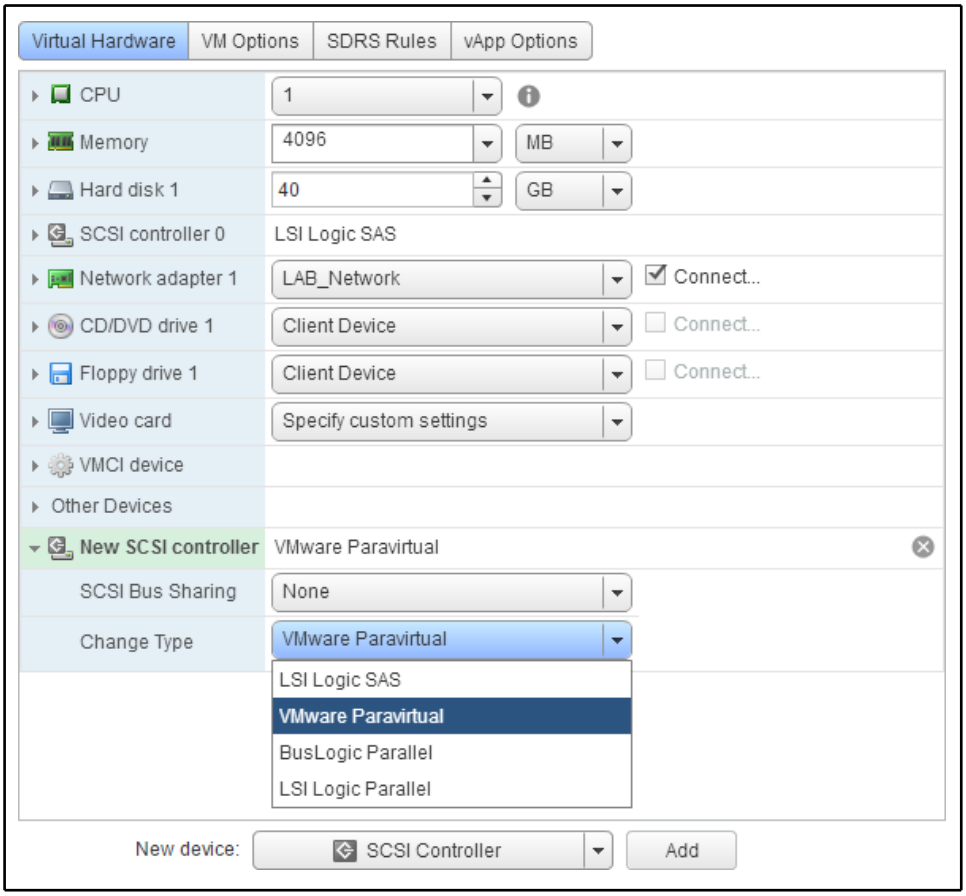
2. Install VMware Tools in the virtual machine.

3. To install the paravirtualized network adapter, edit the virtual machine's **Virtual Hardware**, add a **New Network adapter**, and select **VMXNET3** for the **Adapter Type**, as shown in the following screenshot:



Adding a paravirtual network adapter to a VM

4. To install the paravirtualized SCSI adapter, edit the virtual machine's **Virtual Hardware**, add a **New SCSI controller**, and select **VMware Paravirtual** for the **Adapter Type**, as shown in the following screenshot:



Adding a paravirtual SCSI controller to a VM

How it works…

Once the adapter has been configured, the optimized virtual hardware is presented to the virtual machine guest. The drivers for the paravirtualized hardware adapters are included with VMware Tools. The paravirtualized hardware can be added to a virtual machine before VMware Tools is installed, but the hardware will not be available for use by the guest OS until VMware Tools is installed.

The VMXNET3 adapter provides higher network throughput with less host CPU overhead.

The PVSCSI adapter is suitable for I/O intensive applications. Like the VMXNET3 adapter, the PVSCSI adapter increases storage throughput with minimal host CPU overhead.

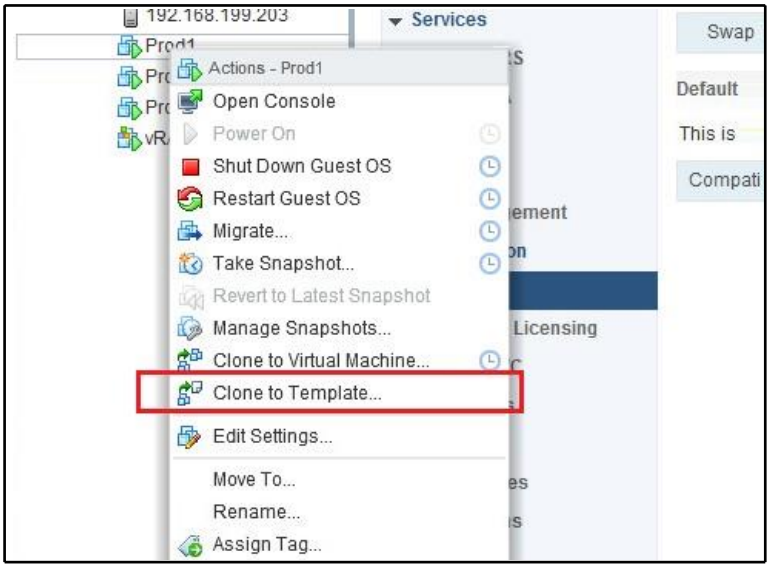
Creating virtual machine templates

Virtual machines can be deployed quickly and in a standardized fashion by using pre-built templates. Virtual machine templates are configured with minimum CPU, memory, and storage resources. The guest operating system and any prerequisite applications are installed in the template. Instead of taking hours (or even days, in some cases) to install the operating system and prepare the server, once a template has been created, a new virtual machine can be deployed within minutes. Virtual machine templates not only allow for quick deployment, but also help to maintain consistency across virtual machines that are deployed in the environment.

How to do it…

The following steps are required to create a virtual machine template:

1. Create a virtual machine; configure the vCPU, memory, and storage resources; install the guest operating system; install the required applications; and apply any application or operating system updates or patches.
2. The virtual machine can be cloned to a template by using the **Clone to Template** wizard, as shown in the following screenshot:

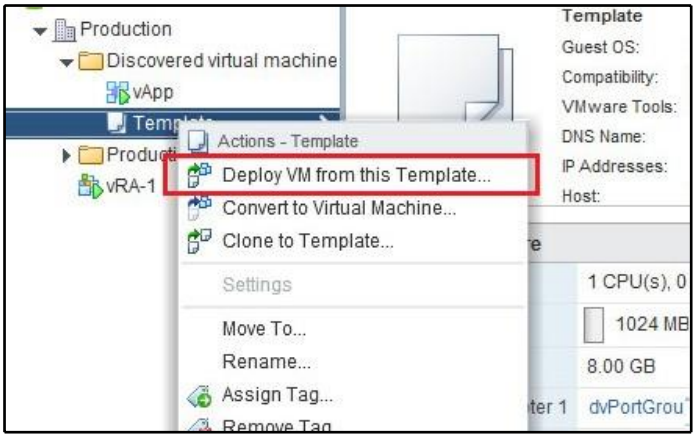


Context menu to clone a VM to template

3. The virtual machine can also be converted into a template by using the **Convert to Template** wizard, as shown in the following screenshot:

Context menu to convert a VM to template

4. Once the virtual machine template has been created, new virtual machines can be deployed from the template by using the **Deploy VM from this Template** wizard, as shown in the following screenshot:



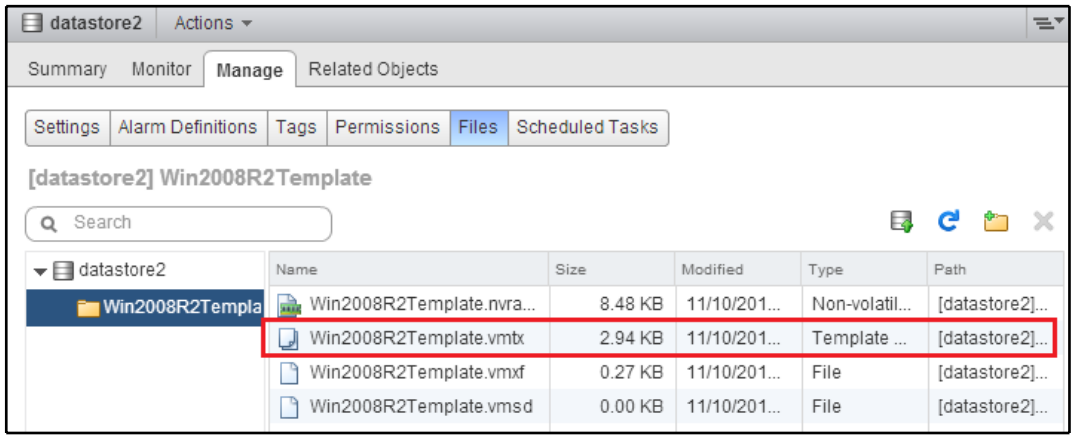
Context menu to deploy a VM from a template

How it works…

When cloning a virtual machine to a template, the **Clone to Template** wizard allows the administrator to choose the data center, cluster, and storage to create the new virtual machine template. Cloning a virtual machine to a template can be done while the source virtual machine is powered on.

When a virtual machine is converted into a template, the virtual machine is converted locally; the template will have the same inventory properties (data center, cluster, and storage) as the virtual machine that has been converted. The virtual machine configuration file (`.vmx`) is changed to a template configuration file (`.vmtx`) when a virtual machine is converted. The virtual machine needs to be powered off to be converted to a virtual machine template.

The following screenshot shows the virtual machine template files on a **datastore** (the virtual machine template configuration file has been boxed in red):



VM template files on datastore

The virtual machine template file is similar to the `.vmx` file, and it contains configuration information about the virtual hardware that's presented to the virtual machine or template.

There's more…

A guest customization specification can be applied to a virtual machine that is being deployed from a template. The customization specification allows for settings that are unique to the deployed virtual machine to be applied during the deployment process. These custom specifications include information such as the computer's name, the licensing, the IP address, and the domain membership.

The **New VM Guest Customization Spec** wizard is displayed in the following screenshot:

Sections of a VM guest customization specification

The customization specification can be saved so that it can be applied to future virtual machines that are deployed from templates. Guest customization specifications can also be applied when cloning a virtual machine.

Upgrading and installing VMware Tools

VMware Tools enhances the performance and improves the management of virtual machines. It does this by loading optimized drivers for virtual hardware and installing utilities to access virtual machine configurations and metrics. VMware Tools is not required, but for optimal virtual machine performance, it should be installed on all virtual machines in the environment.

The status of VMware Tools is displayed on the virtual machine's **Summary** page, as shown in the following screenshot:

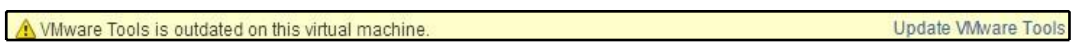
Status of VMware Tools on the VM summary tab

The drivers for the optimized paravirtual hardware, such as the VMXNET3 adapter and the PVSCSI adapter, are included in VMware Tools, and VMware Tools must be installed before the hardware is available for use within the guest. VMware Tools should be installed and kept up to date for every guest operating system.

How to do it…

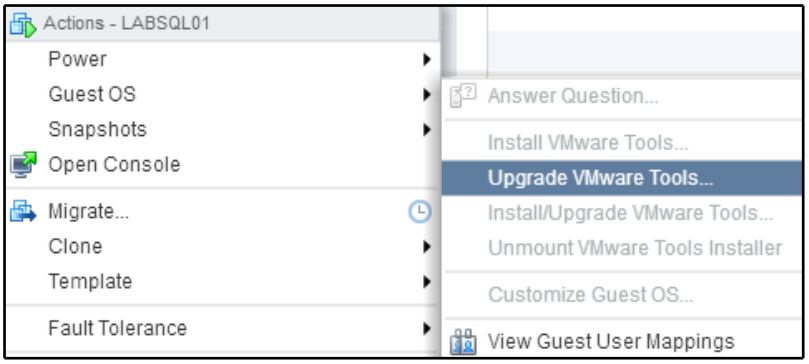
There are multiple options available for upgrading VMware Tools:

1. If VMware Tools is out of date, a warning is displayed in the vSphere Client. VMware Tools can be updated from the VM's **Summary** page by using the **Update VMware Tools**, as shown in the following screenshot:



VMware Tools is out of date warning

2. Right-clicking on the virtual machine and selecting the **Guest OS** menu will allow you to **Install** or **Upgrade VMware Tools**, as shown in the following screenshot:



VM context menu to upgrade VMware Tools

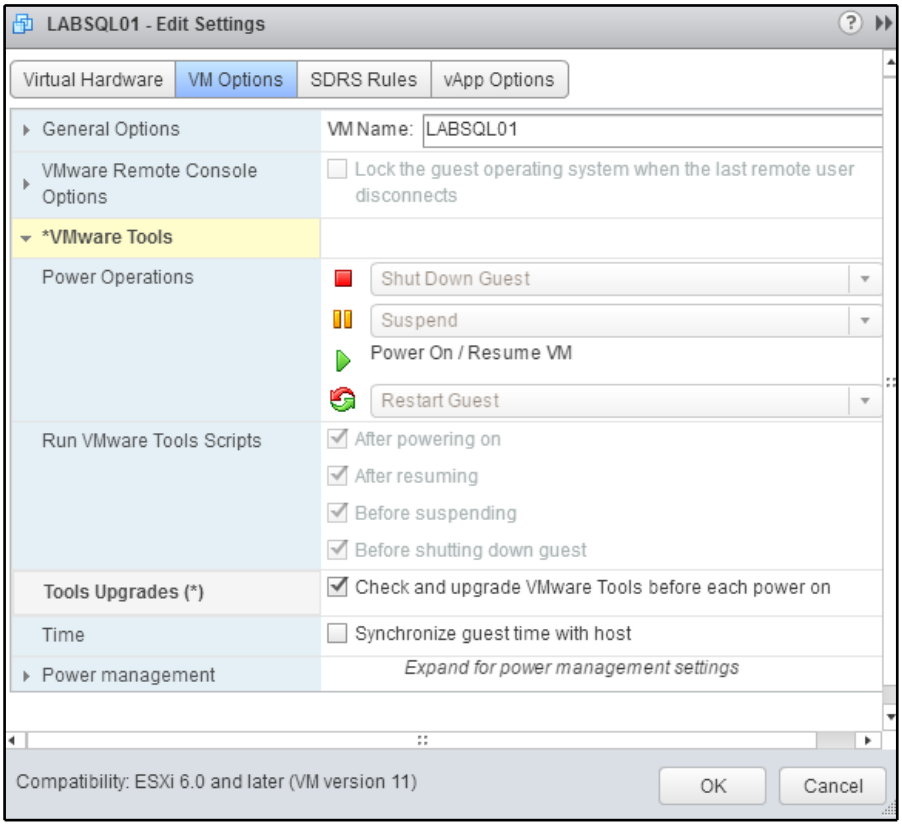
3. If VMware Tools is not installed, the **Install VMware Tools** option will be available.
4. Update or install VMware Tools as required.

How it works…

When upgrading or installing VMware Tools on a virtual machine, a VMware Tools ISO image is connected to the virtual machine. If VMware Tools is already installed, the upgrade automatically runs to update the VMware Tools to the current version. If VMware Tools is not already installed, the installer must be manually run to install VMware Tools. A reboot of the virtual machine will be required once the VMware Tools installation has completed.

There's more…

Virtual machines can be configured to automatically upgrade VMware Tools to the current version. This is done by editing the virtual machine's settings and selecting the **Check and upgrade VMware Tools before each power on** in **VM Options**, as shown in the following screenshot:



Option to automatically upgrade VMware Tools before each power on operation

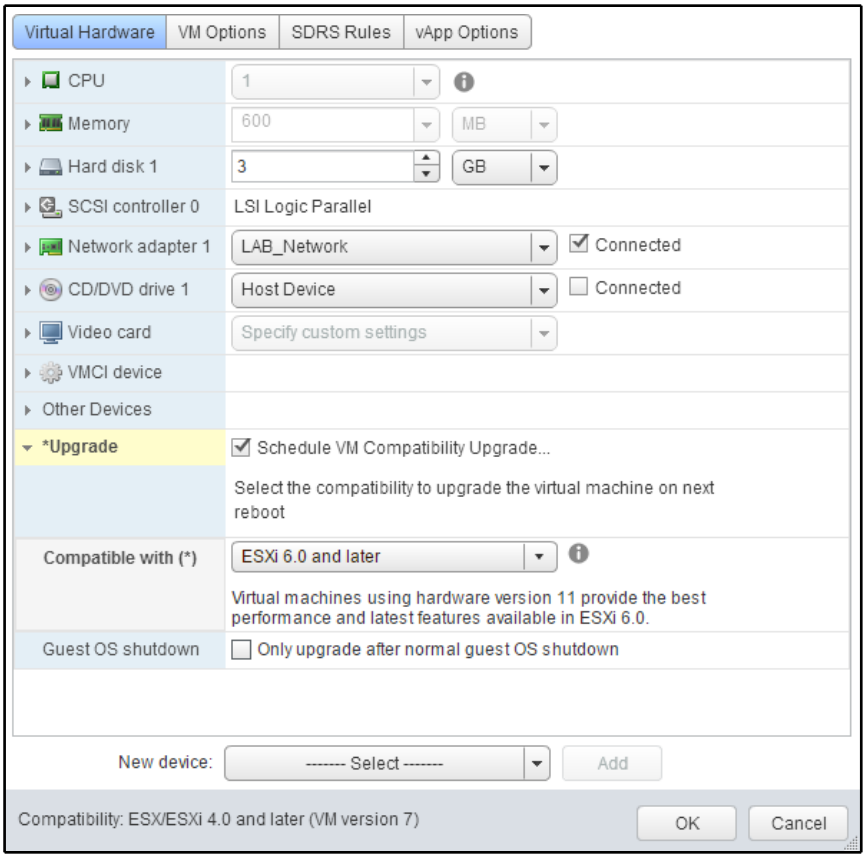
Upgrading VM virtual hardware

The virtual machine hardware version or virtual machine compatibility specifies the version of virtual machine hardware that's presented to the virtual machines and the ESXi versions that the virtual machine is then compatible to run on. Updating the virtual machine hardware exposes new features that are available to virtual machines (for example, the ability to provision vmdks up to 62 TB) and ensures that the virtual hardware is optimized to the version of ESXi.

How to do it…

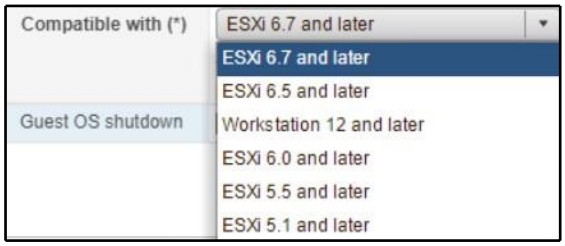
To upgrade the virtual hardware of a virtual machine, use the following steps:

1. Take a snapshot of the virtual machine to ensure that you can fall back to a known good state if the upgrade fails.
2. It's critically important to install or update VMware Tools in the virtual machine before the virtual hardware upgrade.
3. Edit the settings of a virtual machine and access the **Virtual Hardware** tab.
4. If a virtual hardware upgrade is available, the **Upgrade** option will be available. Select the **Schedule VM Compatibility Upgrade**, as shown in the following screenshot:



VM option to schedule virtual hardware upgrades on next reboot

5. Set the compatibility level that the virtual hardware should be upgraded to, as shown in the following screenshot:



Selecting virtual hardware compatibility

6. Shut down and power on the virtual machine to upgrade the virtual machine hardware.

How it works…

The hardware compatibility maps to a virtual hardware version. The following table shows the relationships between the compatibility and the virtual machine hardware version:

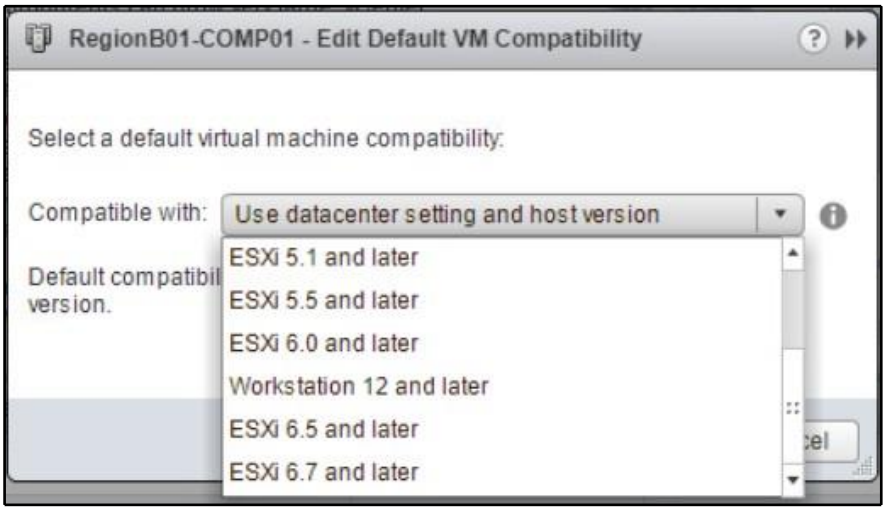
Virtual Machine Hardware Version	Compatibility
VM version 14 (vmx-14)	ESXi 6.7 and later
VM version 13 (vmx-13)	ESXi 6.5 and later
VM version 11 (vmx-11)	ESXi 6.0 and later
VM version 10 (vmx-10)	ESXi 5.5 and later
VM version 9 (vmx-9)	ESXi 5.1 and later
VM version 8 (vmx-8)	ESXi 5.0 and later
VM version 7 (vmx-7)	ESX/ESXi 4.0 and later
VM version 4 (vmx-4)	ESX/ESXi 3.5 and later

When a virtual hardware upgrade is scheduled, the virtual hardware of the virtual machine is upgraded the next time the virtual machine is rebooted.

The virtual machine hardware version should be set to the compatibility of the lowest version of ESXi in the environment to ensure that the virtual machine can run on any host in the environment. For example, if a design includes both 6.0 and 6.7 hosts and a virtual machine's hardware is upgraded to vmx-14, the VM will no longer run on the ESXi 6.0 hosts.

There's more…

The default VM compatibility can be set on a vSphere data center or cluster. Setting the default virtual machine compatibility is done with the **Edit Default VM Compatibility** dialog, which can be accessed by right-clicking on the data center or cluster in the vCenter inventory. The **Edit Default VM Compatibility** dialog is shown in the following screenshot:



Choosing the default virtual hardware compatibility for a cluster

Once a default VM compatibility is set on a data center or cluster, the virtual machines that are deployed in the data center and cluster will be deployed with the default virtual machine hardware, based on the compatibility settings.

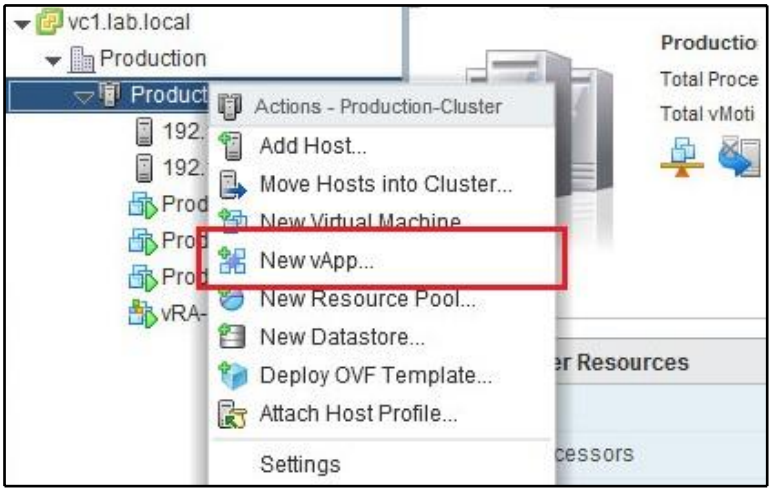
Using vApps to organize virtualized applications

vApps can be used to group individual virtual machines with interdependencies into a single application. A common use case for this would be a multi-tier web application that requires a web server frontend, an application server, and the supporting database server. The application can then be managed as a single inventory object.

How to do it…

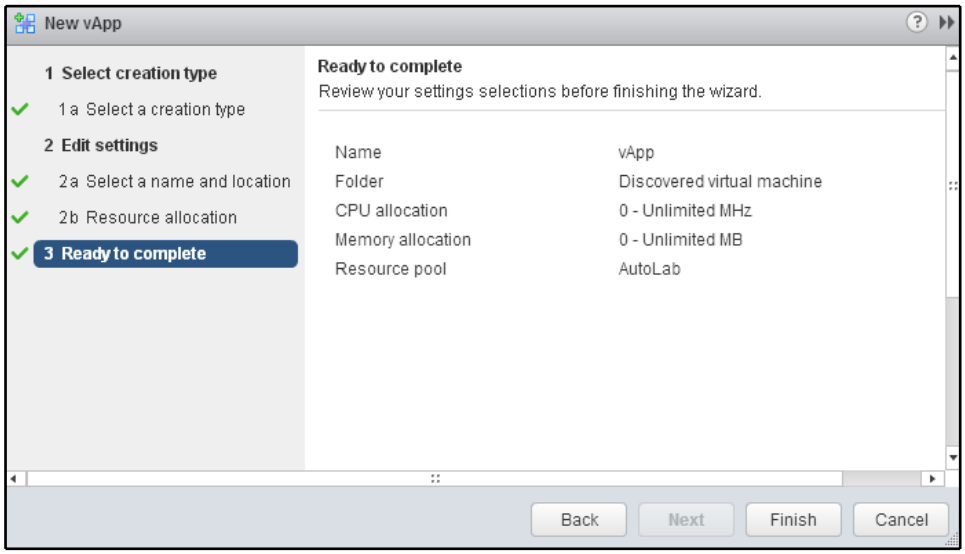
Perform the following steps to use vApps to organize virtual machine workloads:

1. Create a new vApp by launching the **New vApp…** wizard, as shown in the following screenshot:



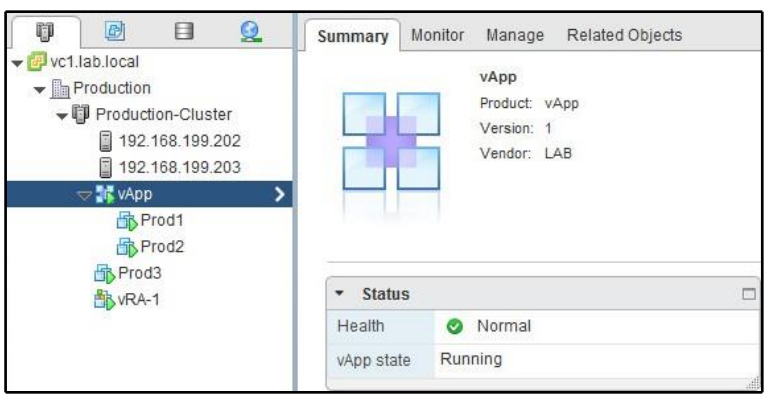
Context menu to create a new vApp

2. The method for creating the vApp (either creating a new vApp or cloning an existing vApp), the vApp **Name**, the **Folder** location, and the resource allocation settings, are configured in the **New vApp** wizard, as shown in the following screenshot:



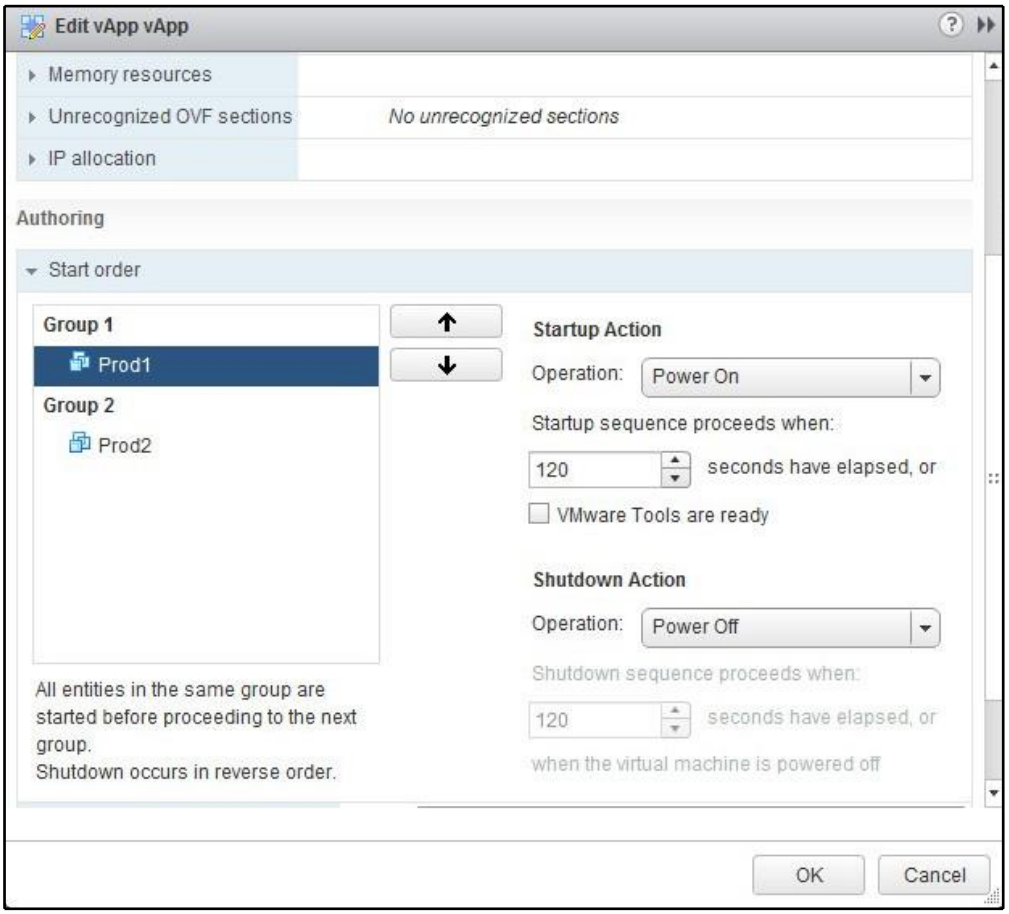
Completing the New vApp wizard

3. Once the vApp has been created, you can add virtual machines to the new vApp by dragging them into the vApp. The following screenshot shows a **vApp** containing the **Prod1** and **Prod2** virtual machines:



vApp summary tab

4. The settings of the vApp can be edited. In the following screenshot, the **Start order** is configured to start the virtual machines in the vApp in a specific order. The **Start order** ensures that virtual machines are started in order of their dependencies when the vApp is powered on:



Editing a vApp for VM startup order

How it works…

A **vApp** is a container of the virtual machines that support an application. Once placed in a vApp, startup and shutdown can be configured based on application dependencies, resources can be reserved or limited, and the entire vApp can be exported in an OVA or OVF format. A vApp can also be cloned to duplicate the application.

Using VM affinity and anti-affinity rules

When virtual machines are powered on in a DRS cluster, vCenter determines where the virtual machines should be placed to balance resource usage across the cluster. The DRS scheduler runs periodically to migrate virtual machines using vMotion. The main purpose of DRS is to ensure that virtual machines are receiving the resources they request and to maintain a balance of resource usage across the cluster. Affinity or anti-affinity rules can be used to control where VMs are placed within a cluster. Affinity rules keep VMs on the same physical host, reducing the load on the physical network by keeping traffic between them from leaving the host. Anti-affinity rules keep VMs separated on different physical hosts, ensuring higher availability.

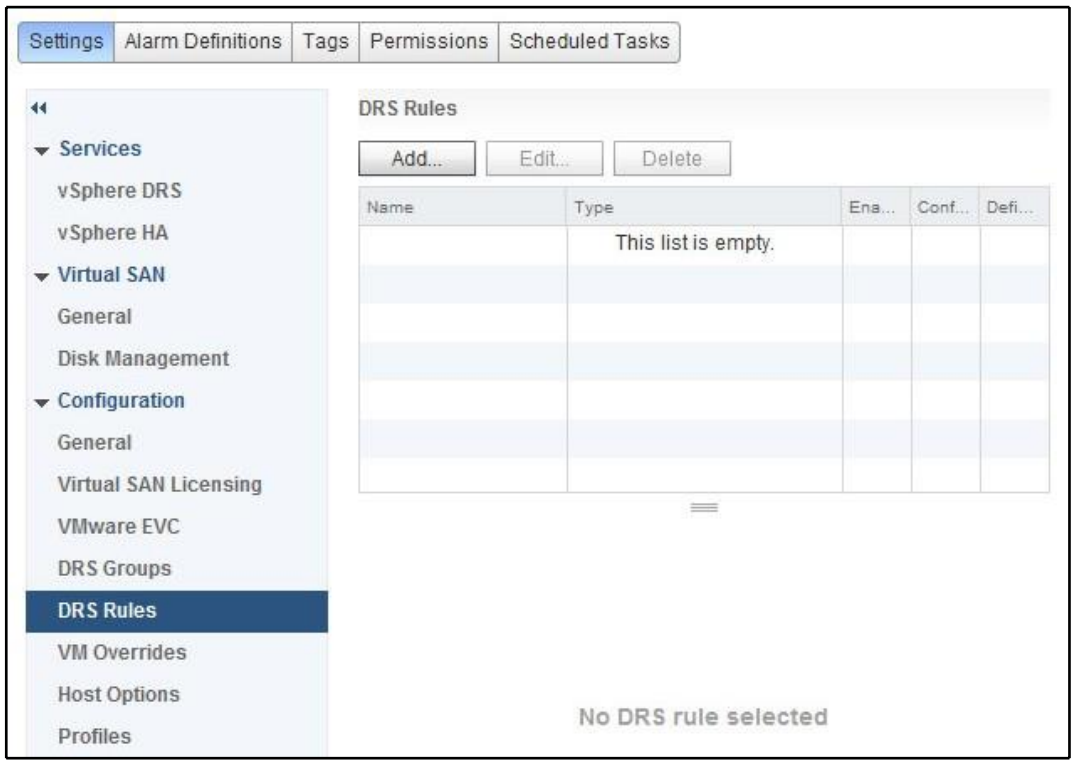
One use case of an affinity rule would be to keep all of the virtual machines supporting an application on the same host. This would ensure that the network communications between the virtual machines supporting the application do not traverse the physical network.

An example use case of an anti-affinity rule would be to keep multiple virtual Active Directory domain controllers running on separate hosts to ensure that not all of the domain controllers are affected by a single host failure.

How to do it…

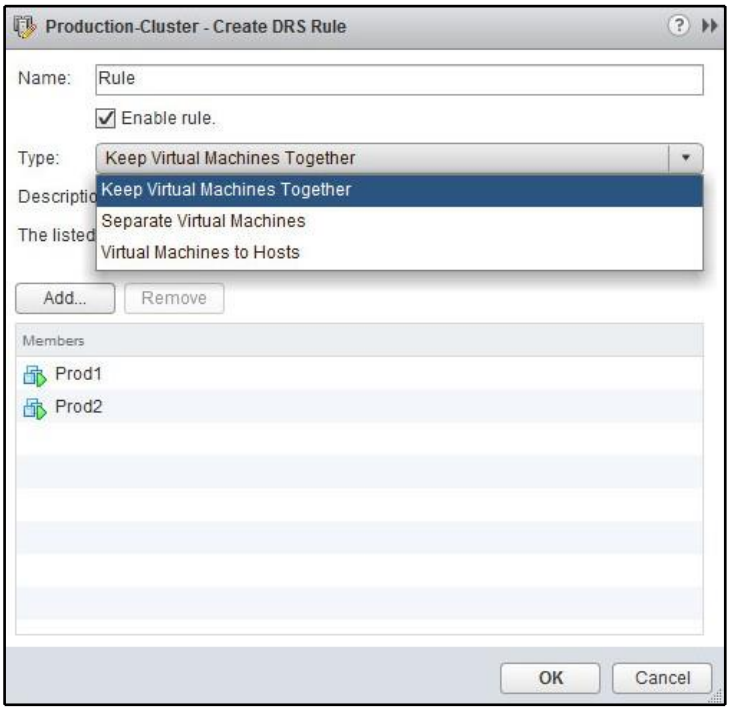
The following steps are required to use VM affinity and anti-affinity rules:

1. DRS rules are created on the **Settings** page of a DRS-enabled cluster, as shown in the following screenshot:



DRS rules section for a cluster

2. DRS rules can be created for three purposes: to **Keep Virtual Machines Together** on the same host, to **Separate Virtual Machines** across different hosts, or to assign **Virtual Machines to Hosts**, as shown in the following screenshot:



Types of DRS rules

How it works…

With the DRS rules configured, the distributed resource scheduler will apply the rules when determining the placement of virtual machines when they are powered on or when migrating virtual machines to other hosts to balance cluster resource usage.

When an affinity rule has been configured to keep several virtual machines together, if DRS migrates one of the virtual machines in the rule, all of the virtual machines that are configured will also be migrated to the new host. When an anti-affinity rule has been configured to keep virtual machines separated, DRS will not migrate a virtual machine to a host running another virtual machine that's been configured in the rule.

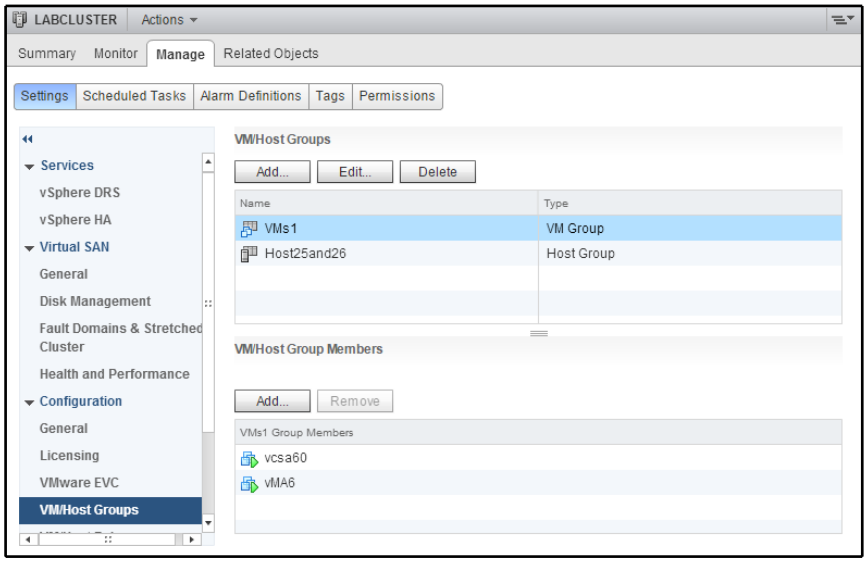
Using VM to Host affinity and anti-affinity rules

Virtual Machine to Hosts rules can be created to keep virtual machines on or off specific hosts (or groups of hosts). These types of DRS rules are useful for keeping the management of virtual machines, such as vCenter Server, on specific hosts to make those virtual machines easier to locate in the event of a failure. This also allows for virtual machines to be separated across different hosts in a rack or blade chassis to ensure that the loss of a rack or chassis does not impact all virtual machines; for example, to split the members of a Microsoft SQL Always On Availability Group across chassis.

How to do it…

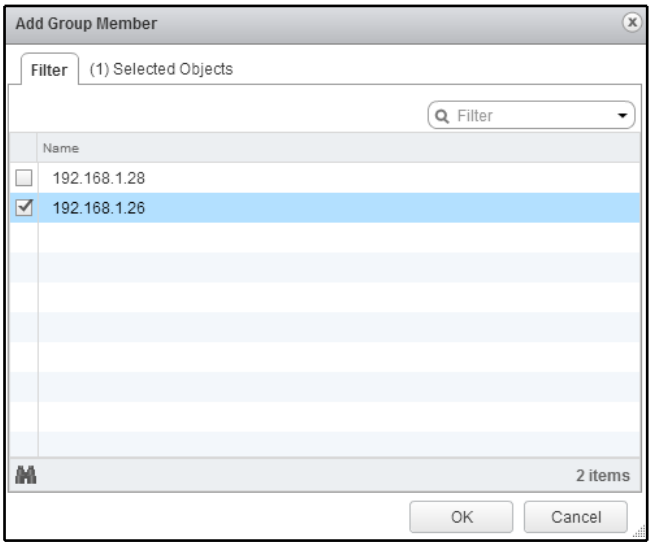
The following process can be used to create **Virtual Machine to Hosts** affinity or anti-affinity rules:

1. From the cluster's **Settings** page, access the **VM/Host Groups** section to manage virtual machine and host groups, as shown in the following screenshot:



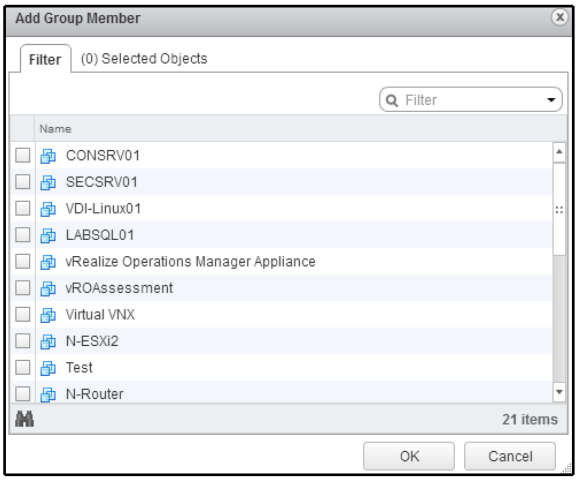
Managing VM / Host groups

2. Select **Add** to create a new host group and select the hosts to add to the group, as shown in the following screenshot:



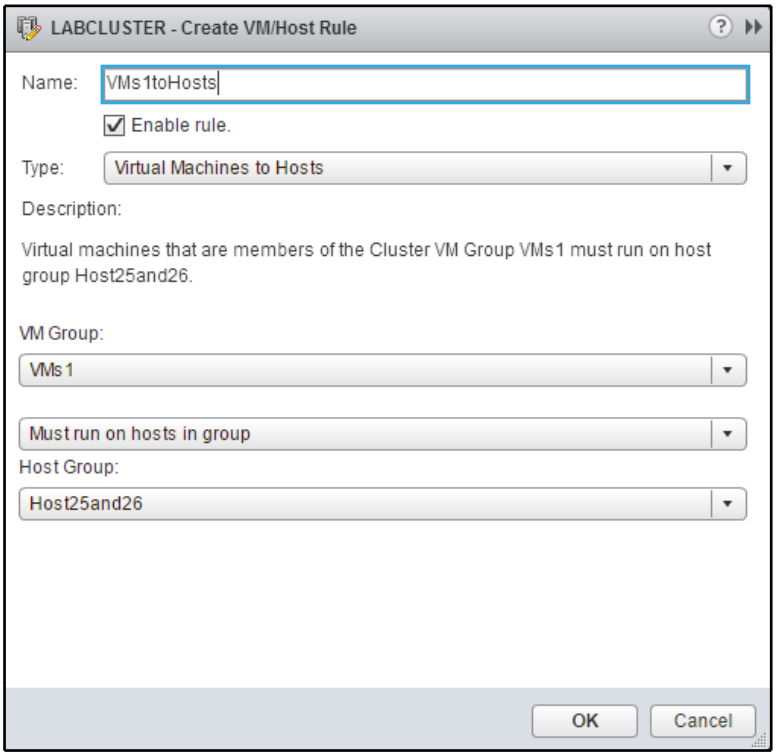
Adding a new host group

3. Select **Add** to create a new VM group, and select the virtual machines to add to the group, as shown in the following screenshot:



Adding a new VM group

4. From the **DRS Rules** menu, create a **VM/Host Rule** by providing a rule **Name**, setting the **Type** to **Virtual Machines to Hosts**, selecting the **VM Group** and **Host Group** to include in the rule, and selecting the required or preferential affinity/anti-affinity rules, as shown in the following screenshot:

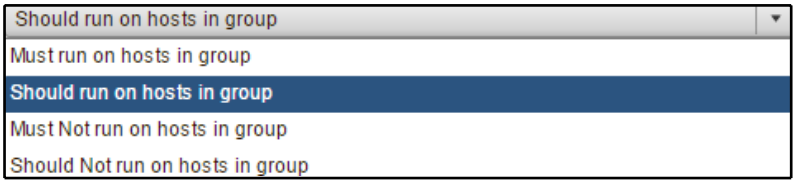


Creating a VM / Host rule

5. Click on **OK** to create the **Virtual Machines to Hosts** rule.

How it works…

When creating **Virtual Machines to Hosts**, the affinity or anti-affinity rules can be set to be required or preferential, as shown in the following screenshot of the dropdown box from the **Create VM/Host Rule** wizard:



Should or Must DRS rule options

The VM/Host affinity and anti-affinity rules are as follows:

- **Must run on hosts in group**: This is a required VM to Host affinity rule, and the virtual machines must run on the specified hosts. This rule will not be violated, even in an HA event.
- **Should run on hosts in group**: This is a preferential VM to Host affinity rule, and the virtual machines will run on the selected host when possible. The virtual machines can run on hosts outside of the group if necessary (for example, in an HA event).
- **Must Not run on hosts in group**: This is a required VM to Host anti-affinity rule, and the virtual machines will not run on hosts within the group. This rule will not be violated, even in an HA event.
- **Should Not run on hosts in group**: This is a preferential VM to Host anti-affinity rule, and the virtual machines can run on hosts within the group if necessary (for example, in an HA or maintenance event).

Once the VM/Host rules have been created, vSphere DRS will apply the rules when managing virtual machine placement and resource balancing within the vSphere cluster.

Converting physical servers with vCenter Converter Standalone

There are two methods for virtualizing the workloads that are running on physical servers. The workloads can be migrated into the virtual environment by creating new virtual machines, loading a guest operating system, installing applications, and migrating the application data to the new virtual machines; or, physical servers can directly be converted into virtual machines by using VMware vCenter Converter Standalone, or similar third-party tools.

How to do it…

The following steps are required to use VMware vCenter Converter Standalone:

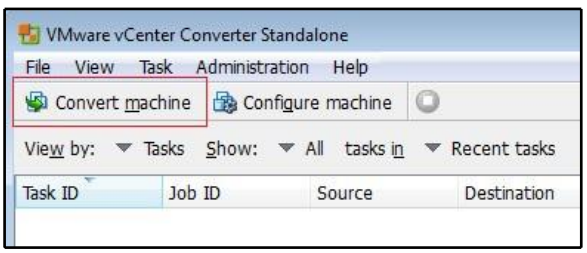
1. Download VMware Converter from `http://www.vmware.com/web/vmware/downloads`. VMware Converter can be installed as either a local installation or a client-server installation. More information on installing VMware Converter is available in the VMware vCenter Converter Standalone guide, at `http://www.vmware.com/support/pubs/converter_pubs.html`.

The local installation is used to convert the physical machine on which the converter software is installed. When it is installed using the client-server installation, the local machine becomes a server that can be managed remotely, which uses the Converter Standalone client to convert the physical servers.

2. Once VMware Converter has been installed, the **VMware vCenter Converter Standalone** client is used to connect to the converter server, which is either local or installed on a remote machine. The **VMware vCenter Converter Standalone** login dialog is shown in the following screenshot:

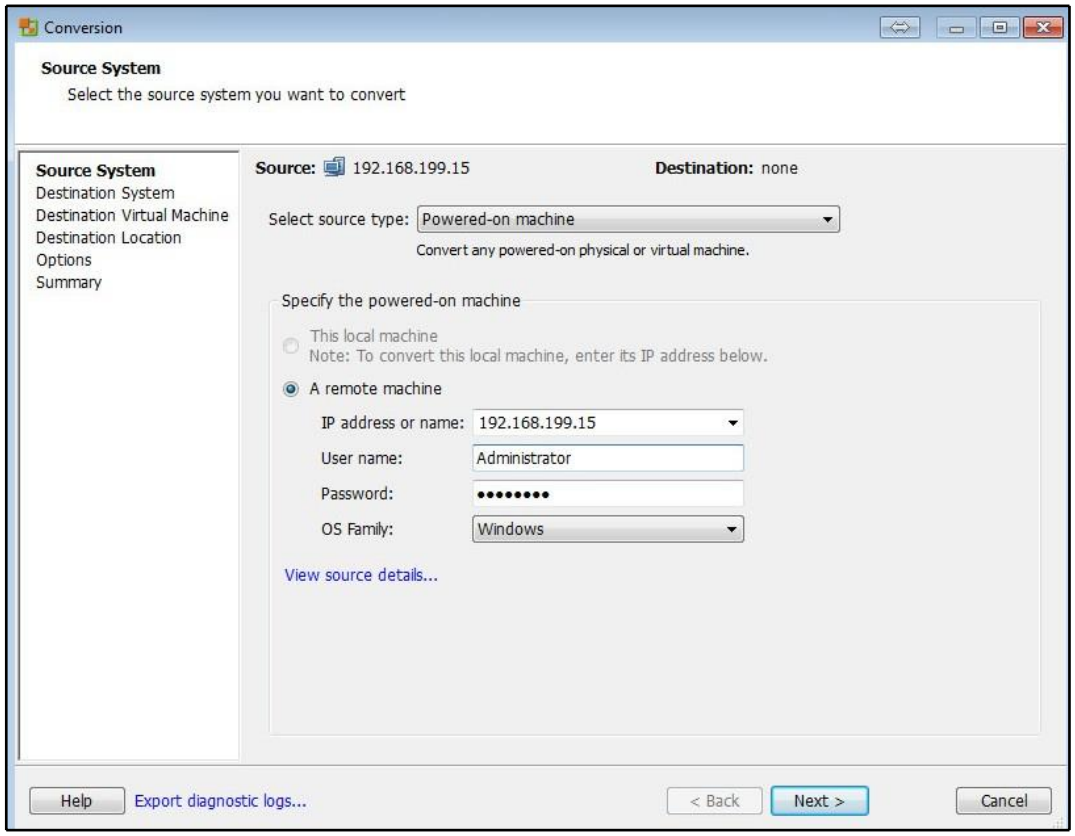
VMware vCenter Converter connection page

3. To convert a machine, select **Convert machine** to start the conversion wizard, as shown in the following screenshot:



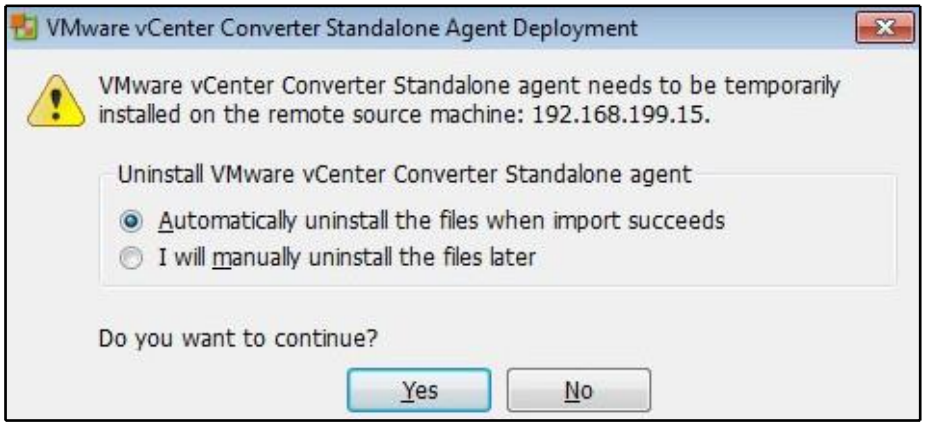
Convert machine option in vCenter Converter

4. The first step of the conversion is to select the **Source System** type. This is the type of system that will be converted into the virtual environment. The source type can be **Powered-on machine** (physical or virtual), **VMware Infrastructure virtual machine**, **Backup image or third-party virtual machine**, or **Hyper-V Server**.
5. Once the source type has been selected, specify the powered-on virtual machine's information. This is the machine that will be converted, and it can be either the local machine or a remote machine. To convert a remote machine, the **IP address or name** field must be filled in, along with administrator credentials and the **OS Family** field. When converting the local machine, the user running the converter must have administrator access to the local machine. The following screenshot shows a sample **Source System** configuration to convert a remote powered-on machine:



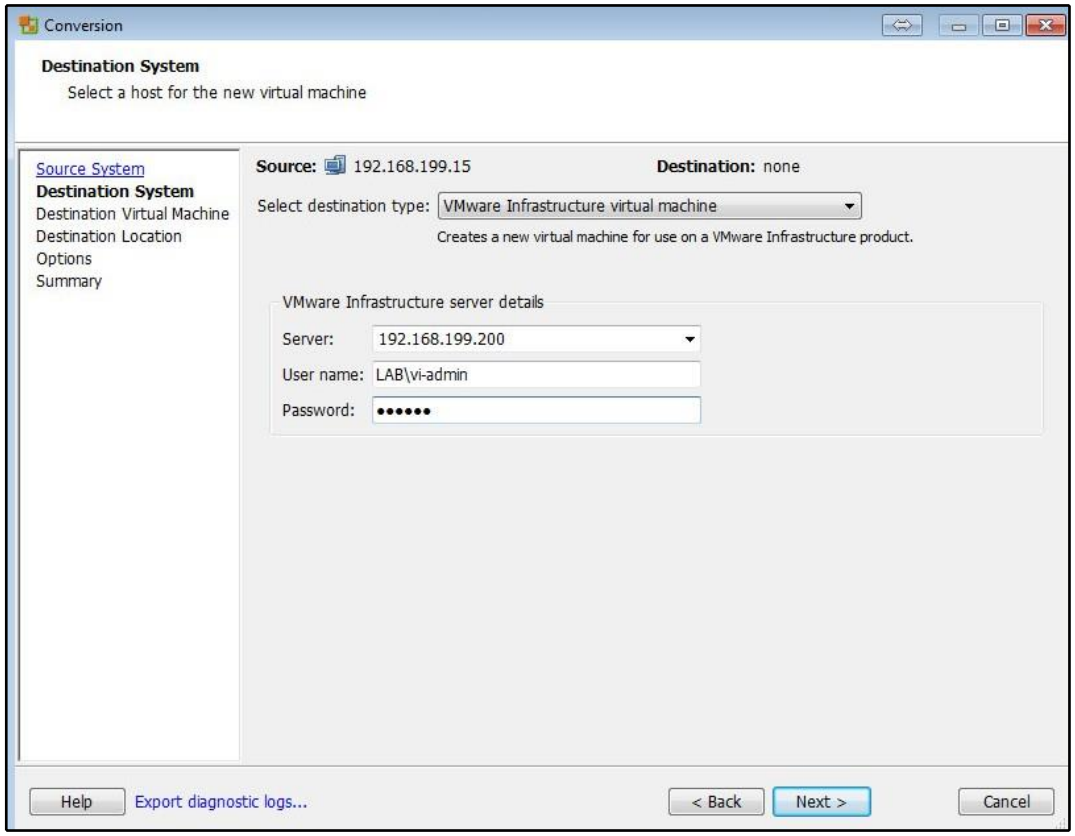
Configuring source system in a vCenter Converter job

6. Once the source system information has been provided, the Converter Standalone agent will be installed on the source system. Once the conversion has finished, the agent can be uninstalled automatically or manually:



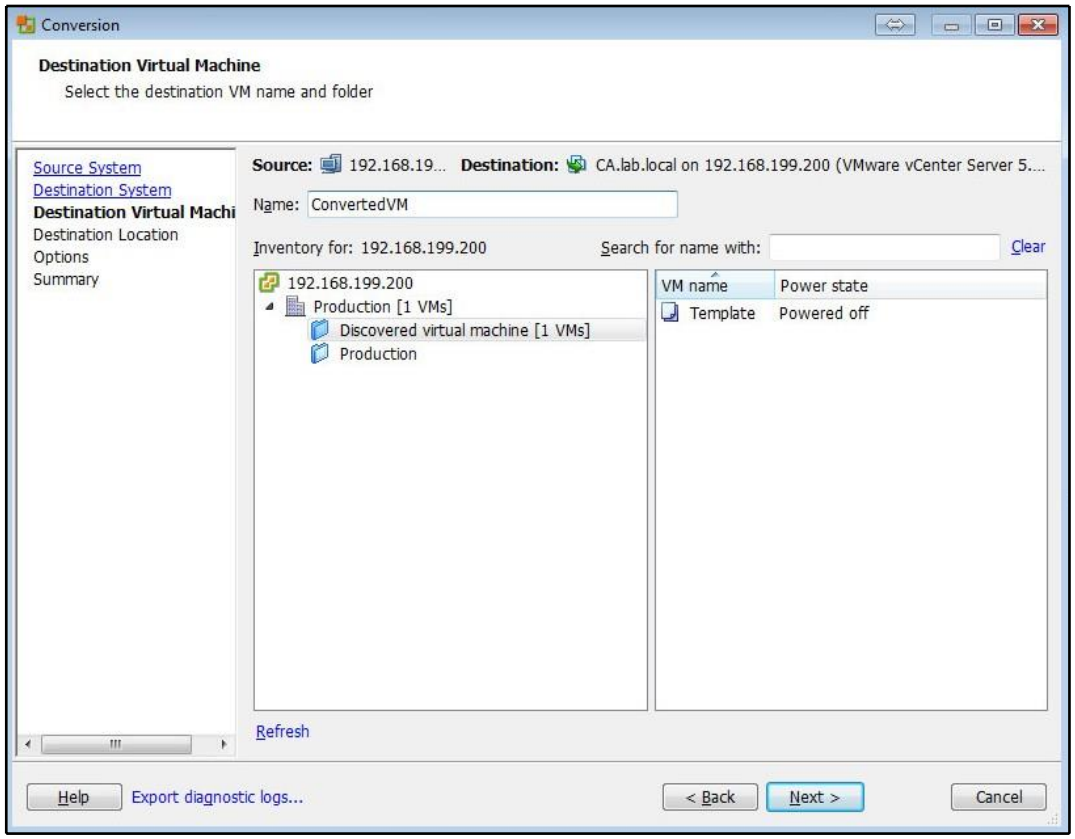
Choosing vCenter Converter uninstall option

7. Once the Converter Standalone agent has been successfully installed, the destination system where the source system will be converted will be configured. The destination type, the destination IP address, and the destination credentials will be configured. The following screenshot shows the configuration of a vCenter Server as the **Destination System**:



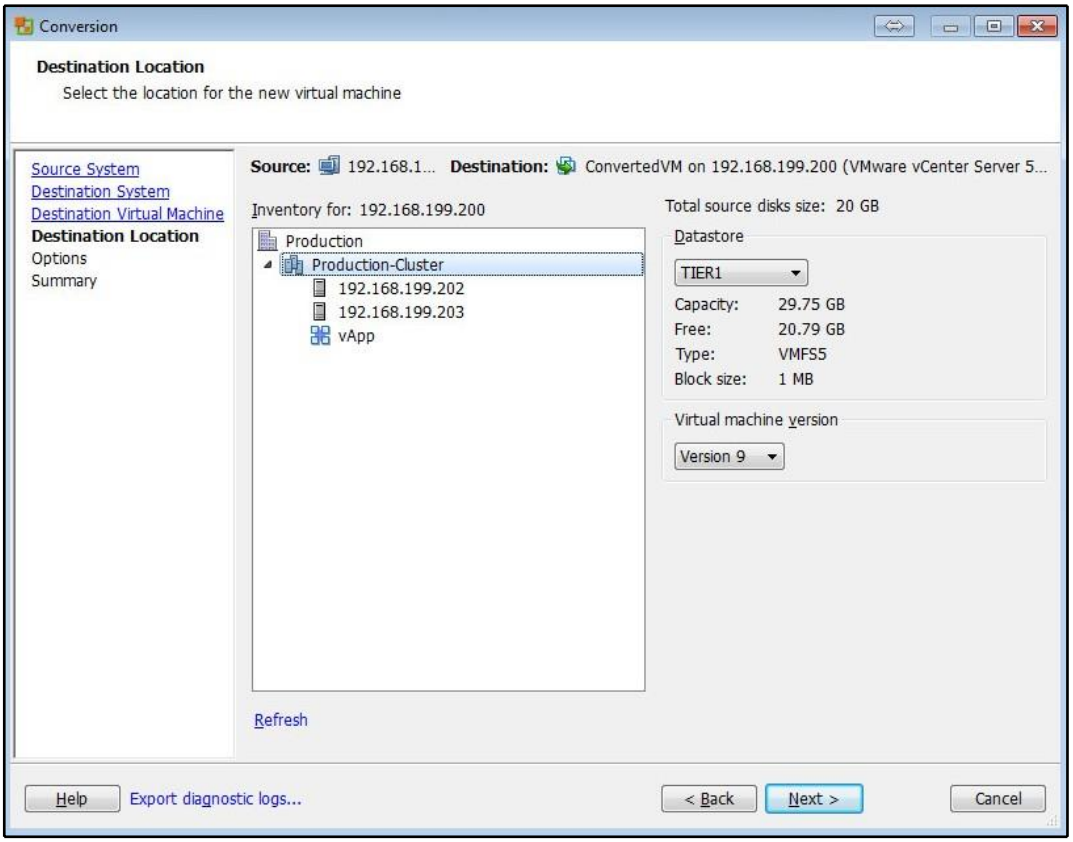
Choosing destination of vCenter Converter job

8. Information about the **Destination Virtual Machine**, such as the name and the virtual machine inventory placement, is then configured. The following screenshot shows the name and inventory placement for a physical-to-virtual conversion:



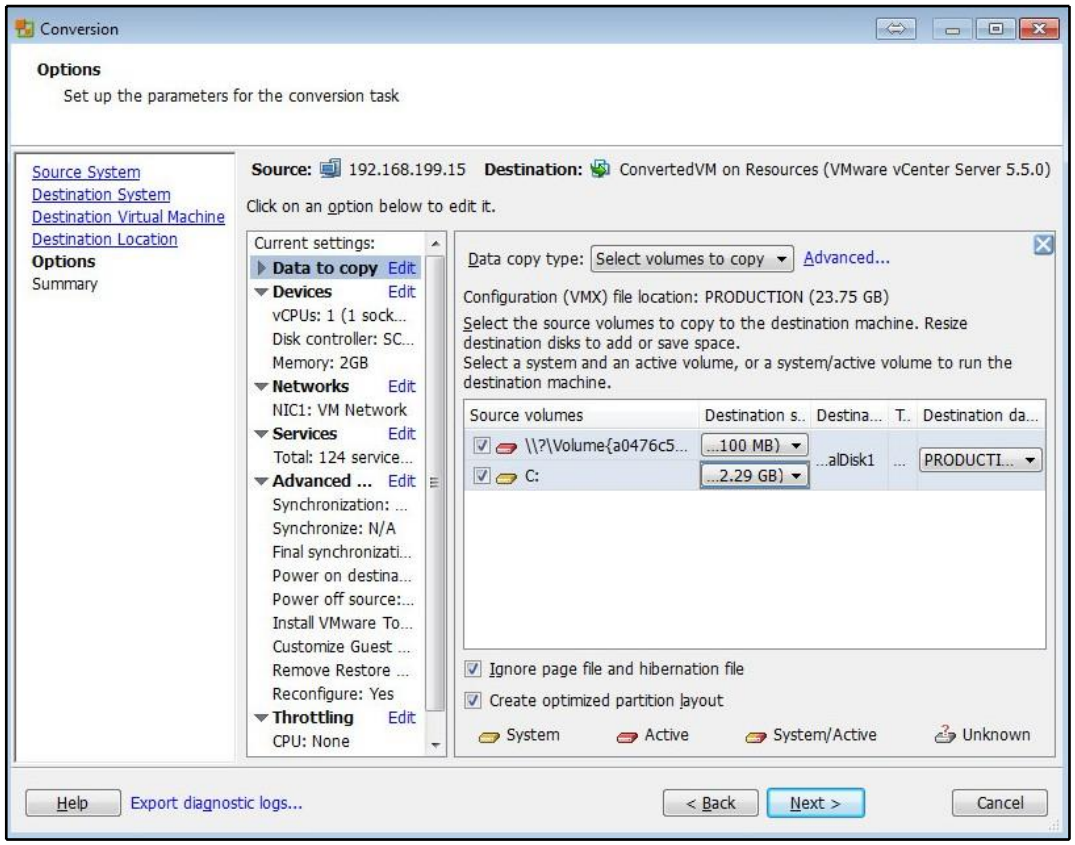
Configuring destination VM options in vCenter Converter

9. The **Destination Location** is then selected, as shown in the following screenshot. This location is the data center, cluster, or host that the converted machine will be deployed to. The datastore where the converted machine configuration file (`.vmx`) will reside and the virtual machine version to use are also configured here:



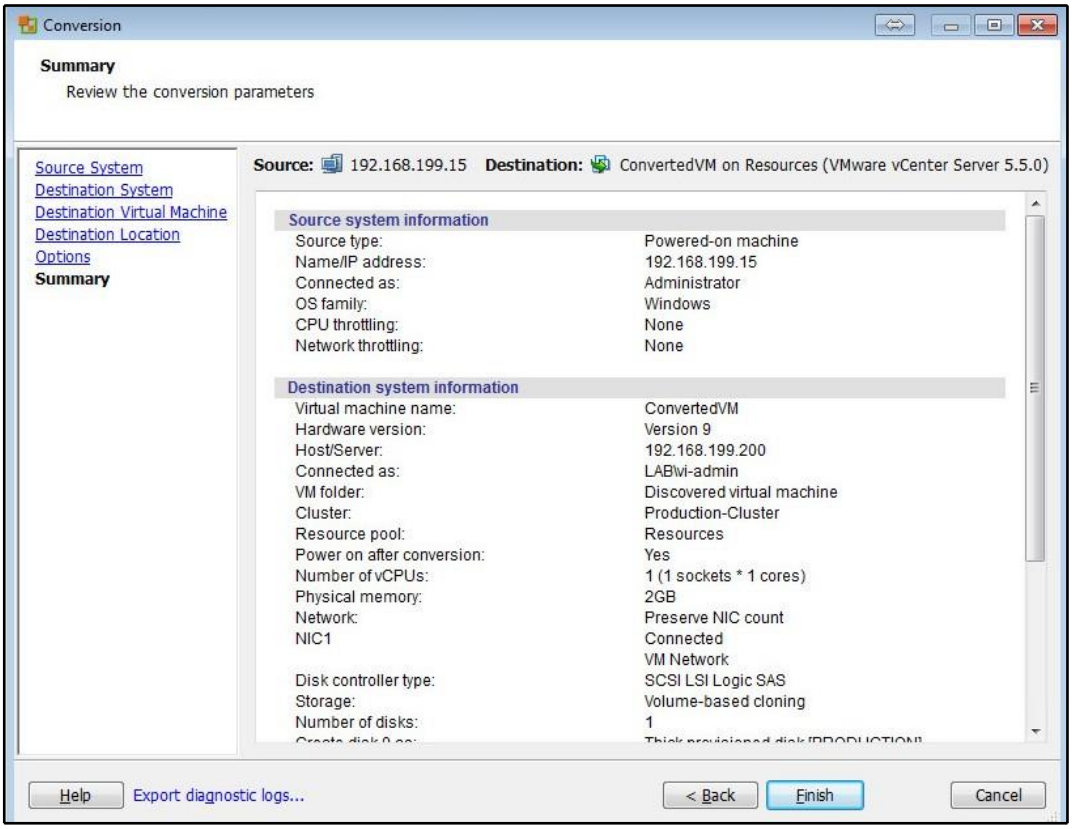
Choosing a destination for vCenter Converter job

10. A number of options can be configured for the converted machine, including what virtual machine network to connect to, what datastore to deploy the converted disk to (and in what format), and the device configuration. The following screenshot shows the **Options** configuration screen with the volume configuration for the machine that is being converted:



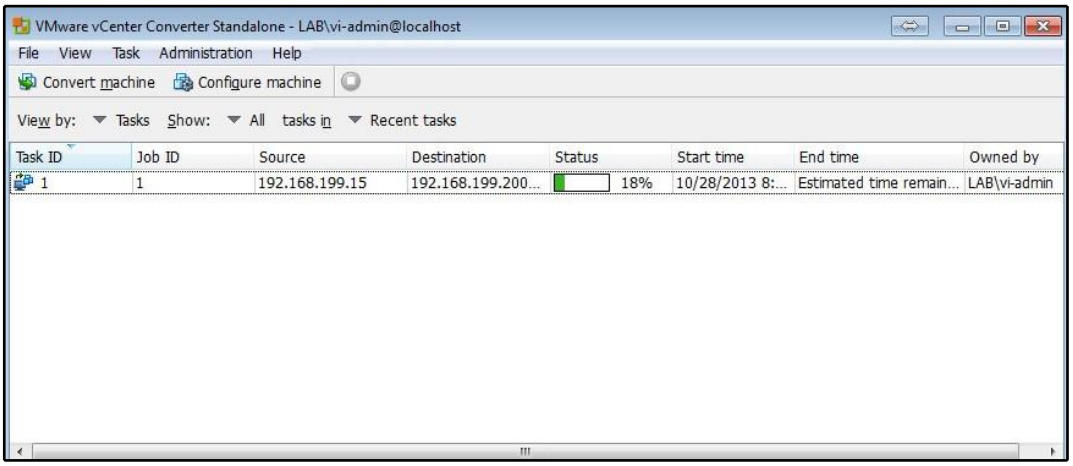
Options configuration screen in vCenter Converter

11. The **Summary** screen is then displayed, where the conversion options can be reviewed before starting the conversion process. An example **Summary** dialog is shown in the following screenshot:



Summary screen of a vCenter Converter job creation

12. Once the conversion starts, the progress can be monitored in the **VMware vCenter Converter Standalone** client, as shown in the following screenshot. The client allows for multiple conversions to be configured and run simultaneously:



Monitoring vCenter Converter job progress

How it works…

When a physical server is converted using vCenter Converter Standalone, the physical server is cloned into the virtual environment. This creates a copy of the physical server as a virtual machine containing a duplicate of the operating system, applications, and data from the physical server.

When the physical server is converted, new virtual hardware is presented to the virtual machine. The physical hardware that was associated with the virtual machines is no longer present, and references to it should be removed. In Windows, this is done by using the Device Manager.

During the physical-to-virtual conversion, the physical hardware is replaced with virtual hardware. During the conversion, references to the physical hardware and the associated drivers are not removed from the operating system. To remove non-present hardware from a Windows Server, set the environment variable `devmgr_show_nonpresent_devices` to `1`. This will enable non-present devices to be visible in the Device Manager.

An old but useful tool to help clean up the effects of **Physical-to-Virtual** (**P2V**) conversions, such as non-present devices, is called the VM Advanced ISO, by Kendrick Coleman. The ISO itself is a compilation of tools, but one in particular should be used to clean up a P2V conversion. You'll find it in the `P2v Clean-Up` directory of the ISO. It's a script called remove non-present devices. You can download the VM Advanced ISO from `http://kendrickcoleman.com/index.php/Tech-Blog/vm-advanced-iso-free-tools-for-advanced-tasks.html`.

The conversion can be verified by booting the new virtual machine while it is disconnected from the virtual switch and checking that the operating system and applications were converted correctly. Once verified, the physical server can be powered off or removed from the network, and the newly converted virtual machine can be connected to the network.

The migration of servers and applications can be time-consuming, and vCenter Converter Standalone provides a way to quickly convert physical servers into virtual machines.

Migrating servers into vSphere

It's not always possible to build new systems in a vSphere environment. Sometimes, it's necessary to migrate existing servers into the environment. The reasons for a migration might include a lack of time to build new systems or a lack of expertise to build new systems and migrate the data. Either way, we'll present some of the ways that you can plan and execute the migration of existing servers into a vSphere environment.

How to do it…

Planning server migrations can be done using the following steps:

1. Identify the existing platform of the server
2. Identify the possible methods for migration
3. Assess the feasibility and risks associated with each migration method
4. Execute the migration

How it works…

Some common source platforms for servers include physical Windows and Linux hosts, that is, operating systems that are installed on bare-metal servers. Some source systems are on other hypervisors, like Microsoft Hyper-V, while some systems already reside in a vSphere environment, but they need to be moved to a separate vSphere environment, perhaps one that is not owned or inherently trusted by the source. The application is an important component as well. Well-known applications that are easy migration candidates include Microsoft **Active Directory** (**AD**), SQL Server, or Exchange. All of these existing platforms are common sources for servers when moving to a new vSphere environment.

The method of migration is dependent on the source platform. Some common migration methods are listed as follows, along with their source platforms and associated drawbacks:

- Physical servers, Windows, or Linux:
    - **vCenter Converter**: This method is called a physical-to-virtual conversion, and it was discussed in the previous section. Drawbacks include downtime for the application during cutover. If the system is resource-intensive, downtime for the entire conversion process may be needed, which could take hours. Only some Linux distributions are supported.
    - **Third-party converter tools like PlateSpin Migrate**: These are very similar to vCenter Converter, but are licensed at a cost. The drawbacks are the same as those of vCenter Converter.
- Microsoft applications:
    - Often, the best way to migrate a server into a vSphere environment is to use native tools that are built into the system. Active Directory, for example, has a well-understood and well-supported replication engine that makes standing up a new Domain Controller relatively easy. The drawback to migrating with Active Directory replication is the need to add complexity to the AD environment and possible **Domain Name System** (**DNS**) changes, which could impact the entire environment.

- **SQL Server:** There are many ways to migrate an SQL database: backup and restore, mirroring, log shipping, **Microsoft Cluster Services** (**MSCS**), and **Always-On Availability Groups** (**AAGs**). The drawbacks of migrating with native SQL tools include varying degrees of downtime for the database.
- **Exchange**: Backup and restore, **Database Availability Group** (**DAG**), mailbox move; drawbacks include downtime and the length of time to migrate.

- Microsoft Hyper-V:
  - vCenter Converter is a good candidate for migrating VMs on Hyper-V. This is known as a **Virtual-to-Virtual** (**V2V**) conversion, and it has the same drawbacks as any other vCenter Converter migration.
  - The Zerto virtual replication software is mainly used for disaster recovery, but it's also an excellent migration tool. Downtime is minimal, and the setup is not difficult. The drawbacks include the cost, as it's a licensed product.

10
Deployment Workflow and Component Installation

VMware vSphere 6.7 is a sophisticated product with several components to install and set up. Understanding the correct sequence of tasks required to install and configure vSphere is the key to a successful deployment. The chapter starts by explaining the components of vSphere with the roles and services they provide. We will walk through the main aspects to consider for the preparation of a deployment plan for your environment, analyzing the criteria for hardware platform selection, storage, and network requirements.

The host deployment plan will then describe the different ways to install ESXi, including Auto Deploy, and other solutions for deploying the host. We'll also detail the deployment of **Platform Services Controller** (**PSC**) and **vCenter Server Appliance** (**vCSA**).

In this chapter, we will cover the following topics:

- vSphere components and workflow
- ESXi deployment plan
- ESXi installation
- vCenter Server components
- vCenter Server Appliance deployment
- vCSA **High Availability** (**HA**)

vSphere components and workflow

To provide services to the infrastructure, vSphere relies on two core components: the hypervisor, which is the virtualization layer for the complete environment, and vCenter Server, which centralizes the management of the ESXi hosts and allows administrators to automate and secure the virtual infrastructure. To complete the vSphere deployment, it is essential to know the interaction between ESXi and vCenter. Let's examine these two components to figure out their role:

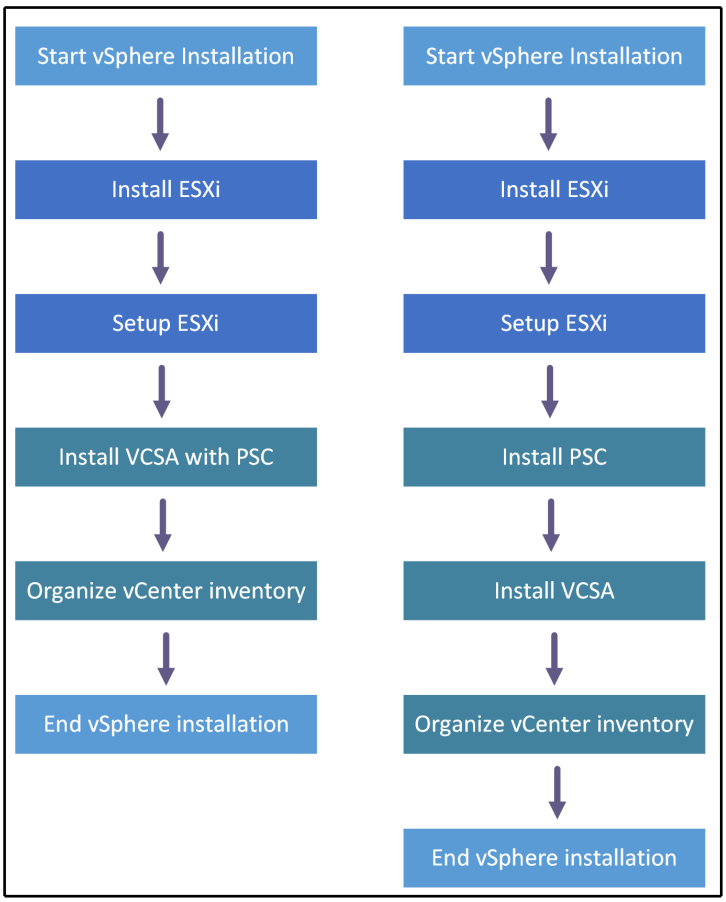
- The **ESXi hypervisor** is the platform on which VMs and virtual appliances run. Its primary function is to provide the resources to workloads regarding CPU and RAM, but it can also provide storage resources through vSAN. To manage the ESXi resources, ensuring performance and reliability, an additional component is necessary—vCenter Server.
- **vCenter Server** is a central management platform that manages ESXi hosts connected in a network and allows you to pool and manage the resources of multiple hosts. VMware vCenter Server can be installed in a virtual or physical machine with Windows Server, or deployed as a vCSA. The installation of vCSA can be launched from Windows, macOS, and Linux OS. Any host you plan on connecting to vCenter Server 6.7 must be running version 6.0 or above.

The **vCenter Server Appliance** (**vCSA**) is a preconfigured Linux-based (with VMware Photon OS) virtual machine that provides all the services required to run vCenter Server and its components. As compared to previous versions, vCSA 6.7 has the capability of providing all services as the Windows-based vCenter Server. **vSphere Update Manager** (**VUM**) is also completely integrated, and you no longer need to install a separate Windows server to host it. In a vSphere environment, vCenter Server is not an essential requirement to deploy the ESXi hosts, and VMs can run without it. However, advanced features available in vSphere can't be used without vCenter Server. You won't be able to provide services such as **vMotion**, **Distributed Resource Scheduler** (**DRS**), **HA**, **FT**, and **Update Manager**, to mention a few.

The services required to run vCenter Server and vCenter components are now bundled in the VMware PSC, a component introduced in version 6.0 of vSphere that provides common infrastructure services for VMware products.

For a correct installation sequence, the PSC (which will be discussed later) must always be installed before deploying the vCenter Server. Depending on the vSphere design, the PSC can be installed embedded in the vCenter Server or installed externally in the vCenter Server.

A correct VMware vSphere 6.7 deployment requires a specific installation sequence to avoid problems due to missing components or services and can be summarized as follows:



With a good design and following the correct workflow, the deployment procedure of vSphere 6.7 is straightforward and shouldn't raise problems. Let's have a look at the following points:

- **ESXi installation**: In this step, you have to verify whether the chosen hardware platform is included in the HCL, and determine what installation method to use and which destination to use to boot the hypervisor.
- **ESXi setup**: This step involves the initial configuration of the ESXi server, especially its management network.

- **vCenter Server and PSC deployment**: You should identify what deployment model best fits in your environment for vCenter Server and the PSC. The vCenter Server can be deployed with an embedded or an external PSC depending on the design (multiple vCenter Server instances, for example). vCenter Server and the PSC can be installed on a Windows machine (physical or virtual) or a vCSA.
- **Connect to vCenter Server**: Use the integrated vSphere Web Client to complete the configuration of the vCenter Server.

When the required steps are clear, let's start the vSphere deployment by examining the first core component of the infrastructure—the ESXi hypervisor.

ESXi deployment plan

A successful vSphere deployment requires an appropriate plan to avoid problems of incompatibility, performance, and instability with the commitment of remaining within the available budget.

Three main areas should be considered when planning a vSphere deployment:

- Choosing the hardware platform
- Identification of the storage architecture and protocols (NFS, iSCSI, FC, or FCoE)
- Network configuration (number of NICs, FC adapters, 1 GbE, or 10 GbE NICs)

Choosing the hardware platform

An important decision to take when planning an ESXi deployment is the choice of the hardware platform of the server. ESXi doesn't support all the hardware available on the market (storage controllers, NICs, and so on) and has some restrictions that can prevent the successful installation of the hypervisor.

Only tested and supported hardware ensures that your ESXi can be installed without any problem and can operate as expected. Before purchasing the hardware for your server, it is strongly recommended you verify whether ESXi supports the chosen hardware platform.

To check for hardware compatibility, you can refer to the VMware Compatibility Guide available at `https://www.vmware.com/resources/compatibility/`. The list of tested hardware is large, and you can find the supported hardware from the leading manufacturers, such as HP, Dell, IBM, and Cisco.

When new hardware is released and certified for compatibility, the list of supported vendors is updated accordingly.

Choosing the right server for your installation is not an easy task, especially if your environment grows quickly and the business requirements change frequently. Capability, scalability, availability, and support are the elements of the server you need to evaluate carefully to be sure the final choice fits in the available budget without affecting the global design.

In some scenarios, it is better to have more, smaller servers in a cluster to provide the required resources than a few big servers.

There are several considerations you need to think about:

- Fewer servers with more resources are usually cheaper compared to the same amount of resources distributed among more smaller servers (you always need to buy a chassis, a motherboard, and power supplies for every server).
- With more smaller servers you can have multiple fault domains. For example, if you have 10 ESXi hypervisors in four racks, when one rack (fault domain) becomes inaccessible, you still have other fault domains compared to 15 powerful servers in a single rack.
- When your ESXi server becomes unavailable, you lose part of the computing power. If you have 10 large servers, you will lose 10% of the total capacity if you encounter downtime. If you have 50 smaller servers, you will lose only 2%.
- With more servers, you will need adequate network infrastructure concerning the physical 1/10/40/100 GbE ports.

The challenge is to find a server that provides the number of resources that meets the requirements but at the same time supports enough expansion (scalability) if the demand for resources grows.

Another factor you should consider is the expected performance of the server. The default hardware BIOS settings of the chosen hardware do not always ensure the best performance. To optimize performance, you should check some of the following in your server's BIOS settings:

- Hyperthreading should be enabled for processors that support it.
- Enable turbo boost if your processors support it.
- In NUMA-capable systems, disabling node interleaving (leaving NUMA enabled) will give you the best performance.
- Hardware-assisted virtualization features, such as VT-x, AMD-V, EPT, and RVI, should be enabled.
- Consider whether you should disable any devices you won't be using from the BIOS (Serial/Parallel ports, unused PCIe cards, and so on).
- For power management, you can choose to enable max performance or leave the control in ESXi with OS Controlled mode.

Identification of the storage architecture

Choosing a suitable storage solution is another piece of the deployment plan. You should consider what protocols will be used and their direct dependencies. For instance, a **Fibre Channel** (**FC**) storage device requires FC adapters to be installed on the server. vSphere supports software and hardware initiators (also known as **host bus adapters** (**HBAs**) or **converged network adapters** (**CNA**)) that add flexibility to your storage architecture design.

An ESXi host may use multiple storage protocols in the same installation to support the design requirements. It is not unusual to see different ESXi installations with FC and NFS storage devices connected at the same host and, in some scenarios, also with the addition of an iSCSI storage.

Defining the network configuration

For a successful deployment plan, you should consider the impact on your environment and how the deployment will integrate with the existing network infrastructure. This is another crucial point to keep in mind, because it is strictly related to the hardware chosen for the server and the storage protocols used.

ESXi generates network traffic that must be controlled and sized to properly manage advanced features such as vMotion, FT, and VM traffic without congesting the network. The question is, *how many NICs should I use?*

The number of NICs supported by the server can profoundly influence the network design and, consequently, the overall host performance. If the server has only four slots available to accommodate the adapters and FC storage is used, the server needs to be equipped with at least two FC adapters to provide FT, taking precious slots intended for additional NICs.

Depending on the design of your ESXi server, general guidelines you might consider when defining the number of NICs to use are the following:

- **ESXi management network**: One NIC is required; two would be better for redundancy.
- **vMotion**: At least one NIC and, due to the amount of data involved during a vMotion process, a **Gigabit Ethernet** (**GbE**) must be used. More NICs can provide more bandwidth, with the right configuration.
- **vSphere FT**: It requires at least 1 GbE NIC but, depending on how many vCPU and FT-enabled VMs are configured, a 10 GbE NIC could be a better choice. A second NIC is recommended for redundancy.
- **Storage**: Except for FC, which uses different adapters, NFS or iSCSI storage protocols need at least a 1 GbE or, better, 10 GbE. Also, for this configuration, more NICs are recommended for redundancy and performance.
- **VM traffic**: To better distribute and balance the load, two or more GbE NICs are recommended.

A general recommendation is to have always two NICs available for specific traffic, although multiple traffic types can be combined into the single NIC. By using two (or more) NICs you can eliminate the possible network outage regarding switch failure, cable to cut or similar situation.

I tend to use 2x GbE NICs for management and 2x 10 GbE NICs for everything else and to separate traffic using Network I/O Control or at least using VLANs if the license does not allow use NIOC.

When you have defined the server and the number of NICs you need for your design, the ESXi installation plan raises a new question—how should I install ESXi?

ESXi installation

Once you have defined the hardware platform and the storage and network setup, you are ready to deploy the ESXi host. The installation is pretty simple and takes only a few minutes.

The latest release of vSphere made an essential enhancement regarding security, introducing a new feature for the hypervisor—Secure Boot. Secure Boot is a solution that ensures that only the trusted EFI firmware loads code before the OS boots. The trust is given by the UEFI firmware that validates the digitally signed ESXi kernel against a digital certificate stored in the UEFI firmware.

Once you have defined the design of the virtual infrastructure, you should evaluate which installation option is suitable for your environment. vSphere 6.7 offers three options to deploy ESXi:

- **Interactive**: Manually providing answers to installation options
- **Unattended**: Using installation scripts
- **Automated**: Using the vSphere Auto Deploy feature

The deployment method to adopt depends on the size of your environment and on the number of hosts to install. Interactive installation is definitively the most straightforward procedure you can use but requires more time if you have several hosts to deploy. Automated installation is more complicated to implement but for large environments is always the preferred choice.

There are, of course, additional steps to follow for unattended or automated installation, but in the end, it will save you a lot of time.

Once you have defined how to install ESXi, you should ask yourself the following question: *where should I install ESXi?* Let's examine the available options you may consider.

Where should I install ESXi?

Before installing ESXi, you need to decide where to store the ESXi files. A local disk, SD card, SAN (LUN), FC, or USB device are all possible destinations you can use for ESXi, but what solution is the best and what you should use is very hard to say. The choice to make depends on your infrastructure design and network configuration, and the installed devices in your target machine.

You can use a remote device available through your SAN (for hardware-based HBAs such as fiber channel, fiber channel over Ethernet, or iSCSI) but you can use this method with software-based initiators for iSCSI and FCoE. An extra configuration is required to set up LUNs and zoning . Anyhow, the use of SAN LUN creates a dependency on an external storage array that, in the case of failure, makes the ESXi unusable, but your storage array and SAN need to be highly available anyway.

From my perspective, using a SAN device for booting is always preferable if your hardware infrastructure allows it since you do not need additional local disks installed in your server.

Using local disks as the destination for the ESXi files is a solution that, until a few years ago, was popular in most ESXi installations because it is a cost-effective solution and doesn't require any extra configuration. If the local hard disk is your choice, I strongly recommend configuring **RAID 1** to provide fault tolerance. There is no need to invest in SSDs as booting devices, because there is no benefit from using them at all. The smallest available disk from your vendor will do the job.

A valid alternative to HDDs is the use of SD cards which offer better performance and, compared to years ago, are more significant and more cost-effective. To use SD cards, the target server needs to be equipped with **Secure Digital** (**SD**) bays, but if your server has only one bay available, it won't be able to provide fault tolerance. Luckily, certain hardware manufacturers, such as Dell and HP, provide servers equipped with double bays for SD cards you can mirror, like you would an HDD, and fit perfectly in this installation method.

The downside of using SD cards is the requirement of some additional configuration. The scratch partition of ESXi needs to be placed in persistent storage (VMFS or NFS volumes attached to the server) to store `vm-support` output, which you need when you create a support bundle.

The USB stick is another possible option you can use for ESXi installation. It is the most economical destination device, but for production servers, I don't recommend its use since you don't have any redundancy in case of failure.

As with SD cards, for USB sticks, no log files will be stored locally (a scratch partition needs to be configured). Although a 1 GB USB or SD device suffices for a minimal installation, it is recommended you use a 4 GB or larger device.

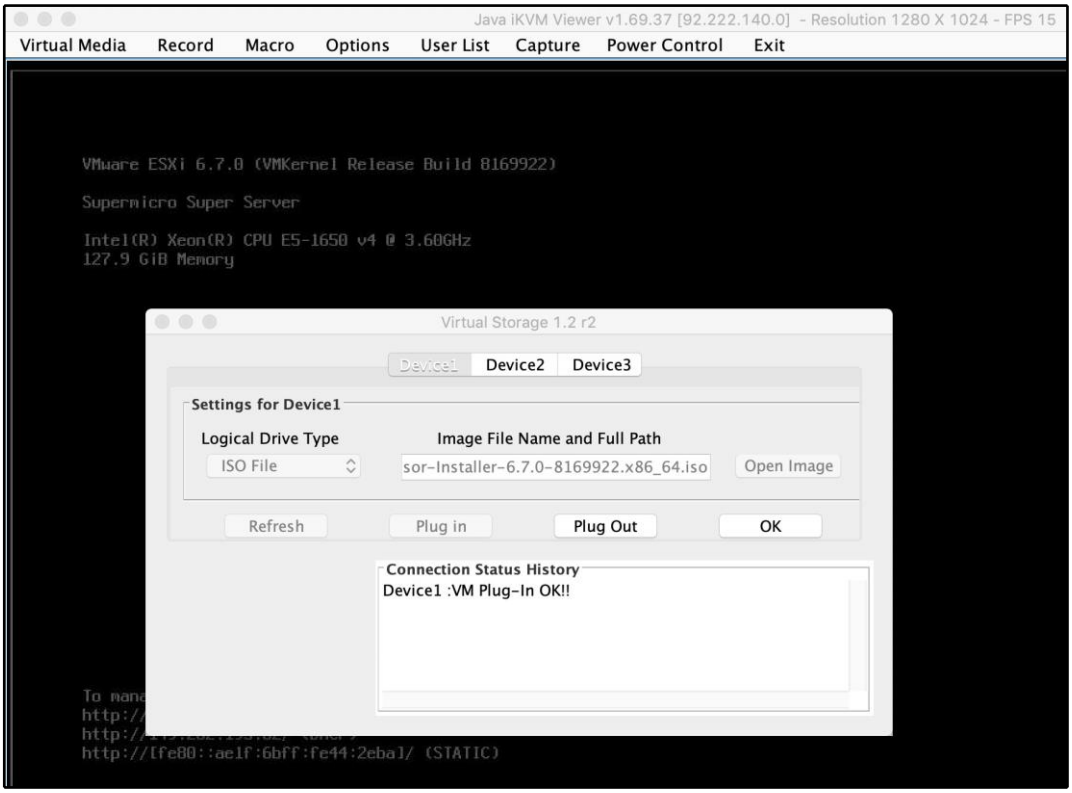
Did you notice that fault tolerance is a recurrent caveat for devices? Is there any reason for that? Yes, of course. Let's talk briefly about the importance of having fault-tolerant components. *What happens if you use just one device for your ESXi installation and suddenly the device fails?* The ESXi stops working, stops providing its resources, and in a situation of inadequate infrastructure design, the network services may be no longer available to users. For this reason, a good design for ESXi should consider the use of two devices configured in mirror RAID 1 to provide fault tolerance and performance.

Preparing for deployment

When the destination for your ESXi has been chosen, you should decide what method to use for the ESXi deployment. Before proceeding with the installation, you can download the installation files at `https://my.vmware.com/web/vmware/downloads/`. The installation files are typically provided in ISO format to be quickly burned to a physical CD/DVD or mounted to a server.

The installation using a physical CD/DVD can be considered old-fashioned and time-consuming, but to install the ESXi, you have also the option of using an USB flash drive, through the network using the **Preboot Execution Environmen**t (**PXE**), or mounting the ISO installation file (virtual CD) if your server is equipped with the remote management tool (iLO, iDRAC, IPMI, or similar). Perhaps the USB key is the fastest solution to use if your server doesn't have any integrated remote management tool since you need to create a bootable USB key. Tools such as **UNetbootin** or **Rufus** can do that using the ISO file without burning any CDs.

Some manufacturers, such as HP, Dell, and Super Micro, provide servers with an integrated remote management tool that allows the use of the virtual CD feature, which remains an excellent way to install your ESXi without the need to burn the ISO (you can sit at your desk without getting cold inside the data center):



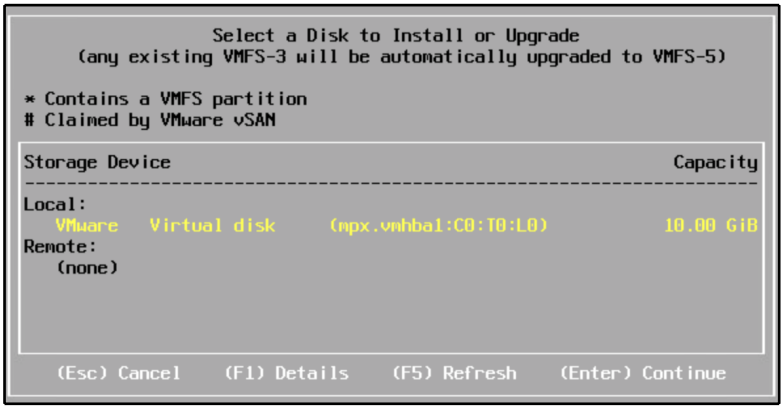
Let's examine the three possible installation options.

Interactive installation

Interactive installation is straightforward, because the procedure makes use of a comfortable and intuitive interface that guides the user during the entire process. The installer is booted from a CD/ DVD, from a bootable USB device, or by PXE booting the installer from a location on the network. The interactive installation method best applies to small environments where the number of ESXi hosts to install is limited. You can install ESXi in a few minutes directly launching the installer from the installation media, and no scripts or dedicated network configurations are required to complete the procedure. If you need to install a few ESXi hosts, this is definitively the fastest and most straightforward option.

Depending on the ESXi installer media used (CD/DVD, USB flash drive, or PXE), remember to set the BIOS server accordingly to configure the correct boot sequence. Perform the following steps to proceed with an interactive installation:

1. Insert the installation medium (CD/DVD, USB flash drive) and power on the server. When the server boots, the installer will display the Boot Menu window.
2. Select the ESXi installer and press *Enter*. The system loads the ESXi installer and displays the welcome screen. Press *Enter* to continue.
3. Accept the **End User License Agreement** (**EULA**) by pressing *F11* and continue with the installation.
4. The next screen displays the available devices on which to install the ESXi, divided into local devices and remote devices (see the following screenshot). Select the desired destination and press *Enter*. Since the disk order shown in the list is determined by the BIOS, make sure the selected device is operative. To get the details of any previous ESXi installation and what VMFS datastore is detected, press *F1*.
5. In the disk selection window, SATA disks, SD cards, SATADOM, and USB flash drives are listed as local devices, while SAN LUNs and SAS devices are listed as remote:



6. If the selected device contains a previous ESXi installation or a VMFS datastore, you have three clear choices to select:

   - **Upgrade ESXi, preserve VMFS datastore**
   - **Install ESXi, preserve VMFS datastore**
   - **Install ESXi, overwrite VMFS datastore**

7. The keyboard layout selection is the next screen. Select your language then press *Enter*.
8. Enter the root password twice and press *Enter*. For security reasons, keep the password in a safe place.
9. At the confirm install screen, press *F11* to proceed with the installation. The procedure only takes a few minutes and begins re-partitioning the disk and installing the host in the selected device.
10. After the installer completes, remove the installation CD/DVD or USB flash drive, and press *Enter* to reboot the host.
11. Once the host has rebooted, the procedure is complete. For new installations, or if an existing VMFS datastore is overwritten, VFAT scratch and VMFS partitions are created on the host disk (only if the destination device is not an SD card or USB stick).

By default, the ESXi is configured to obtain an IP address from a DHCP server used for its management. If your network doesn't have any DHCP server installed, the ESXi won't be able to obtain an IP address, and you will need to configure it manually.

Unattended installation

While interactive installation is straightforward, you need to repeat the same steps for each server to install. If the number of hosts to install increases dramatically, the interactive installation method may not be the most suitable choice. The installation process can be automated using a script to provide an efficient way to deploy multiple hosts.

ESXi supports the use of an installation script to automate the installation process and can be useful if you want to have a consistent configuration for all hosts. Using an installation script, you can quickly deploy multiple instances of ESXi, creating unattended installation routines. These scripts can be saved on a USB flash drive or in a network location accessible through NFS, HTTP, HTTPS, or FTP.

The following table indicates some common boot options for unattended ESXi installation. For a complete list of supported boot options, refer to the *vSphere Installation and Setup Guide* available on the VMware website at `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.esxi.upgrade.doc/GUID-61A14EBB-5CF3-43EE-87EF-DB8EC6D83698.html`:

Boot option	Description
`BOOTIF=hwtype-MAC address`	Similar to the `netdevice` option, except in the PXELINUX format as described in the IPAPPEND option under SYSLINUX at `syslinux.zytor.com`.
`gateway=ip address`	Sets the default gateway to be used for downloading the installation script and installation media.
`ip=ip address`	Used to set a static IP address to be used for downloading the installation script and the installation media.
`ks=cdrom:/path`	Specifies that the path of the installation script, which resides on the CD in the CD-ROM drive. The path of the script must be written in uppercase characters (for example, `ks=cdrom:/KS_CUST.CFG`).
`ks=file://path`	Performs a scripted installation with the script at `path`.
`ks=protocol://serverpath`	Specifies that the script is located on the network at the given URL. Supported protocol can be HTTP, HTTPS, FTP, or NFS (for example, `ks=nfs://host/porturl-path`).
`ks=usb`	Indicates that the installation script is located in an attached USB drive. `ks.cfg` must be in the root directory of the drive. Only FAT16 and FAT32 are supported. If multiple USB flash drives are attached, the system searches until the `ks.cfg` file is found.
`ks=usb:/path`	Specifies the path of the installation script that resides on the USB (for example, `ks=usb:/ks.cfg`).
`ksdevice=device`	Tries to use a network adapter device when looking for an installation script and installation media. If the script has to be retrieved over the network, the first discovered plugged-in NIC is used if one is not specified.

`nameserver=ip address`	Specifies a domain name server to be used for downloading the installation script and installation media.
`netdevice=device`	Tries to use a network adapter device when looking for an installation script and installation media. The device can be specified as `vmnicXX` name. If the script has to be retrieved over the network, the first discovered plugged-in NIC is used if not specified.
`netmask=subnet mask`	Specifies the subnet mask for the network interface that downloads the installation script and the installation medium.
`vlanid=vlanid`	Used to specify the VLAN for the network card.

Common boot options for unattended ESXi installation

The installation script is a text file often named `ks.cfg` that contains supported commands useful to provide the required installation options to the ESXi installer. In the installation medium, VMware includes a default installation script that can be used as a reference to perform an unattended ESXi installation to the first detected disk. You can use this script if it is suitable for your ESXi installation.

The default sample script is as follows:

```
#
# Sample scripted installation file
#
# Accept the VMware End User License Agreement
vmaccepteula
# Set the root password for the DCUI and Tech Support Mode
rootpw mypassword
# Install on the first local disk available on machine
install --firstdisk --overwritevmfs
# Set the network to DHCP on the first network adapter
network --bootproto=dhcp --device=vmnic0
# A sample post-install script
%post --interpreter=python --ignorefailure=true
import time
stampFile = open('/finished.stamp', mode='w')
stampFile.write( time.asctime() )
```

To create a custom installation script or modify the default script, you should use the supported commands available.

A complete list of supported commands to use with installation scripts can be found in the **vSphere Installation and Setup Guide**: `https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-esxi-67-installation-setup-guide.pdf`.

To configure an unattended installation booting from a USB stick, perform the following steps:

1. Navigate to the installation media and edit the `boot.cfg` file. Replace `kernelopt=runweasel` with `kernelopt=runweasel ks=usb:/ks.cfg`. This allows the system to automatically use the script located on the USB drive. Make sure you use an editor that can handle UNIX encoding.
2. Create a `ks.cfg` file in the root directory of the USB device that the installer will use for the unattended installation. Edit the file and create the script. You can use the following simple script as an example:

```
vmaccepteula
rootpw mypassword
install --firstdisk –overwritevmfs
keyboard English
network --bootproto=dhcp --device=vmnic0
reboot
```

3. Save and close the file. Plug in the USB stick and power on the server.
4. To manually run the installer script when the ESXi installer window appears, press *Shift* + *O* to edit boot options. At the `runweasel` command line, type `ks=usb:/ks.cfg`. To specify the path to an installation script, you may also use the `ks=http://ip_address/kickstart/ks.cfg` command, where the IP address refers to the machine where the script resides.
5. The system will boot from the USB stick and do an unattended installation.

The main benefit of using unattended installations for ESXi is not only that it speeds up the installation process, but it also ensures a consistent configuration of all ESXi hosts.

There are certain caveats when using unattended installation such as an IP configuration. Generally, you do not want to use a dynamically assigned IP address, but instead use a statically configured address. To do that without any additional modifications, you would need to have individual config files for each ESXi server you want to configure and select them during the boot process: `ks=http://ip_address/kickstarts/ks_ESXi-36.cfg`.

You can automate the whole procedure using some scripts.

The first requirement is to have MAC address mapping to the individual ESXi servers since the DHCP server knows the mapping between MAC address and the IP address.

Once your ESXi servers connect to the web server where the script resides over IP, you will know which ESXi server it is thanks to this mapping. Now, based on this information, you can dynamically create the correct `kickstart` file with appropriate IP address for the management interface or other variables individual for each ESXi server, such as DNS settings.

You can, of course, use the more convenient method, Auto Deploy, which will do all the work for you, but this feature is only available in Enterprise editions of vSphere.

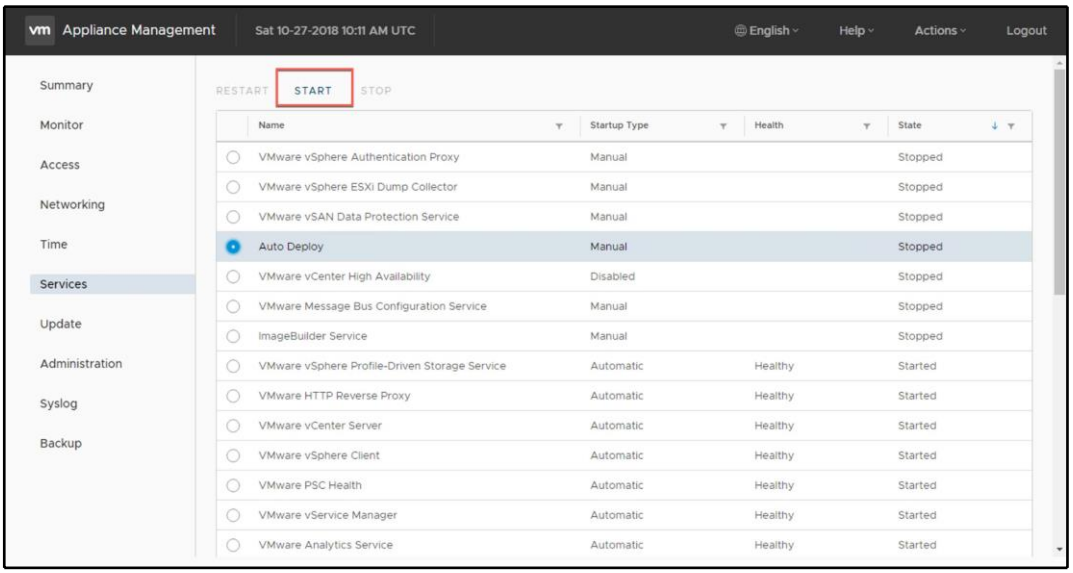
Auto Deploy installation

Auto Deploy installation is a way to PXE-boot your ESXi hosts from a central Auto Deploy server. This method is based on the use of master images with some set of rules to deploy ESXi with the desired specifications. Auto Deploy can also be used with the vSphere Host Profile feature to customize all ESXi hosts, ensuring a consistent configuration within the infrastructure. In a large environment, setting up the vSphere Auto Deploy feature to handle ESXi installations is the most efficient and suitable method to use.

Auto Deploy relies on several components, and the configuration required is more complicated. A vCenter Server must be already present in the vSphere infrastructure to provide the Auto Deploy feature. You also need a DHCP server and a **Trivial File Transfer Protocol** (**TFTP**).

The Auto Deploy feature in VMware vSphere 6.7 introduces a new graphical user interface for managing ESXi images and deployment rules that reduces complexity and helps users during the configuration. PowerCLI is still available and has been enhanced with a new script bundle that allows administrators to add a post-deployment script once all the configurations have been applied to a stateless ESXi host.

The Auto Deploy feature is installed with the vCSA but by default is disabled. To use this functionality, you need to enable the service:



Please note that from the web management of the vCSA you can only start the service, but you can set it to automatic startup.

To start the services and set the automatic startup, you need to use vSphere Web Client (not the HTML5 client):

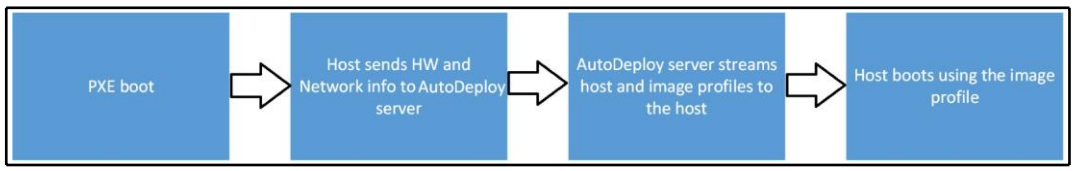
1. Log in to the vSphere Web Client as administrator.
2. Go to **Administration** | **System Configuration** | **Services**.
3. Right-click on **ImageBuilder** and **AutoDeploy Service** and select **Edit Startup Type**. Choose **Automatic** and click **OK**.

4. Select **ImageBuilder** and **AutoDeploy Service** and choose **Start**.
5. Log out of the vSphere Web Client and log in once again. The Auto Deploy icon should now be visible.

Before digging into the installation procedure, let's see how vSphere Auto Deploy works and the configuration of the required components (DHCP, TFTP).

How Auto Deploy works

To take advantage of this deployment method, it is essential to understand how Auto Deploy works and what steps and services are involved during the ESXi deployment process. Different components interact with vSphere Auto Deploy when a fresh host boots:



The ESXi booting process through the Auto Deploy feature involves the following steps:

1. When the server first boots, the host starts a PXE boot sequence. The DHCP server provides an IP address giving instructions to the host on how to contact the TFTP server (DHCP configuration will be discussed later).
2. When the host establishes the connection with the TFTP server, it downloads the iPXE file (executable boot loader), named `undionly.kpxe.vmw-hardwired` for legacy BIOS or `snponly64.efi.vmw-hardwired` for UEFI BIOS, and a configuration file.
3. During the iPXE execution, the host makes an HTTP boot request to the vSphere Auto Deploy server (this info is stored in the iPXE configuration file) to get hardware and network information.
4. The vSphere Auto Deploy server streams the required components as the image profile to the hosts based on the defined rules.
5. Host boots using the image profile assigned by the deploy rule. If a host profile has been assigned as well, it is applied to the host.
6. The host is added to the same vCenter with which Auto Deploy is registered. If any rule does not specify the inventory location, the host is added to the first data center displayed in the vSphere Web Client UI.

Configuring DHCP

To support vSphere Auto Deploy, the DHCP server has to be configured accordingly. First, we have to define basic settings to configure a DHCP scope including the default gateway. If you want to assign a specific IP address to the host to be better identified, you can use DHCP reservation to accomplish this.

When the necessary settings are ready, you need to specify two additional options:

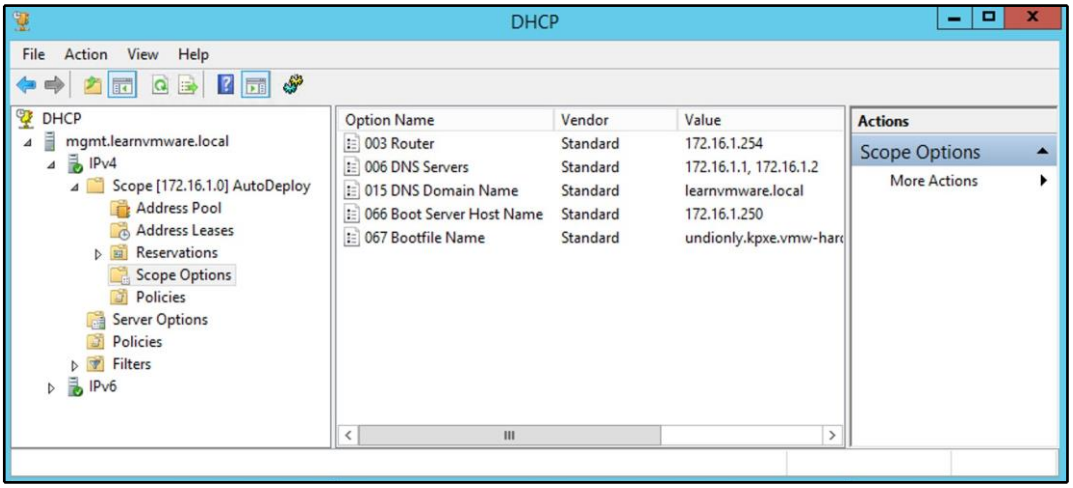
- **Option 66**: In this option, you should specify the **Boot Server Host Name** to be used by the system.
- **Option 67**: The **Bootfile Name** must be specified. The filename `undionly.kpxe.vmw-hardwired` can be found in the Auto Deploy configuration tab in the **BIOS DHCP File Name** field.

If VLANs are used in your vSphere Auto Deploy environment, make sure you set up end-to-end networking correctly because, during the host PXE booting, the firmware driver has to tag the frames with proper VLAN IDs. Changes must be set manually in the UEFI/BIOS interface. If you are running a Windows-based environment, the easiest way is to install a DHCP server role.

Once the DHCP server role is installed, you need to configure desired options.

At the least, you need to configure your new DHCP scope (IP Address Range for DHCP clients) and Option 66 and Option 67.

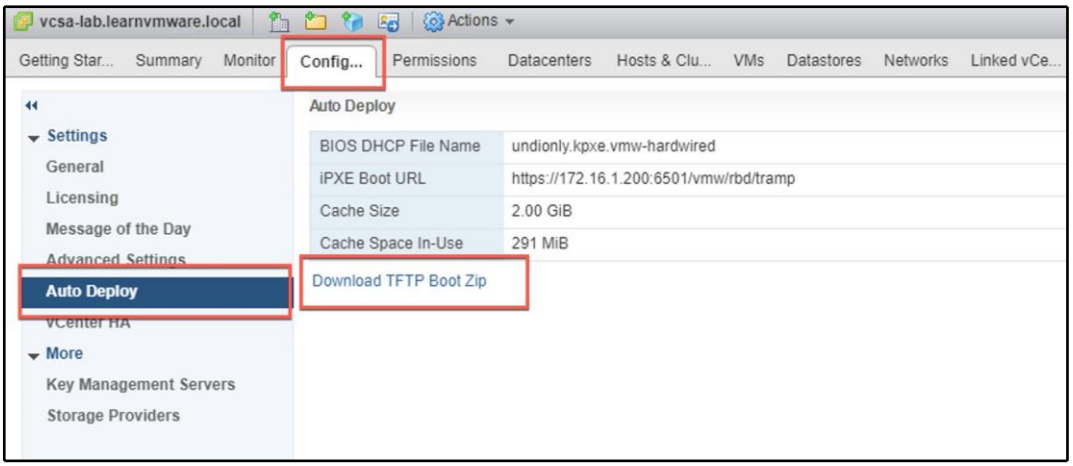
If the vCenter Server is not on the same L2 network, you also need to configure Option 003 – Default Gateway for your DHCP client, and preferably DNS servers using Option 006:



Configuring TFTP

vCenter Server does not include TFTP server, so you need to install third-party TFTP server as well and store the boot files from your vCenter Server.

To access those files, navigate to the **Configure** tab of your vCenter Server, select **Auto Deploy**, and **Download TFTP Boot Zip:**



As the TFTP server, you can use some free tools such as **Pumpkin TFTP** server available at `http://kin.klever.net/pumpkin/binaries`, **SolarWind TFTP Server** available at `http://www.solarwinds.com/free-tools/free-tftp-server`, or **Tftpd32** available at `http://tftpd32.jounin.net/`.

When you boot a new server, it gets the IP address from the DHCP and connects to the TFTP server through Option 66 and Option 67 specified during the DHCP configuration. Your new host will be available in the **Discovered Hosts** tab of **Auto Deploy configuration**, you will be able to see the host with an assigned IP address, or it will be automatically added to the inventory based on your deployment rule.

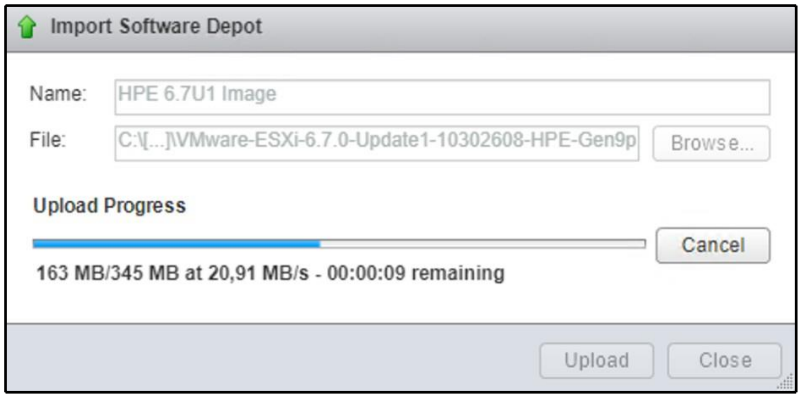
Creating an image profile

Image profiles are a set of **vSphere Installation Bundles** (**VIBs**), a collection of files packaged into a single archive to facilitate distribution and used to boot the ESXi hosts. Image profiles are built and made available in public depots by VMware and VMware partners. You can create custom image profiles, usually by cloning an existing image profile and then adding the required software packages VIBs to the image created.

To create an image profile, you should add at least one software depot, but you can add multiple software depots. A software depot can be a structure of folders and files stored on an HTTP server (online depot) or, more commonly, in the form of a ZIP file (offline depot). The software depot contains the image profiles and software packages VIBs that are used to run ESXi.

You can either use the official VMware depot which contains all VMware ESXi images or you can create your own **Software Depots** and upload your existing **Image Profile:**

If you want to use the official VMware online depot, all you need is to create a new online depot using the following URL: `https://hostupdate.vmware.com/software/VUM/PRODUCTION/main/vmw-depot-index.xml`.



For custom image profiles such as the official profiles from hardware vendors you can follow this steps:

1. Go to the **Auto Deploy** configuration page then select the **Software Depots** tab
2. Click on the green arrow to import a software depot
3. Type a name in the **Name** field and select the file to use an image then click **Upload**

Once the depot is successfully added, you can browse all available image profiles associated with the depot.

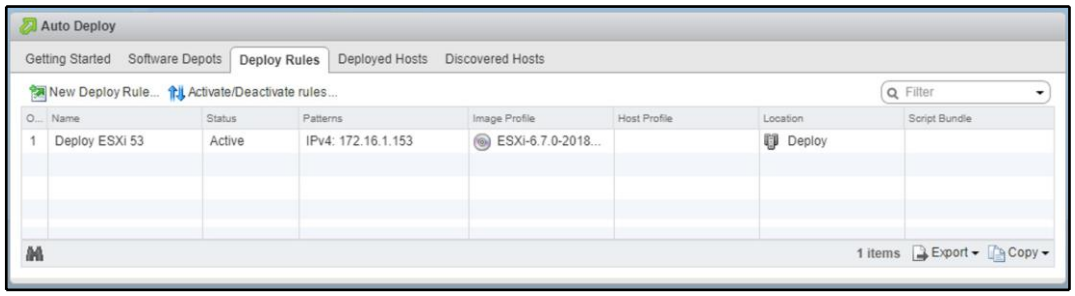
An image profile doesn't contain any configuration (virtual switch, security settings, and so on) and you should use the vSphere Host Profile feature to store the desired ESXi configuration in vCenter Server providing the parameters to the host to provision. If syslog is not configured in the host profile, logs are lost every time the host is rebooted since they are stored in memory.

Creating deployment rules

Deployment rules are used to link the image profiles to hosts and VIBs defined in a specific image profile. To make an image profile available to hosts, VIBs are copied to the Auto Deploy server to be accessible from hosts.

To start provisioning hosts through Auto Deploy, you should define a deployment rule to apply. To create a new deployment rule, proceed with these steps:

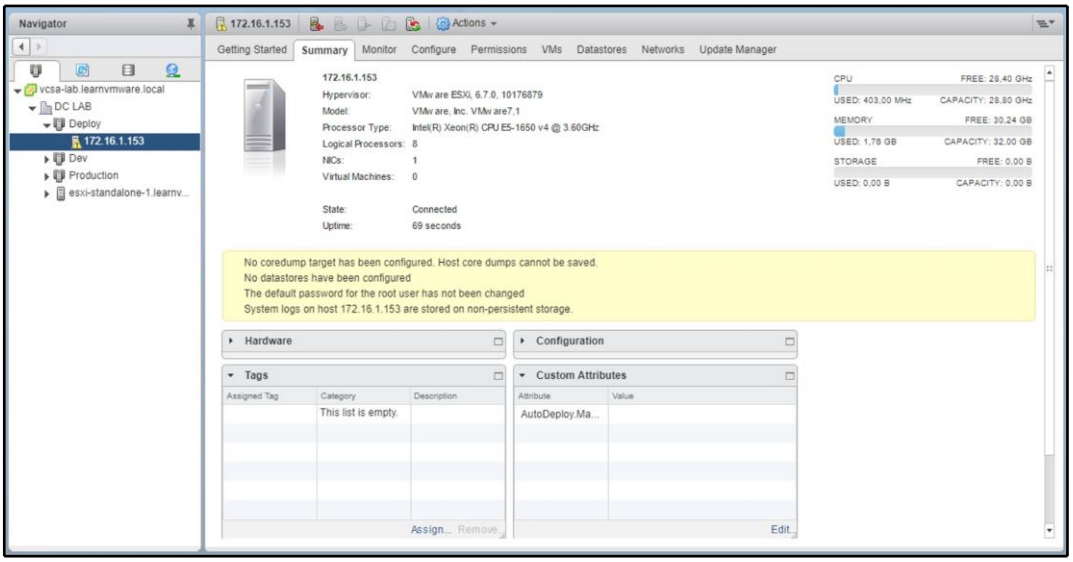
1. Select the **Deploy Rules** tab and click on the **New Deploy Rule** icon. Enter a name in the Name field and specify to which hosts the rule should apply. If you want to apply the rule only to specific hosts, select one or more patterns that the hosts should match. In the example, we want to install the host with the IP address `172.16.1.253`, previously listed in the **Discovered Hosts** tab. Then click **Next.**
2. Select the image to assign to the host then click **Next**.
3. Select the host profile to apply. If you don't have any host profiles available, flag the **Do not include a host profile** option and click **Next**.
4. Specify the location, cluster, or folder where the host should be added and click **Next**. I tend to use a dedicated empty cluster called `Deployment` or similar, so my new ESXi host is added to a dedicated cluster first before everything is tested.
5. Click **Finish** to create the rule. By default, the rule is disabled and must be activated using the **Activate/Deactivate rules...** button as well as specifying the deploy rule order. To modify an existing rule, the rule must be first deactivated from the **Activate/Deactivate rules...** button to allow editing:



6. Restart the host.

The boot process of the ESXi provisioned with vSphere Auto Deploy is different from the interactive or unattended installation methods:

Once the server is up and running, you will see it in the vCenter Server inventory in the location you have specified by the deployment rule:



As you can see, there are several notifications associated with our new host. This is because we did not include a host profile in the deployment rule that specifies how the host should be configured.

Auto Deploy modes

Having completed the Auto Deploy installation procedure, let's walk through the different modes you can use to configure vSphere Auto Deploy. There are three possible installation types you can use:

- **Stateless**: The ESXi image is not technically installed, but it is loaded directly into the host's memory as it boots.
- **Stateless caching**: The image is cached on the local disk, remote disk, or USB. If the Auto Deploy server is not available, the host boots from the local cache.
- **Stateful**: The image is cached on the local disk, remote disk, or USB. As compared to stateless caching, the boot order is inverted; the host boots first from local disk then from the network.

Let's have a look at the different procedures to configure Auto Deploy installations.

Stateless installation

Stateless installation follows the procedure previously seen where the host receives the configured image profile when it boots. This installation method requires an available image profile and a deployment rule that applies to the target host.

When you make a change to the Host Profile that is associated with the ESXi server in the Auto Deploy rule, the change will be applied to the host during the boot.

In this case, the Auto Deploy infrastructure must always be available. Otherwise, the ESXi server won't boot.

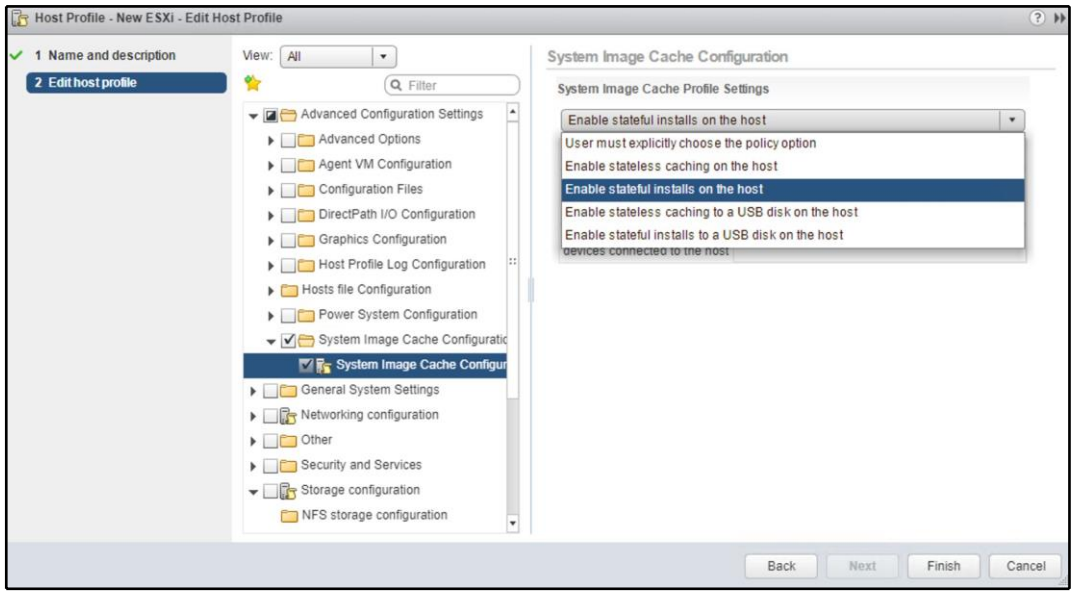
Stateless caching installation

During the ESXi deployment through Auto Deploy, the image is cached on local disk, remote disk, or USB drive. vSphere Auto Deploy always provisions the host, but if the server becomes unavailable due to bottlenecks (for example, hundreds of hosts that attempt to access the Auto Deploy server simultaneously), the host boots from the cache and attempts to reach the Auto Deploy server to complete the configuration.

The stateless caching solution is primarily intended to prevent situations where the deployment process may fail due to server congestion, a typical scenario that occurs in large environments.

To enable stateless caching mode, follow these steps:

1. From vCenter Server, navigate to **Home | Host Profiles**.
2. Edit an existing host profile attached to hosts to provision or create a new one.
3. Under **Advanced Configuration Settings**, select **System Image Cache Configuration**.
4. From the drop-down menu, select **Enable stateless caching on the host** and click **Finish** to save the configuration. You can specify a comma-separated list of disks to use (by default, the first available will be used) using the syntax shown in Table 4.1 to configure an unattended installation:



The host profile configuration must be modified to enable stateless caching.

5. Configure the boot order from the BIOS of your server to boot from the network first then from the local disk. Reboot the host to get a fresh image.
6. After a successful boot, the Auto Deploy image loaded in memory is saved to the local disk.
7. When you reboot the host and Auto Deploy is not available, the host boots from the cached image on local disk.

Stateful installation

The stateful installation method is almost the same as the stateless caching mode, with the exception that the boot order in the host's BIOS is inverted. Stateful installation is a method to perform a network installation because, after the first successful boot, Auto Deploy is no longer needed.

To enable stateful mode, follow these steps:

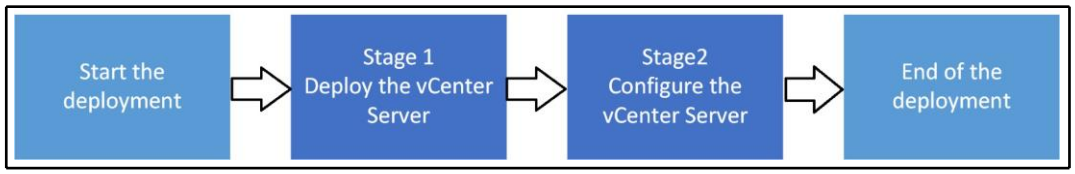
1. From vCenter Server, navigate to **Home | Host Profiles**. Edit an existing host profile attached to hosts to provision or create a new one.
2. Under **Advanced Configuration Settings**, select **System Image Cache Configuration**.
3. From the drop-down menu, select **Enable stateful installs on the host** and click **Finish** to save the configuration. You can specify a comma-separated list of disks to use (by default, the first available will be used).
4. Configure the boot order from the BIOS of your server to boot from the local disk first then from the network. Reboot the host to get a fresh image. During the boot process, settings stored in the host profile are applied to the host.
5. When the host boots, it will enter maintenance mode. At this stage, the settings passed with the host profile configured with Auto Deploy must be applied to the host. The host remediation action must be performed to complete the deployment process.

We will discuss host profiles in more detail in upcoming chapters, so do not worry.

vCenter Server components

vCenter Server is a service that centralizes the management of the ESXi hosts and the VM that run on the hypervisor. This vSphere core component not only interacts with ESXi hypervisors, but also integrates with other VMware products—vRealize Automation, Site Recovery Manager, and vSphere Update Manager, to give you some examples.

vCenter Server is not limited to act as a central management tool. The advanced features such as a **sign-on server** (**SSO**), centralized authentication, vMotion, DRS, HA, and FT are all services that come into play only when vCenter Server is present in the infrastructure. With vCenter Server, you can manage resources, ESXi hosts, VM, templates, logs and stats, alarms and events, and so on. Besides, vCenter Server provides all the functionalities needed to distribute and manage the network services, ensuring the availability of resources and data protection.



Starting from vSphere 6.0, the vCenter installation includes the deployment of two components:

- Platform Service Controller
- vCenter Server

PSC

Introduced in vSphere 6.0, the PSC is a component used to provide common infrastructure services for VMware products.

The PSC is an essential component in the design that provides services not only for vCenter Server and vSphere but the VMware product suite in general. SSO, for example, can also be shared with other VMware products to provide centralized user authentication (for example, vRealize Orchestrator, and vRealize Automation).

Depending on your environment and the infrastructure design, vCenter Server and the PSC can be deployed in two different ways—embedded or external:

- **Embedded**: Preferred deployment for single-sites where you do not need to interconnect different vCenter Servers to the SSO domain. vCenter Server can be deployed with an embedded PSC to simplify the management and, because both components are not connected over the network, outages due to connectivity and name resolution issues between vCenter Server and PSC are avoided. If the vCenter Server used is the Windows-based version, you can also save some Windows licenses. If you install vCenter Server with an embedded PSC, you can reconfigure the setup and switch to vCenter Server with an external PSC later on.
- **External**: Installing the vCenter Server with an external PSC is a solution suitable for large environments with the benefit that shared services in the PSC instances consume fewer resources. This setup increases the management complexity and, in the event of connectivity issues between the vCenter Server and PSC, could cause some outages.

Which method to use strictly depends on the requirements regarding availability for your vCenter Server. You can have a PSC that serves multiple sites or a highly available PSC in a single cluster.

VMware recommends six high-level PSC topologies:

- vCenter Server with embedded PSC
- vCenter Server with external PSC
- PSC in replicated configuration
- PSC in HA configuration
- vCenter Server deployment across sites
- vCenter Server deployment across sites with a load balancer

For more information about moving from a deprecated to a supported vCenter server deployment topology before upgrade or migration, you can visit `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vcenter.upgrade.doc/GUID-080CA000-4BD0-40F8-8324-DABB3A136390.html`.

Some topologies have changed from version 5.5 and are now deprecated. The choice of the right topology depends on different aspects, such as features (do you need enhanced linked mode between multiple vCenters?), availability, scalability, physical topology, and so on.

Although a mixed environment is supported, it is recommended that you use the same platform (only appliances or only Windows-based installations) for both vCenter Server and PSC to ensure easy manageability and maintenance.

There are three core services provided by the PSC essential for the vSphere functionality— SSO, VMware License Service, and certificate management:

- **SSO:** This is a prerequisite to installing vCenter Server (it cannot be installed without SSO). This service solves the problem of authentication in an environment with multiple ESXi hosts. Using a secure token mechanism, vSphere components can communicate with each other without requiring a separated authentication for each component. For each administrator who needs access to a specific server, without having a vCenter Server in your environment, you need to create a separate user account and grant access permissions for each ESXi. If the number of ESXi hosts grows, the number of accounts to manage also grows. Joining the ESXi to AD to centralize the authentication can be an option, but adds another dependency in the infrastructure—the **Domain Controller** (**DC**). The SSO authentication service is easier to manage and more secure for the authentication against VMware products.
- **VMware License Service:** This centralizes the management of all the information related to the license of the vSphere environment and VMware products that support PSC. This capability allows licensing information between vCenter Servers not configured in the Linked Mode group installed in geographically different locations to replicate every 30 seconds (by default).
- **Certificate Management:** This is required to communicate securely with each other and, with ESXi hosts, vCenter Server services make use of SSL. The **VMware Certificate Authority** (**VMCA**) provisions ESXi hosts and services with a certificate signed by VMCA by default.

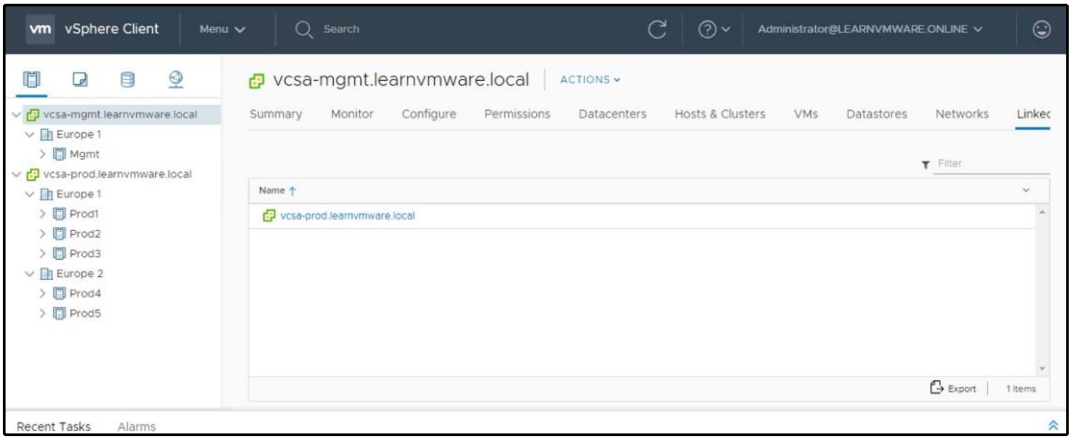
Other services provided by PSC are as follows:

- VMware Appliance Management Service (only in appliance-based PSC)
- VMware Component Manager
- VMware Identity Management Service
- VMware HTTP Reverse Proxy
- VMware Service Control Agent
- VMware Security Token Service
- VMware Common Logging Service
- VMware Syslog Health Service
- VMware Authentication Framework
- VMware Directory Service

Linked Mode

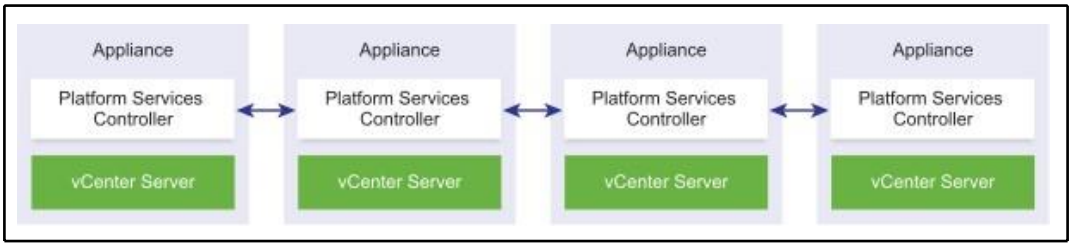
Linked Mode is a feature where you can link multiple vCenter Servers to the same PSC, allowing you to manage multiple vCenter Servers from the single vSphere client.

With this, you can join multiple vCenter Server systems using vCenter Linked Mode, and this will enable them to share information with each other. You can also view and manage the inventories of other vCenter Server systems when a server is connected to it using Linked Mode:



In the past, there were some limitations regarding Linked Mode and vCSA. Those limits no longer apply, and you can even link multiple vCSA appliances with embedded PSC. This mode is called Embedded Linked Mode.

vCenter Embedded Linked Mode is supported starting with vSphere 6.5 Update 2 and suitable for most deployments:



According to the vSphere-vCenter installation guide, `https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-vcenter-server-67-installation-guide.pdf`, the other features of vCenter Embedded Linked Mode include the following:

- No external PSC. This provides a more simplified domain architecture than an external deployment along with the enhanced linked mode.
- Provides a simplified HA process, removing the need for load balancers.
- Up to 15 vCenter Server Appliances can be linked together using vCenter Embedded Linked Mode and displayed in a single inventory view.

vCenter Server

VMware vSphere 6.7 is the last version that will support vCenter for Windows deployment. From the next version, vCSA will be the only option to run vCenter Server.

If you want to install vCenter for Windows with version 6.7 it is still fully supported deployment, but keep in mind that with the next upgrade, you will have to migrate vCenter for Windows to the vCSA.

vCenter Server provides the following services:

- **Web Client**: The vSphere Web Client lets you connect to vCenter Server instances by using a web browser so that you can manage your vSphere infrastructure.
- **Inventory Service**: This stores vCenter server inventory data and server application data. It also allows you to search the inventory objects across linked vCenter server instances.
- **Profile driven storage**: This is a component in VMware vSphere that allows users to intelligently provision applications, mapping VMs to storage levels according to pre-defined service levels, storage availability, performance requirements, or cost.
- **Auto Deploy**: This feature allows you to deploy and provision hundreds of physical servers and automatically install the ESXi hypervisor. Optionally, you can specify host profiles to apply to the hosts and a vCenter Server location (folder or cluster) for each host.
- **Syslog Collector**: ESXi system logs can be redirected to a vCenter server over the network, rather than storing them on a local disk. Using Syslog Collector, you can centralize the log management of all your ESXi servers.
- **Network Dump Collector**: This is the vCenter Server support tool. ESXi can be configured to save the VMkernel memory to a network server, rather than saving it to a disk when the system encounters a critical failure. Such memory dumps over the network will be collected by the vSphere ESXi Dump Collector.

Since version 6.5, vCenter Server Appliance is a preferred deployment type, and vCenter Server for Windows is still fully supported in vSphere 6.7 as well. If you want to follow VMware best practices, you should switch to vCSA.

Migration from vCenter for Windows to vCSA

If you are thinking about moving from vCenter for Windows to vCSA, you can use the migration wizard that is fully integrated into the vCSA installation.

During the migration, new vCSA appliance and PSC will be provisioned with a temporary IP address.

Once the temporary appliance (or appliances if you have a dedicated platform service controller) is up and running, the migration wizard will connect to the source vCenter for Windows (and PSC) and migrate the configuration and all performance data to the new vCSA.

In the end, the migration wizard will disconnect the source vCenter Server from the network (if you run the vCenter server in a virtualized environment) and change the temporary IP address of the new vCSA appliance to the production IP address of the previous vCenter server.

Where to install – physical or virtual?

One recurrent question about vCenter Server installation is whether it should be installed on a physical server or a VM. Technically, vCenter Server can be installed on both destinations, but I prefer deploying on a VM. Why?

If you have the vCenter Server installed on a VM and the ESXi that hosts the vCenter Server fails, HA will restart the VM on another node, ensuring service availability. If a physical server with vCenter Server installed fails, you lose not only the vCenter Server but all the services it provides. The best option would be having a management cluster with vCenter running on it (this perhaps makes more sense for large environments), but it would be an expensive solution the business could not afford/approve.

Another option could be placing the vCenter Server in the running cluster of your vSphere environment, a common approach for small environments. In large environments, if you need to shut down the infrastructure or perform some maintenance, it could be useful to know precisely which ESXi is hosting the vCenter Server without wasting time on research between hosts.

VMware recommends deploying vCenter Server on a VM, suggesting the use of the vCSA. vCSA is replacing the Windows-based vCenter Server which will be deprecated quite soon.

vCenter Server Appliance deployment

vCSA is prepackaged and preinstalled Photon Linux-based VM that provides vCenter and PSC services. As compared with old versions, vCSA now offers the same capabilities provided by the Windows-based version plus some exclusive services, such as native HA, native backup and restore, a migration tool, and improved appliance management.

With vSphere 6.7 you can run the vCSA GUI and CLI installers on Microsoft Windows 2012 x64 bit, Microsoft Windows 2012 R2 x64 bit, Microsoft Windows 2016 x64 bit, and macOS Sierra. The new capabilities make the appliance complete and ready to take over the Windows-based version.

The following is a short description of features available in vSphere 6.7:

- **Native HA**: This feature, available for vCSA only, is a solution to provide HA to your vCenter. You could have the active vCSA in one data center and the passive vCSA located in a DR or secondary data center. It removes the dependency on expensive third-party database clustering solutions of RDMs.
- **VUM**: This is now embedded into vCSA. You no longer need a separate Windows VM and an additional license.
- **HTML5 interface:** You no longer need to use the old Flex client. In vCenter 6.7, almost 95% of features are fully integrated into the new HTML5 client. VMware promises that all features will be available in vCenter 6.7U1, which should be available in Q4 2018.
- **Native backup and restore:** The process has been simplified with a new native file-based solution. It restores the vCenter Server configuration to a new appliance and streams backups to external storage using HTTP, FTP, or SCP protocols (vCSA only).

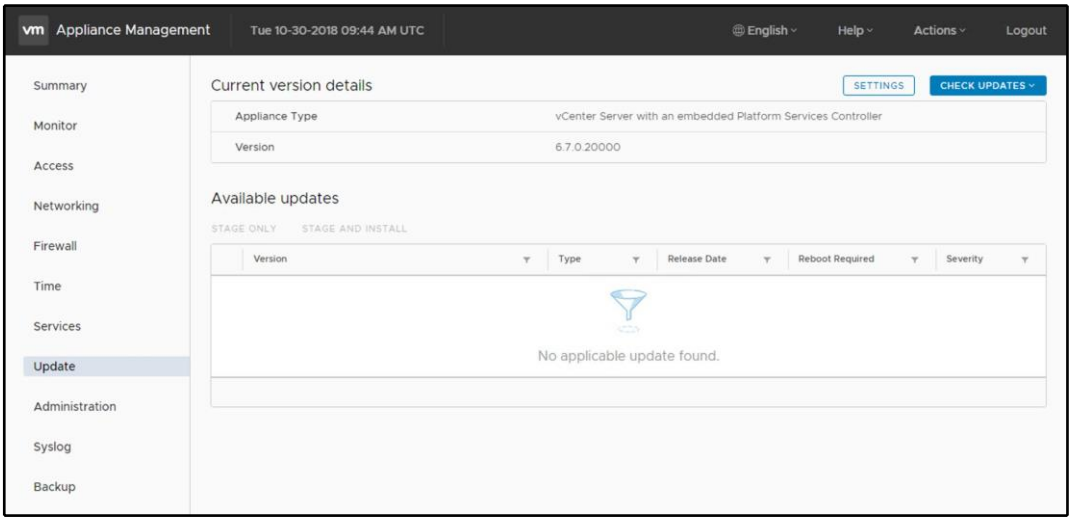
The installation procedure has been simplified, and now vCSA, and PSC installation is a two-stage process:

- **Stage 1:** Deploying OVF
- **Stage 2:** Configuration

This is a significant enhancement and provides not only better validation checks, but also you can take a snapshot between stages for rollback. Besides, you can create a template for additional deployments.

Introduced with vSphere 6.0 Update 1, the Appliance Management client (accessible at the address `https://<VCSA_IP>:5480`) simplified the configuration and upgrade process. Now you can patch or upgrade the appliance through ISO-or URL-based patching, simplifying the process and allowing you to save precious time.

vCSA updates can be applied directly from the VAMI using an intuitive GUI:



Please note that the update of vCSA is available only between minor versions, for example, from 6.7.0.10000 to 6.7.0.20000, not between major versions.

There are many more features of the management interface of vCSA, such as backup configuration, services overview, and storage consumption.

Why deploy vCSA instead of the Windows version?

There are several reasons why you should deploy vCSA:

- Being a packaged and installed vCenter, the deployment is quick, and you only need to supply a few details.
- The embedded PostgreSQL database supports up to 2,000 hosts and 35,000 VMs, so there is no difference in configuration maximums between vCenter for Windows and vCSA.
- No need for extra Microsoft Windows licenses. Since VUM is now embedded, there is no need for a separate Windows box.

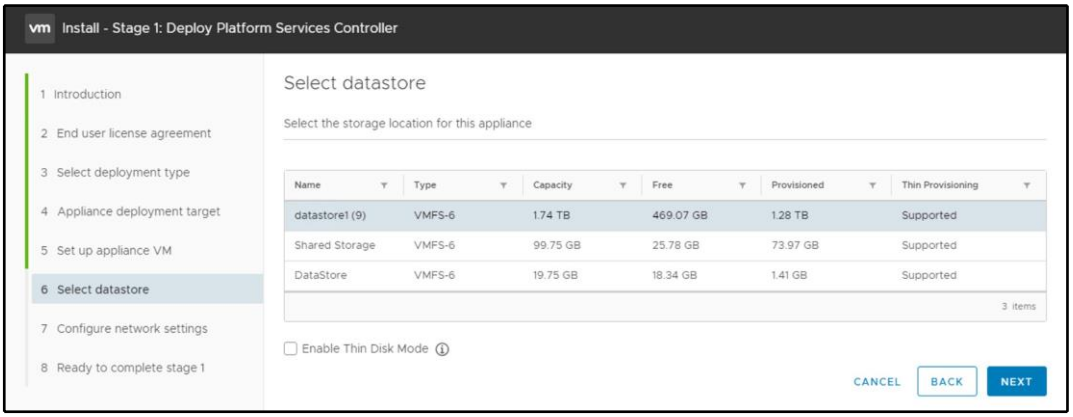
- Having now identical features, it's only a matter of time before VMware drops the vCenter Windows version permanently.
- vCSA 6.7 runs on Photon Linux OS and generally is a more secure OS compared to Windows.
- Less hardware to use since vCSA can be deployed only as a VM. This allows you to reduce costs.

After having analyzed the reasons for making vCSA the preferred choice for your vCenter Server and the benefits it brings, let's walk through the installation process.

Installing the vCSA PSC

As the vCSA installer with a new look, independent of a browser, now also support macOS, Linux, and Windows, you can use the system you are more familiar with. Before proceeding with the installation, make sure you enter the new host in the DNS to both forward and reverse resolve. Perform the following steps:

1. Mount the ISO and run the installer.
2. When the main screen appears, there are four actions you can do—**Install**, **Upgrade**, **Migrate**, and **Restore**. Click on **Install**.
3. Click **Next** to begin stage 1. When prompted, accept the EULA and click **Next**.
4. As seen previously, the deployment type to use depends on the size of your environment. For this example, we are going to install vCenter Server with an external PSC. Select the option and click **Next**.
5. vCenter Server can be deployed with an external PSC by selecting the correct option in the installation wizard.
6. Specify the ESXi or vCenter target settings and the host credentials. Click **Next**.
7. Click **Yes** to accept the self-signed SSL certificate.
8. Enter the PSC name and the root password then click **Next**.
9. Next, you need to specify storage options. Here, you have the option to enable thin-provisioned disks, but this is not recommended for production environments. Click **Next**:



10. Configure the networking the vCSA appliance should use. Make sure the DNS for the IP used can both forward and reverse resolve to avoid errors. Click **Next**.
11. In the **Summary** window, click **Finish** to deploy the PSC.
12. When the deployment completes, click **Continue**. Stage 1 is now complete.

At this point, you can take a snapshot before proceeding with stage 2.

The installation continues with stage 2, performing the configuration of NTP and SSO services:

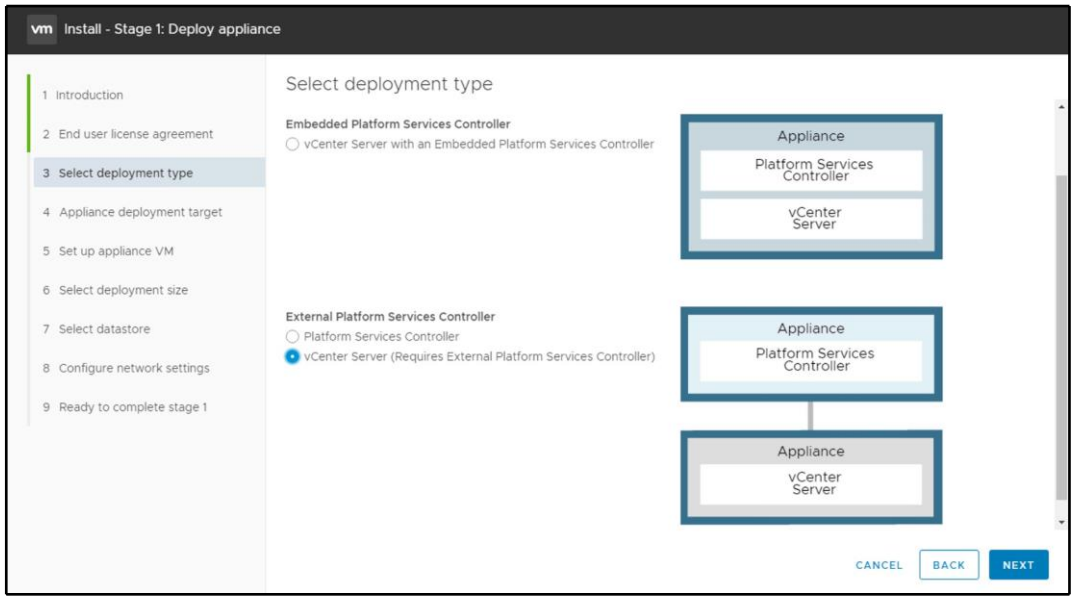
1. From the main screen, click **Next** to begin.
2. Time synchronization is the first option to configure to avoid communication issues with hosts. Here, you can also enable or disable SSH. Click **Next**.
3. Configure SSO, specifying a domain name, password, and site name. Click **Next**.
4. Feel free to join the **Customer Experience Improvement Program** (**CEIP**). Make your choice and click **Next**.
5. In the Summary window, click **Finish**. This will complete stage 2 and the installation of the PSC

When the installation process is complete, the PSC is fully working and can be accessed via the browser.

Installing the vCSA vCenter

To install the vCSA vCenter, you should run the installer once again, repeating a similar procedure to that which was used to deploy the PSC component:

1. During the vCenter Server deployment procedure, at step 3 click **vCenter Server (Requires External Platform Services Controller)** under the **External Platform Services Controller** option and then click **Next**:

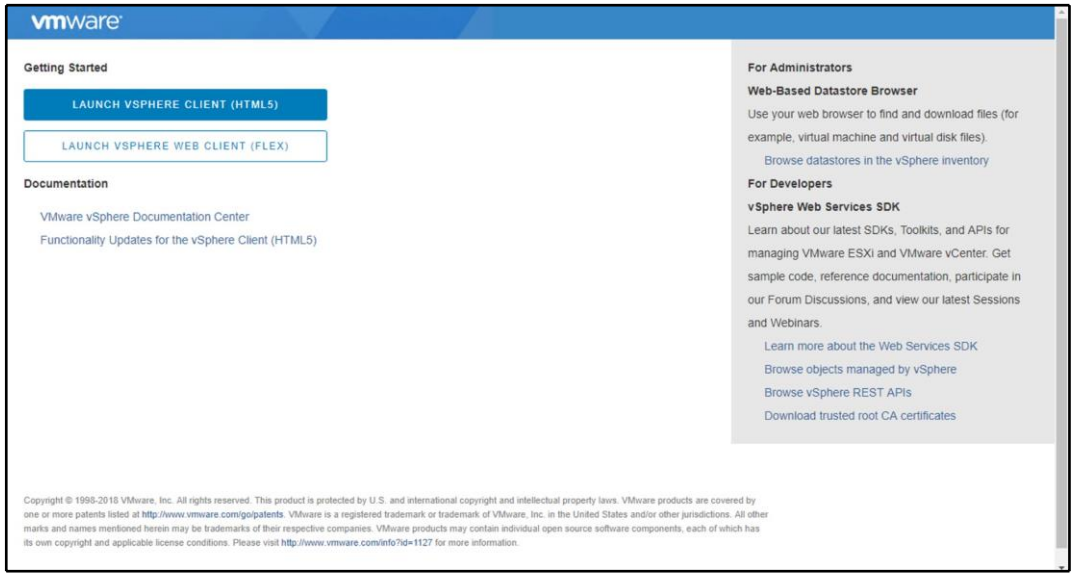


2. After specifying the storage options, in step 8 during stage 1 of the PSC deployment, you should specify the deployment size for the vCenter based on your environment. Make your choice then click **Next**.
3. Continue the installation procedure by following the remaining steps until you complete stage 1.
4. When the stage 2 installation process begins, click **Next**.

5. Specify NTP servers in the NTP servers (comma-separated list) field and set the SSH access option as Enabled. Click **Next** to continue the configuration.
6. In the SSO configuration page, specify the PSC appliance to connect, enter the SSO domain and SSO password, then click **Next**. You can create a new, or join an existing, SSO domain.
7. In summary, click **Finish** to complete the vCenter Server installation

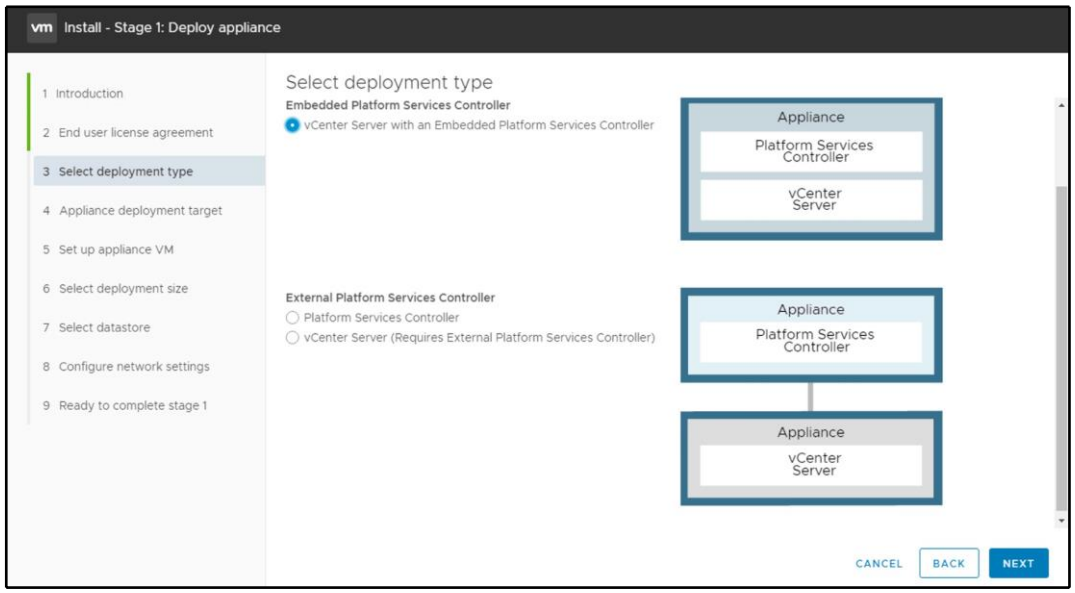
In vSphere 6.7, the only way to log in to vCenter is through the two integrated web clients:

- **Flash-based web client**: `https://<VCSA_IP>/vsphere-client`
- **HTML5 web client**: `https://< VCSA _IP>/ui`



Installing the vCSA with Embedded Platform Service Controller

If you do not need to run separated Platform Service Controllers, the installation procedure is the same except in step 3 of the installation wizard. At this point, select **vCenter Server with an Embedded Platform Services Controller** as the deployment type:



In stage 2 of the installation, instead of pointing your vCenter Server to the Platform Service Controller, you will have an option to create new SSO domain – a similar one as with PSC stage 2 configuration.

If you do not need to run separated PSCs (for example, because of the mixture of vCSA and vCenter Server for Windows), there is no need to run a dedicated PSC.

vCSA HA

In the past, you could only rely on vSphere HA which would automatically restart your vCSA in case of the hardware failure, but this might lead in the corrupted system state as with any other OS.

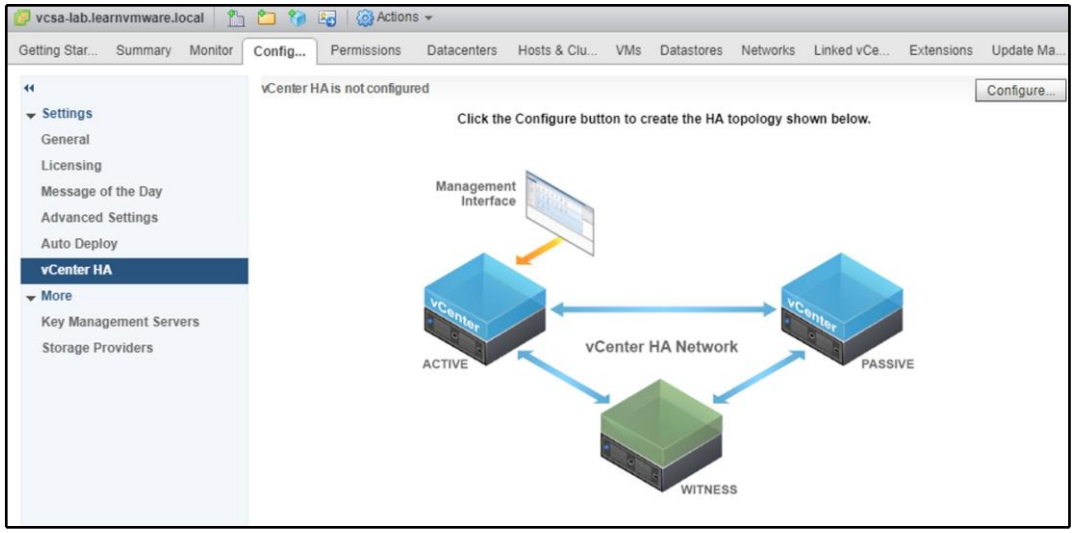
If you aren't familiar with vCenter HA, it is a feature introduced in vSphere 6.5 and available only for the vCSA. When you enable vCenter HA, secondary passive vCSA is deployed along with the witness appliance.

vCenter HA provides short RTO (about five minutes) for recovery of the vCenter Server. When the hardware where the active node is running fails, the passive vCenter Server will take over, shortening the total downtime of the vCenter Server. vCenter HA is a part of the vCenter Server Standard license, so no additional licensing is required.

vCenter HA is only available in the vCSA, and you can't deploy this configuration with vCenter for Windows.

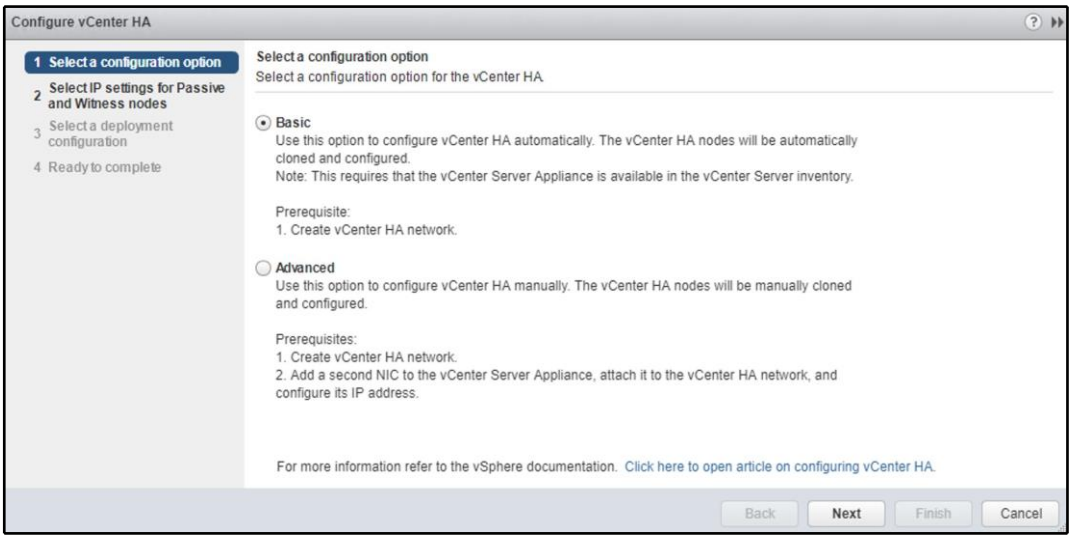
vCenter HA configuration

The configuration of vCenter HA is pretty simple, and it's done from the vCenter server web client. At this stage, you can't use the new HTML5 interface, and the configuration is done through the older FLEX client:



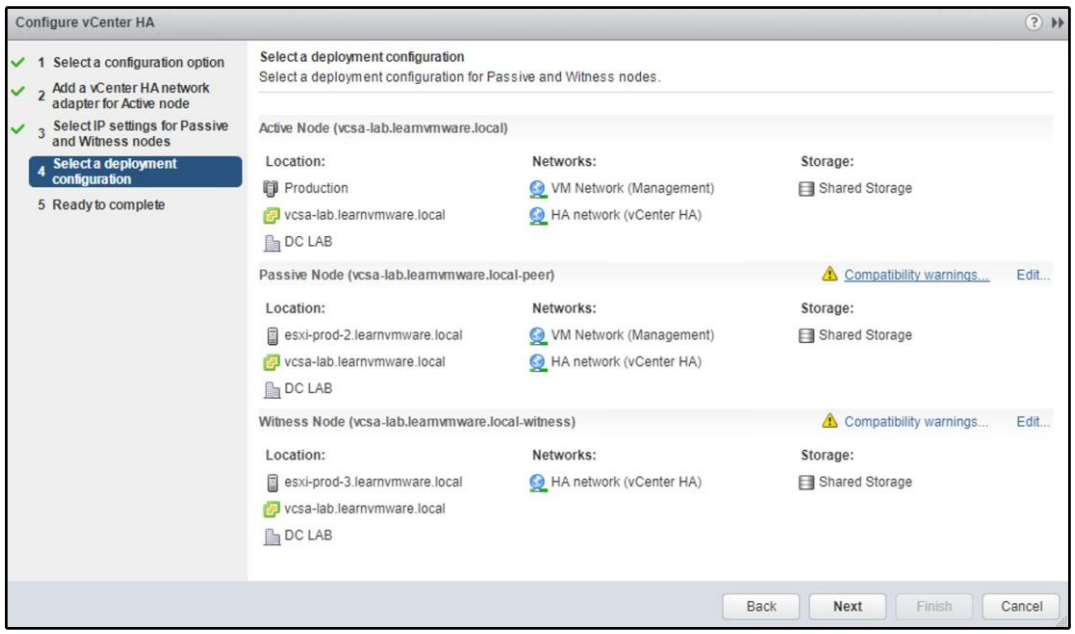
There are two deployment modes—basic and advanced.

With basic deployment, you need to run vCSA on the same environment that vCSA is managing. If you have a dedicated management cluster where your vCSA is installed, you need to perform advanced configuration since, during the vCenter HA setup, the source vCSA will be cloned, and this won't be possible if the vCSA is not located within the same environment, of course:



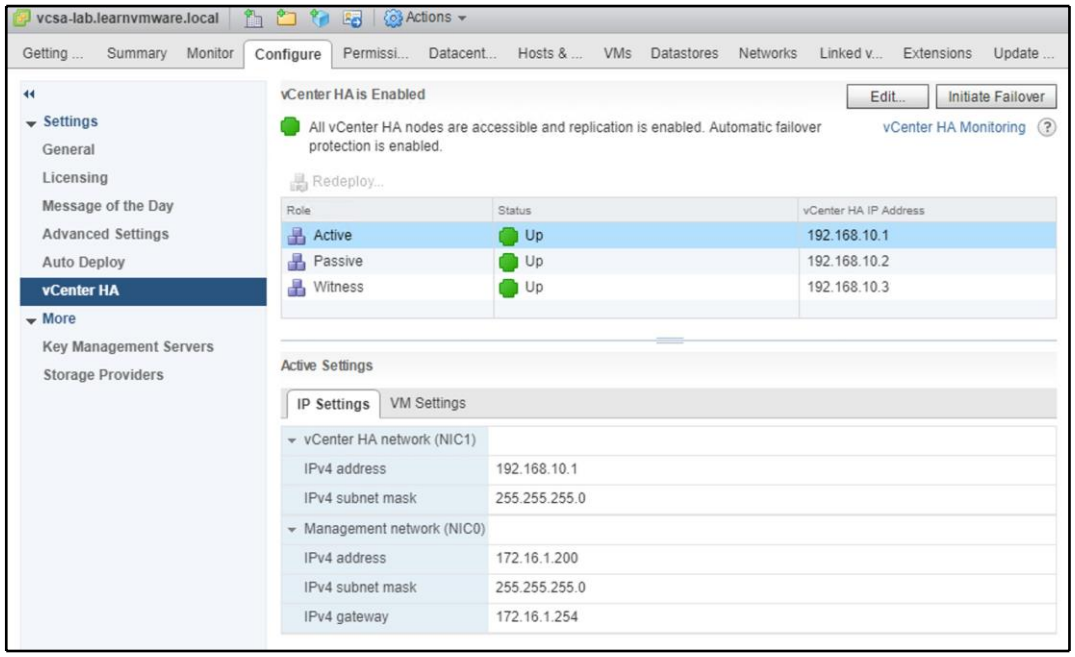
Once you have decided what deployment type you will need, you need to set up the HA network. The HA network is used for internal communication between two vCSA servers and the witness appliance.

As the last step, you need to configure settings for the peer vCSA and the witness appliance. On which datastore the appliances will be stored, portgroup association or cluster on which the VMs will be run:

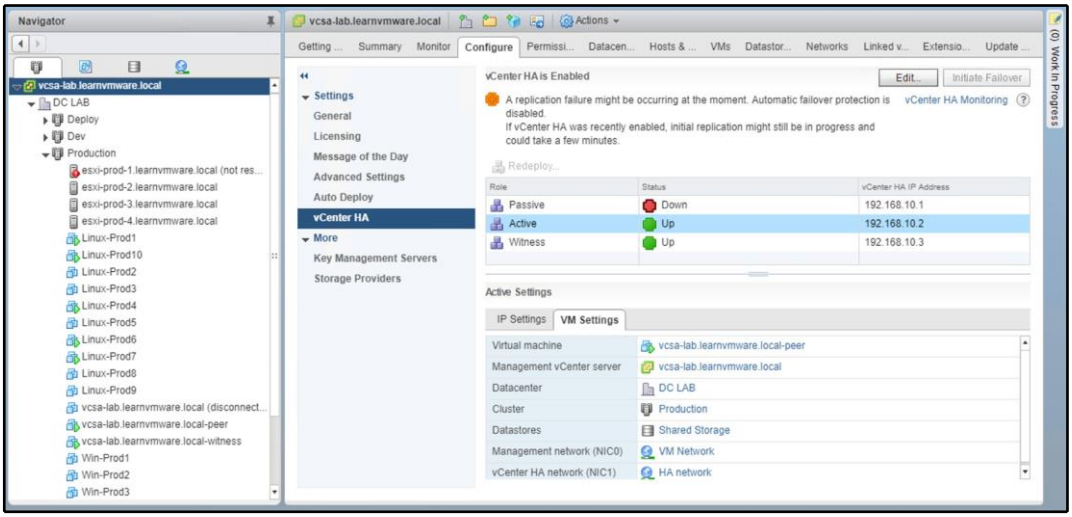


In this case, the configuration wizard complains about using the same datastore as the production VMs, which is not the best practice. You should always use dedicated datastores for your management and the production workloads. Once you hit **Finish**, you can check your running tasks to see at what stage the deployment is.

Once the deployment is finished, you will see the current state of the cluster, which is the active node, and the state of the passive node and witness appliances:



In the case of hardware failure, the passive appliance becomes active, and you will still be able to manage your environment even if the originating vCSA appliance is no longer available:



Please note that the failover process takes several minutes. If you try to access the web client of the vCSA during the failover, you will see that the failover is in progress.

11
Configuring and Managing vSphere 6.7

This chapter will cover the configurations required by ESXi and vCenter Server to provide services and resources to a **virtual machine** (**VM**). We will look at how to set up the hypervisor properly, how to assign the correct IP address, and how to configure a time-synced network to get a working infrastructure.

This chapter will also walk through the configuration of the main parameters and features of **vCenter Server Appliance** (**vCSA**), such as **single sign-on** (**SSO**), **Active Directory** (**AD**), roles, permissions, and more. We'll explore how to manage data centers, clusters, and hosts efficiently using the new vSphere Client (HTML5 client). We will also focus on backing up the configuration of the ESXi hypervisor and vCSA.

The use of PowerCLI and the vSphere REST API are other important topics that will be covered in this chapter, because time-consuming tasks can be automated and executed in seconds using scripts, therefore reducing the workload for IT staff.

In this chapter, we will cover the following topics:

- Using the VMware vSphere HTML5 client
- Configuring ESXi
- Backing up and restoring ESXi
- Configuring vCSA
- Exporting and importing vCSA configuration
- Managing data centers, clusters, and hosts
- Automating tasks with scripts

Using the VMware vSphere HTML5 client

The HTML5 client was introduced in VMware vSphere 6.5 and it evolved significantly between that and VMware vSphere 6.7. vSphere 6.7 U1 was released on October 27, 2018, and, according to the enhancement list, the HTML5 interface now supports all the functions available in the FLEX client.

The new client, called vSphere Client (in this book, we will call it the HTML5 client to clarify the type of client), comes from the vSphere HTML5 Web Client Flings project (`https://labs.vmware.com/flings/vsphere-html5-web-client`). This client is still available if you want to add this functionality to a vSphere 6.0 infrastructure. With the release of vSphere 6.7, the reach of the HTML5 client development increased, covering 95% of the workflow.

The new client is entirely built on HTML5. It requires no plugins and is lighter and much faster than the flash-based client. The vSphere **Client Integration Plugin** (**CIP**) has been deprecated, and it no longer works for connecting vSphere 6.7 components. Both the flash and HTML5 clients are automatically installed as part of the vCenter deployment process. The HTML5 client can be accessed through `https://<VCSA_IP>/u`.

To log in to the ESXi host, version 6.7 provides a new built-in HTML5 client, which is available at `https://<VESXi_IP>/ui`. The HTML5 client started as the Flings project (`https://labs.vmware.com/flings/esxi-embedded-host-client`) and was later integrated into ESXi 6.0 U2, becoming the only client in version 6.7.

Configuring ESXi

When the installation of the ESXi host is complete, there is some configuration you need to do to connect the hypervisor with the network infrastructure. There are also other suggested configurations that you should perform after a clean installation of the ESXi hypervisor.

In this section, we'll walk through the main settings that you should configure.

Management network configuration

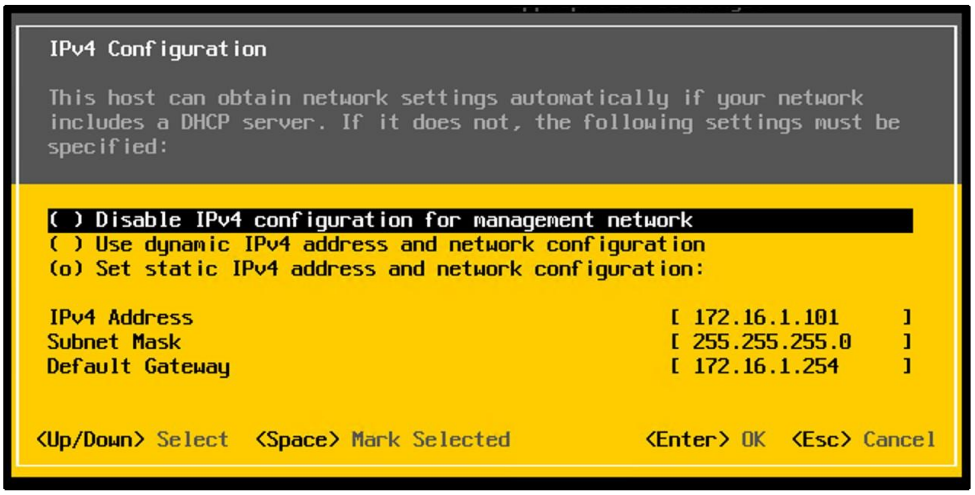
By default, ESXi is configured to receive the IP address for the management console through the **Dynamic Host Configuration Protocol** (**DHCP**). If no DHCP server is present in your network, the hypervisor doesn't obtain an IP, and you won't be able to connect.

Assigning a dynamic IP address to ESXi is not a recommended configuration because management and services are linked to specific IP addresses assigned to the server. If the IP changes (after rebooting the host or for an expired lease), some services may not work as expected. If the server has multiple physical **network interface controllers** (**NICs**) installed and the DHCP assigns an IP address to an NIC linked to a wrong vSwitch, you may experience connectivity issues that will impede the connection to the management console, and you may not be able to manage the host.

Configuring a static IP address to the ESXi management console is the recommended configuration to adopt. The ESXi management console can be configured by accessing the **Direct Console User Interface** (**DCUI**) directly, or through the iLO, iDrac, IPMI, or similar (if your server has an integrated remote management console). Proceed with the following steps:

1. Access the ESXi console and press *F2* to access the **Customize System/View Logs** option.
2. When requested, enter the root password set during the installation process.
3. Select **Configure Management Network** from the **System Customization** menu and press *Enter*.
4. Select **Network Adapters** from the **Configure Management Network** menu and press *Enter*.
5. Using the spacebar, select the NIC to use for the ESXi management and press *Enter*. Press *D* to see details related to the selected NIC (for example, the attached vSwitch).
6. Now, select **IPv4 Configuration** and press *Enter* to assign a static IP address.

7. Use the spacebar to select the **Set static IPv4 address and network configuration** option, then configure the **IPv4 Address**, the **Subnet Mask**, and the **Default Gateway**. Press *Enter* to save the configuration, as demonstrated in the following screenshot:

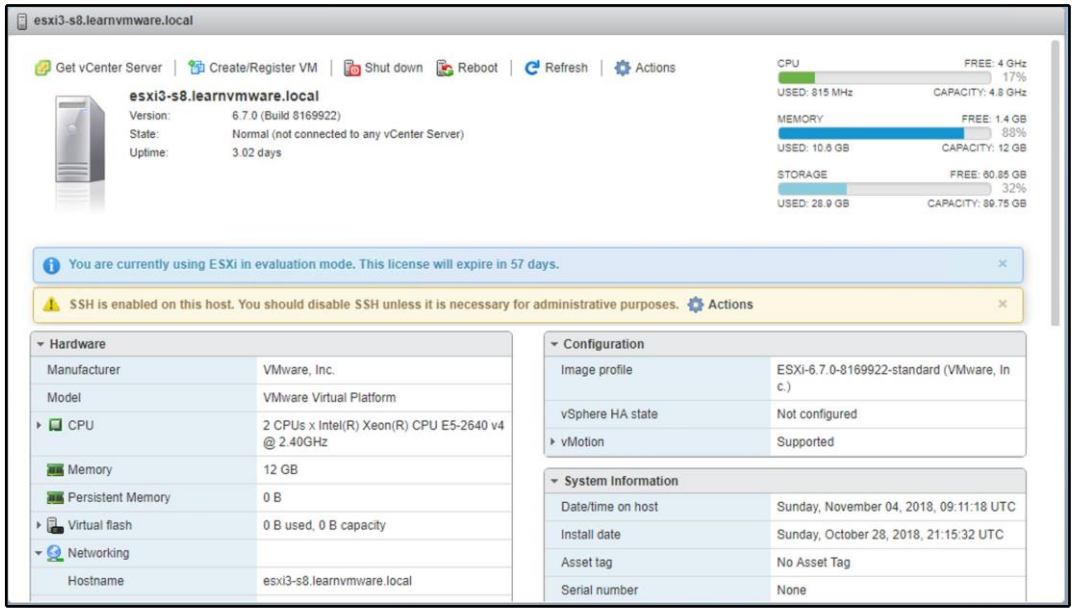


8. Select **DNS Configuration** to specify the **primary and alternate DNS servers** and the **hostname**. Press *Enter* to confirm the settings.
9. Select **Custom DNS Suffixes** to specify the suffix to use (`lab.local`, for instance). Press *Enter* to confirm.
10. Press *Esc* to exit the **Configure Management Network** console. Press *Y* to apply the changes when prompted.

As a best practice, you should always configure the DNS name of the system and use the **fully qualified domain name** (**FQDN**) when adding a host to vCenter Server.

Enabling Secure Shell (SSH) access

You may need to access the hypervisor through SSH for troubleshooting, or to perform some actions using **command-line interface** (**CLI**) commands. To access the ESXi host through SSH, you must enable the SSH protocol first, because it is disabled by default. For security reasons, it is suggested that you keep the SSH protocol disabled if it is not used. A warning message advises you that the SSH protocol is enabled, as shown in the following screenshot:



To enable SSH from the web console, log in to ESXi, and right-click the host. Select **Services | Enable Secure Shell (SSH)**.

To enable SSH from the DCUI, perform the following steps:

1. From **System Customization**, select the **Troubleshooting** option and press *Enter*.
2. Select **Enable SSH** and press *Enter* to change. SSH is now enabled.
3. The same procedure must be performed if you want to enable the ESXi Shell.
4. Press *Esc* to exit.

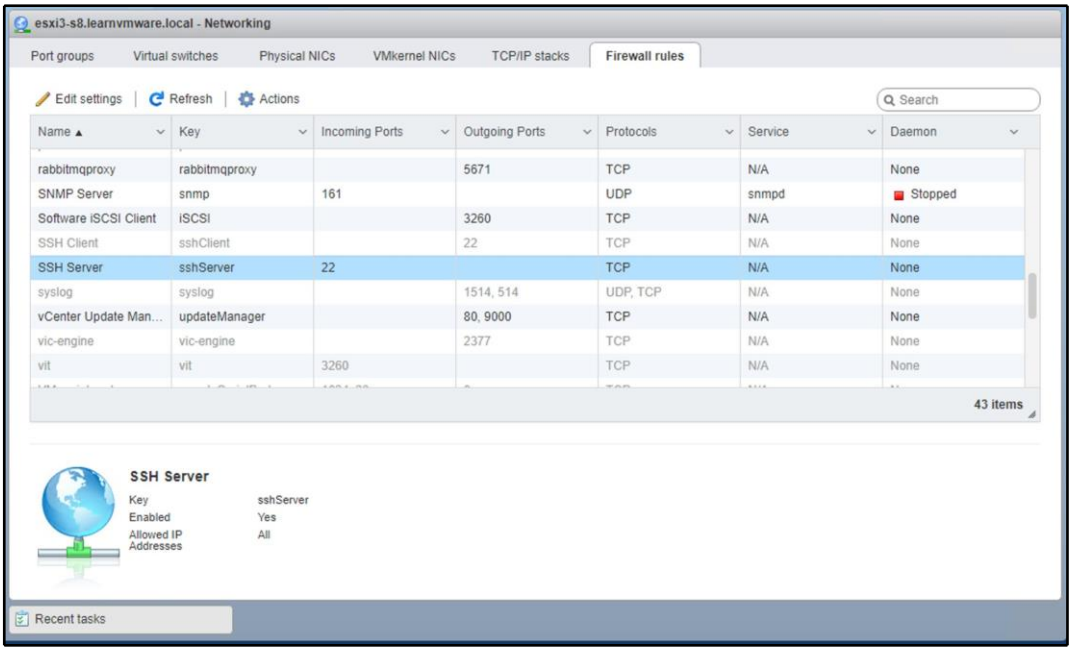
Based on VMware best practices and security hardening (which will be discussed in *Chapter 16, Securing and Protecting Your Environment*), I prefer to keep SSH running all the time. If there is an issue that needs to be troubleshot, I prefer to connect directly to the system without additional configuration steps.

If you don't like the warning notification that is displayed when the SSH service is enabled, you can alter the behavior with advanced configuration parameters. To configure the advanced parameters, perform the following steps:

1. From **Navigator**, select **Manage**, then from the **System** tab, select the **Advanced settings** option
2. Find the `User.Vars.SuppressShellWarning` variable
3. Edit the value of the property and change it to `1`

ESXi firewall

If you would like to harden the access to any ESXi service, you can also use the integrated firewall option of the ESXi hypervisor, so that only a limited number of IP addresses or IP ranges can access the service, as demonstrated in the following screenshot:



To reconfigure the ESXi firewall, perform the following steps:

1. Select **Networking** in **Navigator** and switch to **Firewall Rules**
2. Select the service you want to protect by a firewall
3. Click **Edit settings**
4. Configure the desired IP addresses or IP ranges

Note that you need to perform this configuration on every ESXi hypervisor. Also, keep in mind that misconfiguration can lead to several network-related consequences, as different VMware products may need to connect to the ESXi server as well.

More information about the required TCP and UDP ports can be found at the following link: `https://kb.vmware.com/s/article/1012382`.

Configuring the Network Time Protocol (NTP)

Time synchronization in your network should always be configured, but sometimes users underestimate its importance because they believe that having the network time-synced is not that important. It is, however, critical. If the ESXi hosts are not in sync, you might face some communication issues between vSphere components that could cause a service outage. If you use AD in your network, for example, the **Domain Controllers** (**DCs**) and clients must be time-synced to avoid authentication problems. If the time between the DCs and clients differs by more than five minutes, Kerberos tickets will fail, and you will not be able to log in. By default, machines joined to a domain will contact the DC that holds the **Primary Domain Controller** (**PDC**) emulator role to synchronize the time.

If your network is not time-synced, you may experience authentication issues between **vCenter Server** and the **Platform Services Controller** (**PSC**). When vSphere components are not time-synced, the login procedure may fail due to communication issues between the PSC and vCenter.

VMs use VMware Tools to synchronize the time with the host. Although a VM can be time-synced with the ESXi host using VMware Tools (VMs automatically synchronize the time when specific events occur, such as VM vMotion, snapshot creation, or guest OS reboots), it is recommended to synchronize the guest OS time with the NTP source instead.

To keep the time synchronized, ESXi supports the NTP, which you can configure through the vSphere Client. As a time source for your network, you should use a reliable external source, such as the `pool.ntp.org` project (a big virtual cluster of time servers providing a reliable, easy-to-use NTP service) or an internal source, such as a DC synchronized with an external time source.

Let's take a look at how to configure an NTP in your ESXi by performing the following steps:

1. Open the vSphere Client by typing the address, `https://<ESXi_IP>/ui` into your favorite browser, and log in to the host.
2. In the **Navigator**, select **Manage**. Go to the **System** tab and select **Time & date**.
3. Click **Edit settings** to open the time configuration window.
4. Select **Use Network Time Protocol (enable NTP client)** to specify the NTP parameters.
5. Select **Start and stop with port usage** (the recommended option) in the NTP service startup policy drop-down menu. In the NTP servers field, enter the NTP server to use. Specify the `pool.ntp.org` NTP servers to point the host to an external source directly, or enter the AD DC that holds the PDC emulator role configured to synchronize the time to an external source, to ensure the correct time.
6. Click **Save** to save the configuration.
7. Click **Action** and select **NTP service** | **Start** to start the service.

The time of the ESXi host is now synchronized with a reliable NTP server.

VMware recommends that you use NTP instead of VMware Tools time synchronization, as NTP provides more precise timekeeping on VMs. For resiliency, you should always use two independent time source servers.

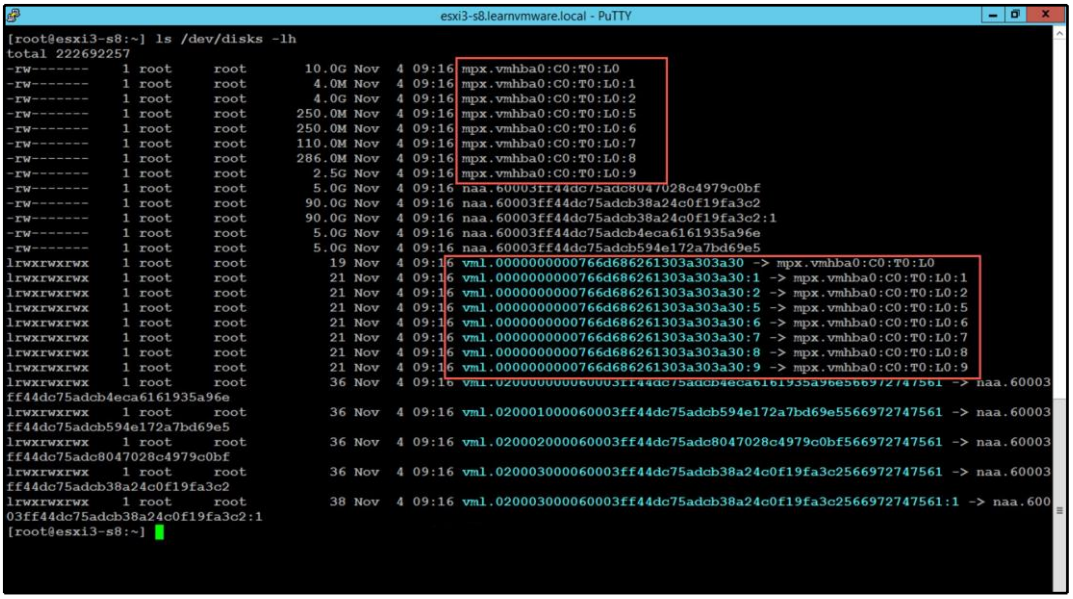
ESXi 6.7 partition layout

Regardless of the installation method you have chosen for your host, once the ESXi has been installed on the destination device, a specific partition layout is created on the disk. It is not possible to modify the partition layout during the installation process, and all the partitions are created automatically.

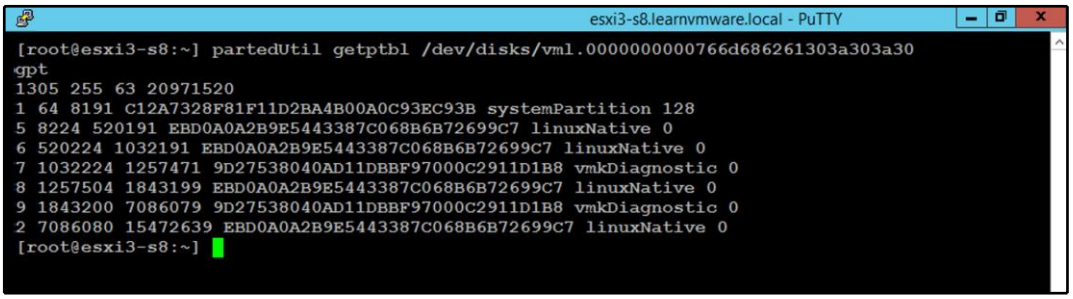
To identify the partition layout created by the installer in vSphere 6.5, you should use the `partedUtil` command, because the `fdisk` command was compatible with previous releases only. With the introduction of the **GUID Partition Table** (**GPT**) partition from ESXi 5.x, the `fdisk` command has been deprecated because it doesn't work anymore. To display the partition table, you need to access the ESXi console and run some specific commands.

Proceed as follows to display the partition table information:

1. SSH the ESXi and run the `ls /dev/disks -lh` command to identify the name of the system disk (usually, it is the only disk with more than one partition). As you can see in the following screenshot, the `mpx.vmhba0:C0:T0:L0` device has multiple partitions:



2. Once you have identified the system disk, you can use the `partedUtil` command with the `getptbl` option to see the partition size, as shown in the following screenshot:



You can also check the partition layout from the ESXi web client, as follows:

1. Select **Storage** from **Navigator** and switch to the **Devices** tab
2. Click on the device that was used for the installation

Looking at the preceding screenshots, the ESXi 6.7 host partition layout created by the ESXi installer can be composed of up to eight partitions. Partitions 2 and 3 may not be visible if the host is installed on SD cards or USB flash drives. Here are details regarding the partitions:

- **1 (systemPartition 4 MB)**: The partition needed for booting.
- **5 (linuxNative 250 MB—/bootbank)**: The core hypervisor VMkernel.
- **6 (linuxNative 250 MB—/altbootbank)**: This partition is initially empty because no previous version of ESXi is available.
- **7 (vmkDiagnostic 110 MB)**: This partition is used to write the host dump file if ESXi crashes.
- **8 (linuxNative 286 MB—/store)**: This partition contains the VMware Tools ISO file for the supported OS.
- **9 (vmkDiagnostic 2.5 GB)**: This is the second diagnostic partition.
- **2 (linuxNative 4.5 GB—/scratch)**: This partition is created to store the VM-support output needed for VMware support. It is not created on SD cards or USB flash drives.
- **3 (VMFS datastore)**: The available and unallocated space of the disk is formatted as VMFS5 or VMFS, depending on the ESXi version. This partition is not created on SD cards or USB flash drives.

Boot banks

Looking at the preceding partition list, you may notice that partitions 5 and 6 are named **primary boot bank** and **alternate boot bank**. These partitions are a failsafe. The ESXi system has two independent banks of memory, each of which stores a full system image. When a fresh ESXi installation is performed, partition 6 is empty.

During the system upgrade, the new version is loaded into the inactive bank of memory and the updated bank is set to be used when the ESXi reboots. If the boot process fails for any reason, the system automatically boots from the previously used bank of memory. You can also manually choose which image to use for that boot at boot time.

Scratch partition

In the partition layout, we saw that a scratch partition is created during the ESXi installation procedure. A scratch partition is a 4 GB VFAT partition used for storing temporary data, including logs, diagnostic information, and system swaps. Although a scratch partition is not required, VMware recommends that ESXi has a persistent scratch location available. If a scratch partition is not configured, `/scratch` is located on the **ramdisk** linked to `/tmp/scratch`.

Leaving the scratch partition on the ramdisk will affect the performance and the memory optimization, so it is recommended to create the partition in a suitable destination. If ESXi is installed on a destination, such as an SD card or a USB stick, the scratch partition is not created. As a result, an annoying warning message will be displayed in the UI, which advises you to set persistent storage for logs.

To configure the scratch partition, it is necessary to have a VMFS or NFS volume attached to the server to host the log files, but of course, you would have that anyway, for your VM to live on.

Perform the steps as follows:

1. Access ESXi using the **vSphere Client** and click the **Manage** item.
2. Go to the **System** tab and select **Advanced settings** to access the advanced settings.

3. In the search field, type `scratch`, then press **Enter** to find the parameter key needed to modify the partition location. The `scratchConfig.CurrentScratchLocation` key contains the current location of the scratch partition. Edit the `ScratchConfig.ConfiguredScratchLocation` key and enter a unique directory path for this host, such as `/vmfs/volumes/DatastoreUUID/DatastoreFolder`.
4. Reboot the host for the changes to take effect.

Messages from the VMkernel and other system components are useful to identify the status of the host. Potential issues are written to the log by the ESXi's syslog service, `vmsyslogd`.

To modify or configure the log location, you should perform the following steps:

1. Open the vSphere Client and select **Manage**.
2. Go to the **System** tab and click **Advanced settings** under **System**.
3. Search for the `Syslog.global.logDir` key that specifies where the logs are stored. The `/scratch` directory can be located on mounted NFS or VMFS volumes using the `[datastorename] path_to_file` syntax, where the path is relative to the root of the volume backing the datastore.
4. Click **OK** to save the configuration. Changes to the syslog options take effect immediately.

Centralized log management

As a best practice, you should consider using a centralized system for all logs in your infrastructure to cover all elements, instead of only using VMware vSphere to correlate events.

For example, let's say that at 17:01:03, your ESXi hypervisor loses half of the active paths to the storage array. If you have a centralized log management system in place, you can easily search the logs, and you might find that at the same time, your data center switched to the UPS system because of a power failure, but one of the switches went down (the power supply in the switch was not connected to the UPS system).

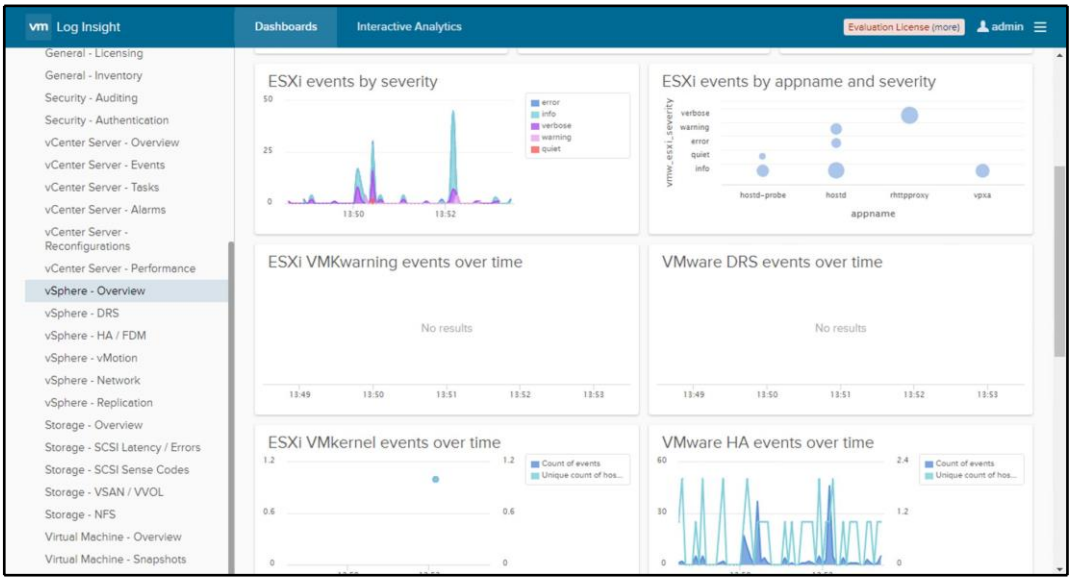
There are several products you can use for this kind of task. Some are free, and others aren't.

vRealize Log Insight

vRealize Log Insight is a product offered by VMware that has many capabilities on a single syslog server.

It has powerful add-ons that help you better understand your vSphere environment and it allows you to easily drill down through your whole environment.

vRealize Log Insight comes as an OVA appliance that you can quickly deploy to your vSphere environment. Using its powerful HTML5 interface, you can access all the logs from the entire infrastructure, as demonstrated in the following screenshot:



In the past, Log Insight was a free product that was available automatically with your vCenter server license. Recently, however, there were changes in the licensing. vRealize Log Insight 4.6 is the last available version that you can use for free.

VMware is announcing the **End of Availability** (**EoA**) of vRealize Log Insight for vCenter Server starting with the next release and all future releases of Log Insight. The current version of Log Insight, version 4.6.x, is the last release that will support the Log Insight for vCenter Server capability. The next release will not accept vCenter Server license keys for activation. For more information, please visit the following link: `https://blogs.vmware.com/vsphere/2018/07/vrealize-log-insight-for-vcenter-server-end-of-availability.html`.

Free syslog servers

There are many free syslog servers that you can easily install to your Windows or Linux environment to provide centralized log management capabilities, including the following:

- **Splunk Light**: `https://www.splunk.com/en_us/download/splunk-light.html`
- **Kiwi syslog server**: `https://www.solarwinds.com/free-tools/kiwi-free-syslog-server`
- **PRTG**: `https://www.paessler.com/free_syslog_server`

Syslog configuration

To configure the syslog server, you should perform the following configuration tasks on each ESXi hypervisor:

1. From **Navigator**, select **Manage** | **System** | **Advanced settings**
2. Search for the `Syslog.global.loghost` configuration value
3. Click **Edit Settings** and provide the **IP address or FQDN** of your syslog server

Backing up and restoring ESXi

ESXi hypervisor is the same as any other server or network device, so you need to take into account the **configuration backup**. While the installation only takes a couple of minutes, the backup contains configuration files concerning virtual switches and their configuration, shared storage (datastore configurations), multi-paths, local users and groups, and also licensing information.

This is why you should perform regular backups of your ESXi infrastructure. If something goes wrong, you can install a new ESXi hypervisor and restore the configuration to get the desired state within the blink of an eye.

Backing up and restoring ESXi using CLI

If your infrastructure is not that big, you can perform individual backups of the ESXi servers using CLI. This might not be a good approach for larger infrastructures, for which you might prefer to automate the whole task.

To perform an individual backup using CLI, perform the following steps:

1. Connect to the ESXi server using SSH
2. Run the following command: `vim-cmd hostsvc/firmware/sync_config And vim-cmd hostsvc/firmware/backup_config`
3. As an output, you will receive a URL, from which you can download the backup file

To restore an individual ESXi server, perform the following steps:

1. Upload the backup file to the ESXi hypervisor (either using the SCP protocol with WinSCP, for example, or by directly uploading the file to the datastore).
2. Connect to the ESXi server over SSH.
3. Enter **Maintenance Mode** using `vim-cmd hostsvc/maintenance_mode_enter`.
4. The backup file should be located in the `/tmp/` folder. To copy the file to the location, you can use the `cp` command: `cp /vmfs/volumes/datastore1/RESTORE/configBundle-esxi-prod-4.learnvmware.local.tgz /tmp/configBundle.tgz`.
5. Restore the configuration using `vim-cmd hostsvc/firmware/restore_config /tmp/configBundle.tgz`.
6. ESXi will automatically reboot so be prepared for this.

Backing up and restoring ESXi using PowerCLI

Sometimes, you might want to either automate the backup tasks, schedule them to run periodically, or perform them on dozens of ESXi hypervisors. Using a simple CLI approach is not the most effective way to do this, but you can use PowerCLI, an automation tool, to perform these kind of operations. We will discuss PowerCLI itself later in this chapter, so at this stage, let's take a look at how to use it.

Backing up using PowerCLI

To perform a backup of the ESXi server, perform the following steps:

1. Launch PowerCLI
2. Connect to the ESXi server using the `Connect-VIserver` command
3. Issue the `GetVMhostFirmware` command to receive the backup file

Restoring using PowerCLI

If you need to restore the ESXi server, perform the following steps:

1. Launch PowerCLI
2. Connect to the destination ESXi server using `Connect-VIserver`
3. Place the host in maintenance mode using `Set-VMhost ESXi-name -State Maintenance`
4. Restore the configuration through `Set-VMhostFirmware`

It might look as though something went wrong, and the restore operation did not happen, but this is not the case. Again, immediately after issuing this command, the host initializes the reboot cycle, so PowerCLI loses its connection to the ESXi server.

Backing up all ESXi servers within a single vCenter server

Let's have a look at a simple script that will backup all of your ESXi servers that are connected to a single vCenter Server.

The most simple version of the script will include the following commands:

```
connect-viserver -server VCSA_FQDN -user administrator@vsphere.local -
password PassW0rd

$esxi_all = get-vmhost

foreach ($esxi in $esxi_all){
 Get-VMHostFirmware -vmhost $esxi -BackupConfiguration -
DestinationPath c:\esxibackups
}
```

Now, if you check your backup folder, you will see the backups from all ESXi servers.

You can easily tweak the script to save the files in dedicated folders for each ESXi server, and store several versions of the configuration based on your preferences and backup schedule.

Configuring vCSA

When the deployment of the vCSA is complete, as part of the installation, the vSphere Web Client and the new HTML5 client are available to access the appliance. Both clients rely on the Tomcat web service to access the appliance, and no third-party software is required. As explained in *Chapter 10*, *Deployment Workflow and Component Installation*, in vSphere 6.5, login to the appliance can be done using both flash-based and HTML5-based web clients.

In your favorite browser, type the following addresses:

- **Flash-based client (vSphere Web Client)**: `https://<VCSA_IP>/vsphere-client`
- **HTML5 client (vSphere Client)**: `https://< VCSA _IP>/ui`

Basic setup using the vCenter Server Appliance Management Interface (VAMI)

Let's walk through the basic configuration that you should do on your vCenter Server instance to ensure the correct functionality. The configuration of the vCSA can be easily managed using the VAMI, which allows you to export logs, configure NTP, enable/disable SSH, and more. The same configuration can also be made using the vSphere Client.

Modifying the IP address and DNS

Although you configure the IP address and DNS during the deployment process, you can further modify the parameters through the VAMI. The steps to do this are as follows:

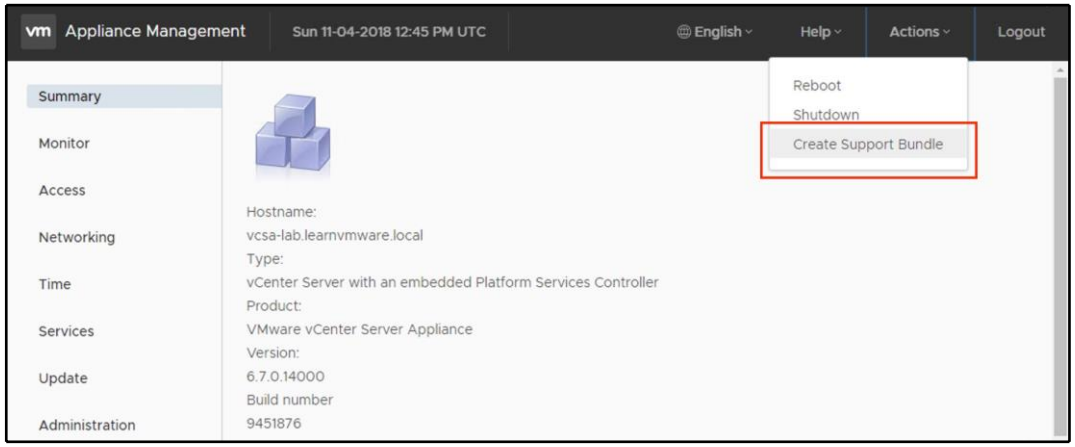
1. To modify the IP address and DNS, log in as root to the VAMI and select **Networking**.
2. Access the **Manage** tab and click the **Edit button** in the **Hostname**, **Name Servers**, and **Gateways** area and modify the network parameters.
3. Click **OK** to apply the new settings.

Exporting a support bundle

For diagnosing and troubleshooting purposes, you can export a support bundle that contains the log files of the running vCenter Server instance. The bundle can be submitted to VMware support for assistance or analyzed locally on your machine.

To export the log files, proceed as follows:

1. Log in as root to the VAMI
2. Click on the **Actions** and select a **Create Support Bundle** button to save the bundle in the `.tgz` format somewhere in your local machine, as demonstrated in the following screenshot:



Configuring time synchronization

We have already discussed the importance of having the network time synchronized. If vCenter Server is connected to an external PSC and the time is not synchronized, you may experience authentication issues. To avoid this problem, make sure you configure the same time synchronization source.

To configure time synchronization, follow these steps:

1. From the VAMI, go to the **Time** tab to configure the time zone and time synchronization.
2. In the **Time zone** area, click the **Edit** button to configure the correct time zone.
3. In the **Time Synchronization** area, click the **Edit** button and set the **Mode** field as NTP, then specify the **NTP source servers**.
4. Click **OK** to save the setup.

Changing the vCSA password

Changing the vCSA root password on a regular basis is a good way to enforce security.

The steps for changing the vCSA password are as follows:

1. From the VAMI, go to the **Administration** tab to change the password
2. Under the **Password** area, click **Change** and type the current password and enter a new one twice
3. Click **Submit** to save the changes

Licensing

VMware vSphere 6.7 is available as a 60-day, fully working trial, to give administrators the opportunity to test the product's functionalities and the services provided. When the evaluation license expires, you need to insert a valid license composed of a 25-character alphanumeric string to re-enable the functionalities in ESXi and vCenter Server in order to avoid a service outage.

The available services are strictly related to the applied license. VMware vSphere ESXi is licensed per processor, and this means that you need a valid license key for each physical CPU installed in the physical server. The license key can be used on different servers since it doesn't contain any server-related information and it's not tied to a specific hardware. You don't have any restrictions concerning physical cores or physical RAM, and the number of VMs you can run is unlimited if the proper license is applied.

VMware vSphere 6.7 comes in the following three editions:

- **vSphere Standard Edition**: This is the entry-level solution that allows for basic server consolidation.
- **vSphere Enterprise Plus Edition**: This edition offers all the features of vSphere and ensures application availability and business continuity.
- **vSphere with Operations Management Enterprise Plus Edition**: This edition offers all the features of vSphere.

Besides this, there are two Essential editions provided as full kits developed for small environments that need to save costs, where you can have up to three hosts with a maximum of two physical CPUs each (each kit includes six processor licenses and one vCenter Server Essential license):

- **Essential**: This provides basic functionality only and doesn't protect the running VM if one ESXi fails.
- **Essential Plus**: This offers services, such as vMotion or vSphere HA, to ensure business continuity and data protection.

To centralize the management of ESXi hosts and VMs and enable the available services, you need one instance of a vCenter Server. vCenter Server comes in the three following editions:

- **vCenter Server Essentials**: This is used for management of vSphere Essential kits and is integrated with the bundle.
- **vCenter Server Foundation**: This license is bundled with several vSphere bundles, especially with ROBO licenses. This vCenter server can manage a maximum of four ESXi hypervisors.
- **vCenter Server Standard**: This allows you to take advantage of all of the features available in vSphere, such as vSphere vMotion, vSphere HA, vSphere DRS, and so on.

Using vCSA is the simplest method to apply and manage the license across the infrastructure. Bear in mind that licensing is a service provided by the PSC.

If you configure the vCenter Server HA feature, you don't need to license a separate vCenter Server Standard instance for the Passive or Witness node.

To enter a new vCenter Server license, proceed with the following steps. First, you need to install the license itself. Installing the license does not mean that you assign it:

1. From the vSphere **Web Client**, go to the **Administration** and select **Licenses** under the **Licensing** option. Click on the **Add New License** button to insert a new license key.
2. Fill in the license key you want to add to the inventory. It might be vCenter Server license, an ESXi license, or any other VMware product.
3. Once the license is installed, you can browse all your available licensed products in the **Assets** tab.
4. To assign a license to a windows server, go to the **Configure** tab of your vCenter Server.
5. Under **Settings**, click **Licensing**.
6. Click **Assign License** and select the license you want to assign to this vCenter Server.

Depending on the license you have chosen, in the **Overview** tab, you will see which license is currently assigned, the license expiry date (some of the licenses may be valid only for a limited time), and which features are included in the license.

Roles and permissions

Permissions specify the privileges (the tasks a user can perform) an authenticated user or group has on a specific vCenter Server object and can be assigned at different levels of a hierarchy. For example, you can assign permissions to a cluster object or a data center object. The best practice is to assign only the required permissions, to increase the security and to have a more explicit permissions structure. The use of folders to group objects based on specific permissions makes the vSphere administration simpler.

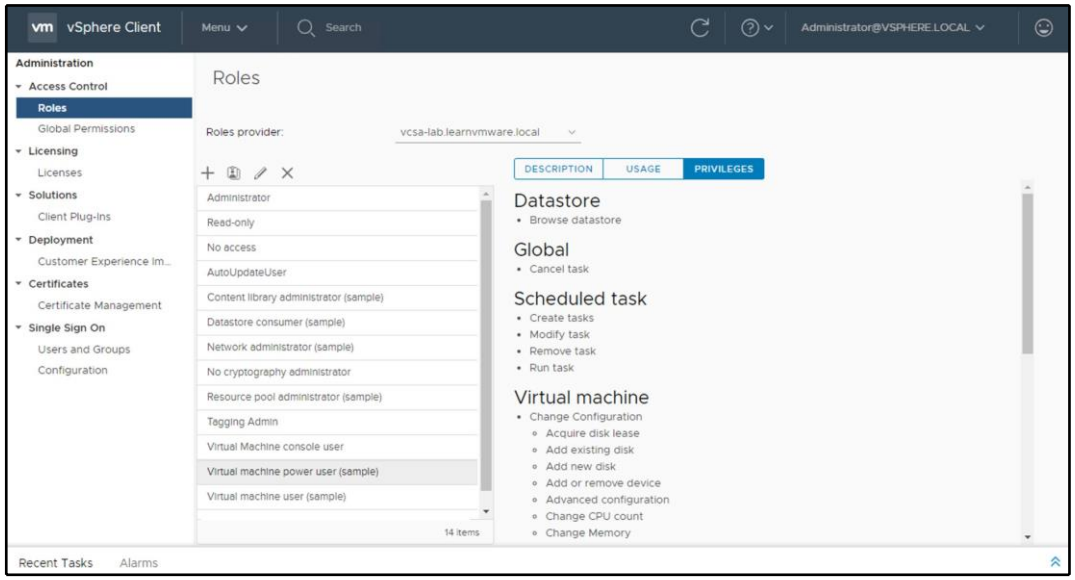
There are also global permissions that are applied to a global root object to grant the user or group privileges for all objects in all hierarchies. Use global permissions carefully, because you assign permissions to all objects in the inventory.

Roles are a set of permissions you can assign to users to perform specific tasks on inventory objects. There are some default roles predefined on vCenter Server, such as **Administrator**, **Read-only**, and **No access**, which cannot be modified. Other roles, such as network administrator, are defined as sample roles. You can create new roles or clone and modify existing roles. It is advisable to clone an existing profile instead of creating a new one to avoid potential security issues.

From vSphere 6.5, there is a new role called no cryptography administrator. This role contains the same set of permissions as the administrator role, but the user assigned with this role is not able to perform any encryption or decryption tasks. The idea is that sometimes you need to ensure that the VMs stay encrypted at all costs, but at the same time your vSphere administrators must be able to perform any configurations necessary. For this reason, no cryptography administrator role was introduced.

You can manage the vCSA roles from the **Administration** menu. Follow these steps to create or modify a new role:

1. To create a new role, select the role you want to start from and click on the clone role action icon.
2. Specify a role name, add a description (optional), then click **OK**.
3. Select the just-created role and click the edit icon to edit the role action.
4. Enable all the actions the new role should be able to perform, then click **Next**.
5. You can modify the role name and the description of the role if necessary. Click **Finish** to save the role configuration. You can navigate the **DESCRIPTION**, **USAGE**, and **PRIVILEGES** tabs to get an overview of the granted permissions and to which objects the created role has been assigned, as shown in the following screenshot:



Once a role has been defined, you need to assign the role to an authenticated user or group. Where possible, it's recommended to assign permissions to groups instead of users for better and more efficient management.

To assign a role to a user or a group, proceed with the following steps:

1. From the vSphere Client, select the object you want to assign permissions to and click on the **Permissions** tab.
2. Click the add icon button to access the wizard.
3. Specify the domain to use from the **User/Group** drop-down menu, then search for or type the user or group name you want to use. The user or group can be a member of localos, SSO domain, AD, or other identity sources.
4. From the **Role** drop-down menu, select the role you want to assign to the selected user or group. It is recommended to enable the **Propagate to children** option to also apply the role to child objects. This will not only propagate the permission to the current child, but to the newly created children as well.
5. Click **OK** to save the settings.

6. **Defined In** refers to which objects in the hierarchy the permission is configured on. Let's assume that we have created a permission somewhere within the hierarchy. If you click on any object that is a child of that object, you will see the level on which the permission was configured.

AD integration

The vCenter Server can be integrated with an external identity source, so you do not need to configure individual user accounts or groups on the vCenter Server level, but instead use a centralized database.

There are three possible integrations, as follows:

- **Active Directory** through **Integrated Authentication**
- **Active Directory** through **LDAP**
- **LDAP server**

As you can see, you can use either Active Directory as a central user and groups database or any LDAP-enabled identity source. In most environments, you will find that Active Directory through the Integrated Authentication mode is used more than the traditional LDAP approach since the configuration is much simpler. If you want to configure the Integrated Authentication mode, you must join the PSC instance or the vCSA to the AD domain. This allows the AD users to log in to vCenter Server using the Windows session authentication **Security Support Provider Interface** (**SSPI**).

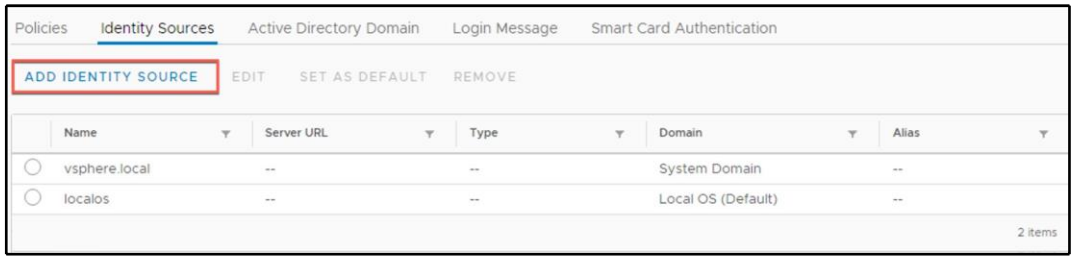
The procedure to join vCenter Server to an AD domain depends on how the vCSA and the PSC have been deployed:

- If you deployed the **vCSA with an embedded PSC**, you need to join the vCSA to the AD domain
- If you deployed the **vCSA with an external PSC**, you need to join the PSC to the AD domain

The use of a **Read-Only Domain Controller** (**RODC**) in an AD domain to join a PSC or a VCSA with an embedded PSC is not supported. Only a writable DC must be used to join the AD domain.

To join a vCSA with an embedded PSC to the AD, follow these steps:

1. Select **Administration**, **Single Sign-On**, and **Configuration**.
2. Click on the **Active Directory** tab and click **Join AD**.
3. Enter the domain to join in the **Domain** field and, optionally, the organizational unit. Specify the AD username in the UPN format (`username@domain.com`) with the privileges to join the PSC and the password. Click **OK** to confirm.
4. When the process completes, the joined domain is listed in the **Domain** field, and a new **Leave** button is displayed.
5. You need to reboot the node to enable the changes. Since this option is not available from the vSphere Client, switch to the VAMI management of the vCSA and, from **Actions**, click **Reboot**.
6. When the node has been rebooted, navigate to **Configuration** | **Identity Sources** to add the AD domain. Click to open the **ADD IDENTITY SOURCE** wizard, as demonstrated in the following screenshot:



7. Select the **Active Directory (Integrated Windows Authentication)** option and enter the joined FQDN domain name if it's not displayed automatically.
8. Select the **Use machine account** option to use the local machine account as **Service Principal Name** (**SPN**). If you expect to rename the machine, don't use this option, because it will break the authentication process. Click **OK** to confirm the specified AD domain as the new identity source.
9. In the **Identity Sources** tab, the joined AD domain is now displayed. You can assign permissions to users or group members of the AD domain.

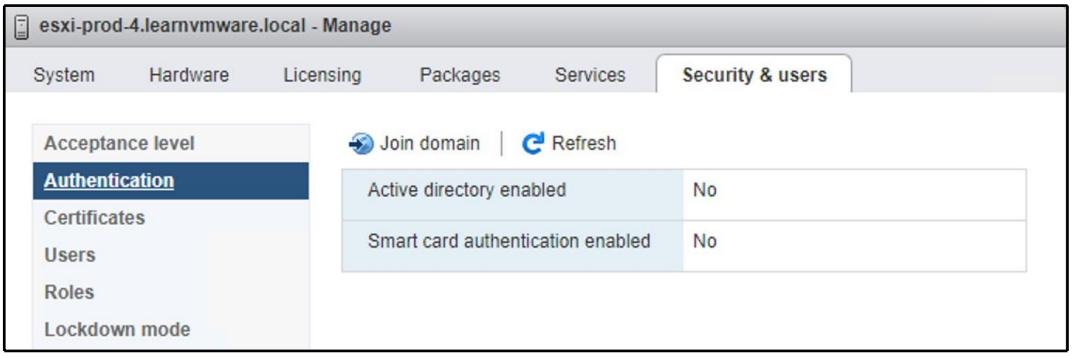
You can select the added **AD domain** and click on the **Set as Default Domain** icon to make the new identity source the default domain.

Once the integration is done, you can assign the permissions for Active Directory Users or Groups. All you need to do is select the Active Directory domain instead of the default single sign-on domain.

Configuring ESXi with AD authentication

An ESXi host can also be joined to an AD domain to allow users and groups to manage the hypervisor. When the host is added to AD, the domain group **ESX Admins** is granted full administrative access to the host, as follows:

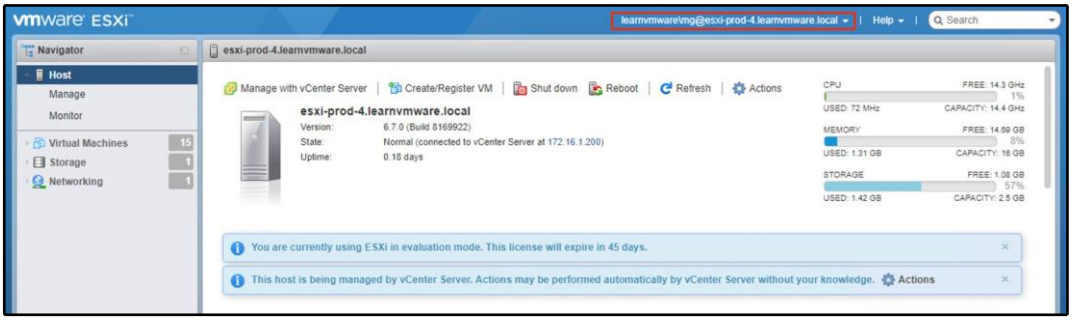
1. Log in to the host through a web console by entering the address `https://<ESXi_IP>/ui` in your favorite browser, then select the **Manage** menu.
2. Go to the **Security & users** tab and select the **Authentication** sub-menu. Click on the **Join domain** field to join the host to the domain, as shown in the following screenshot:



3. Enter the domain name and the credentials of an AD user with sufficient permissions to join computers to the domain. Click on the **Join Domain** button.

You might also change the default group that is granted an administrator role within ESXi by changing the advanced default configuration setting `Config.HostAgent.plugins.hostsvc.esxAdminsGroup`.

Once you have created the **ESX Admins** Active Directory group and assigned an Active Directory user to it, you can try to **log in** to the ESXi server using AD credentials, as shown in the following screenshot:



Of course, you can also assign individual Active Directory users or groups specific permissions on the ESXi itself. There is one caveat, however, which is that the web UI of the ESXi server is not able to browse the Active Directory itself, so you need to manually enter the username or the group name using `domain\user_or_group` as the user account.

Installing the VMware Enhanced Authentication plugin

To allow users to log in using **Integrated Windows Authentication**, you need to install the **VMware Enhanced Authentication** plugin. This plugin replaces the **Client Integration Plugin** (**CIP**) from vSphere 6.0.

In addition to Integrated Windows Authentication, the VMware Enhanced Authentication plugin also provides Windows-based smart card functionality. If you have the old CIP from a previous vSphere version installed on your machine, both plugins can coexist, and there are no conflicts.

The installation of the plugin is simple and straightforward:

1. Using your favorite browser, open the vSphere Client by typing the address of your vCenter Server, `https://<VCSA_IP>/ui`.
2. Click the **Download Enhanced Authentication** plugin option at the bottom of the page.
3. Save the plugin on your machine and run the installer.
4. When the installation has completed, refresh your browser. A **Launch Application** window may pop up in this step, asking for permission to run the **Enhanced Authentication** plugin.

If the VMware **Enhanced Authentication** plugin is installed from an Internet Explorer browser, you need to disable **Protected Mode** and enable pop-up windows.

vCSA and PSC

As seen previously, from vSphere 6.0, vCenter is composed of the PSC and vCenter Server components. The PSC is a multi-master model component that provides licensing, authentication, and certificate services. If the PSC fails, these services stop working and consequently, the entire infrastructure will no longer work.

During the design of your virtual infrastructure, you should consider the option of installing and configuring two PSCs at the site to ensure availability. You may need to connect vCenter Server to another external PSC or to point the vCSA with an embedded PSC to an external PSC.

Let's see how to configure vCenter Server to point to different PSCs.

Repointing the vCSA to another external PSC

If the external PSC fails, or if you want to distribute the load of an external PSC, you can configure the vCenter Server instance to point to a different PSC in the same domain and site.

The steps are as follows:

1. SSH the vCenter Server instance using the root credentials and enable the shell.
2. To repoint vCenter Server, run the following command:

```
cmsso-util repoint --repoint-psc psc_fqdn
```

   Here, `psc_fqdn` is the FQDN (the value is case-sensitive) or the static IP address of the external PSC.

3. Using the vSphere Client, log in to the vCenter Server instance to verify that the instance is running and that you can manage it. The vCenter Server instance is now registered with the new PSC.

For more information visit the official VMware **Knowledge Base** (**KB**): `https://kb.vmware.com/s/article/2113917`.

Pointing the vCSA with an embedded PSC to an external PSC

If the vCenter Server instance has been deployed with an **embedded PSC** and you want additional vCenter Server instances in your SSO domain, you can modify the vCenter Server instance configuration to point to an external PSC. This configuration is a **one-way process** only, and it's not possible to switch back to the previous configuration with an embedded PSC.

Before proceeding, take snapshots of both the vCenter Server with the embedded PSC and the external PSC, so that you can go back if anything goes wrong during configuration.

To point the vCSA with an embedded PSC to an external PSC, you should perform the following steps:

1. SSH the vCenter Server with the embedded PSC using the root credentials and enable the shell.
2. Verify that all services are running in the PSC using the following command:

```
service-control --status --all
```

The services that must be running are as follows:

- VMware License Service
- VMware Identity Management Service
- VMware Security Token Service
- VMware Certificate Service
- VMware Directory Service

3. To reconfigure the vCenter Server, use the following command:

```
cmsso-util reconfigure --repoint-psc psc_fqdn --username
username --domain-name domain --passwd password
```

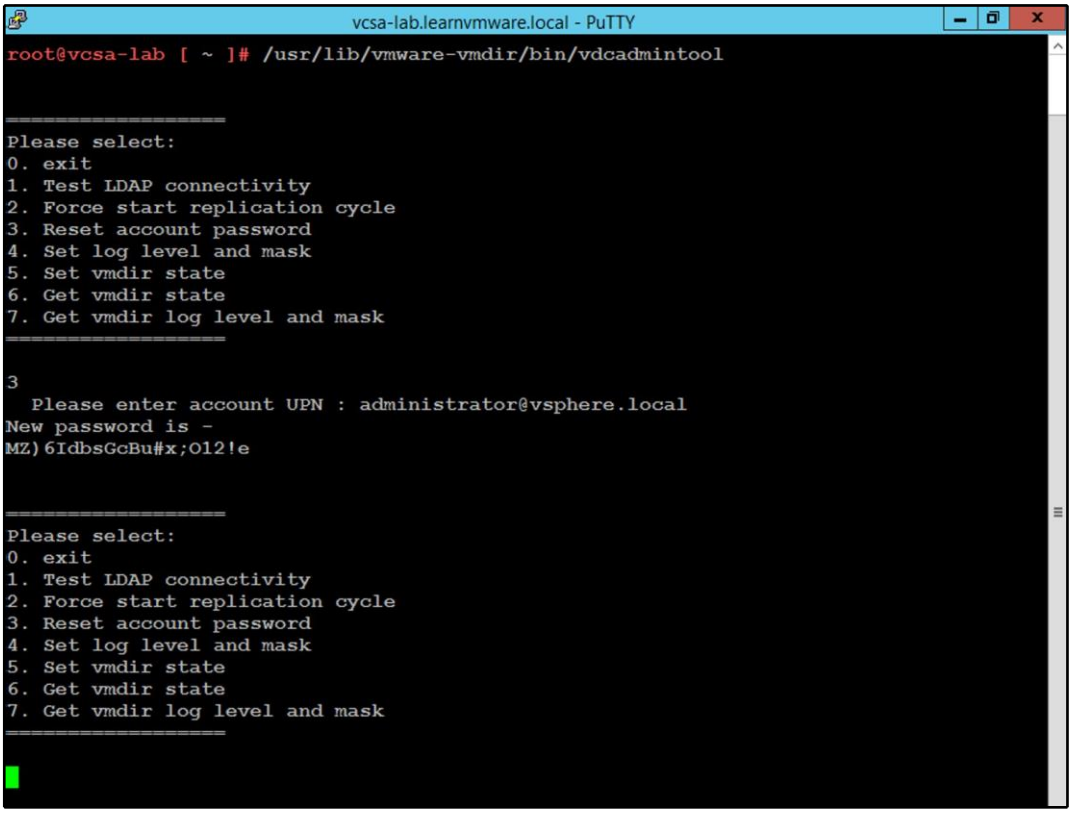
4. Using the vSphere Client, log in to the vCenter Server instance to verify that the instance is running, and that you can manage it. If the procedure has completed successfully, the vCenter Server with the embedded PSC is now demoted and redirected to the external PSC.

Resetting the SSO password

There are some situations in which you need to reset the SSO password to recover access to the PSC due to a forgotten password.

Follow this procedure to reset the SSO password:

1. SSH the PSC or vCenter Server with the embedded PSC appliance as a root user and enable the shell with the `shell.set --enabled true` command, then type `shell`. Press *Enter*.
2. Run the `/usr/lib/vmware-vmdir/bin/vdcadmintool` command to load the console and manage the password reset.
3. Select **option 3**. Reset the account password and, when prompted, enter the account UPN (such as `administrator@vsphere.local`), as shown in the following screenshot:



4. A new password is generated. Use this password to log in to the system with the user for which you want to reset the password (for example, `administrator@vsphere.local`).
5. Once you are successfully logged in to the vSphere Client using the password generated by the system, click **Users and Groups** under the **Single Sign-On** menu, and then select the **Users** tab.
6. Select the account used to log in and click the edit icon to set a **new password**. Enter the new password twice and click **OK** to confirm the change.

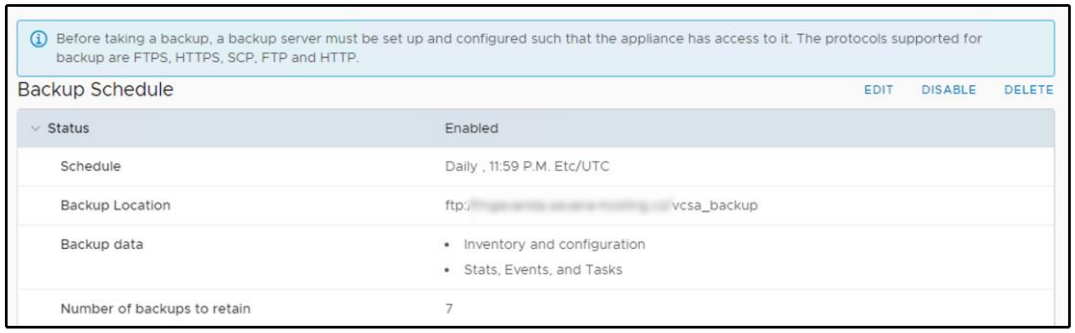
Exporting and importing the vCSA configuration

As we have already described, backing up your infrastructure should be a routine task. This should also be the case for the vCSA. In the past, it was possible to configure a one-time backup of the vCSA using the VAMI, but for periodical backups, you have to create a custom shell script. This is not required anymore; with VMware vSphere 6.7, you can even configure a backup schedule through the VAMI.

The vCSA backup procedure

To configure a backup of the vCSA, follow these steps:

1. Connect to the vCSA VAMI
2. Select **Backup** from the left-hand menu
3. Click **Configure** to configure a backup schedule
4. You have the option to configure several parameters of the backup schedule, as follows:
    - **Backup location**: Points to the `ftp/scp/nfs` location that will be used for backups
    - **Backup server credentials**: The login used to access the backup server
    - **Schedule**: Indicates when the backup should be performed
    - **Encryption**: You might choose to encrypt the backup if the data is stored in a non-secure location
    - **Number of backups**: How many backups should be retained
    - **Data**: Which tables should be saved in the backup
5. Once the backup is configured, it will start to perform the backups based on your schedule, as demonstrated in the following screenshot:



In the **Activity** window, you will see all backups that were performed and some additional information about them. From here, you have the option to take a manual backup outside of the configured backup schedule.

You can also explore the backup server location. Depending on the backup server configuration, you might use, for example WinSCP, to browse the backup server (for SCP or FTP backup-type locations).

As you can see, at this time, I have only one backup on the backup server. We can see that it was a manual backup (as indicated by the `M_` in the name) with the timestamp when the backup was taken.

vCSA restoration procedure

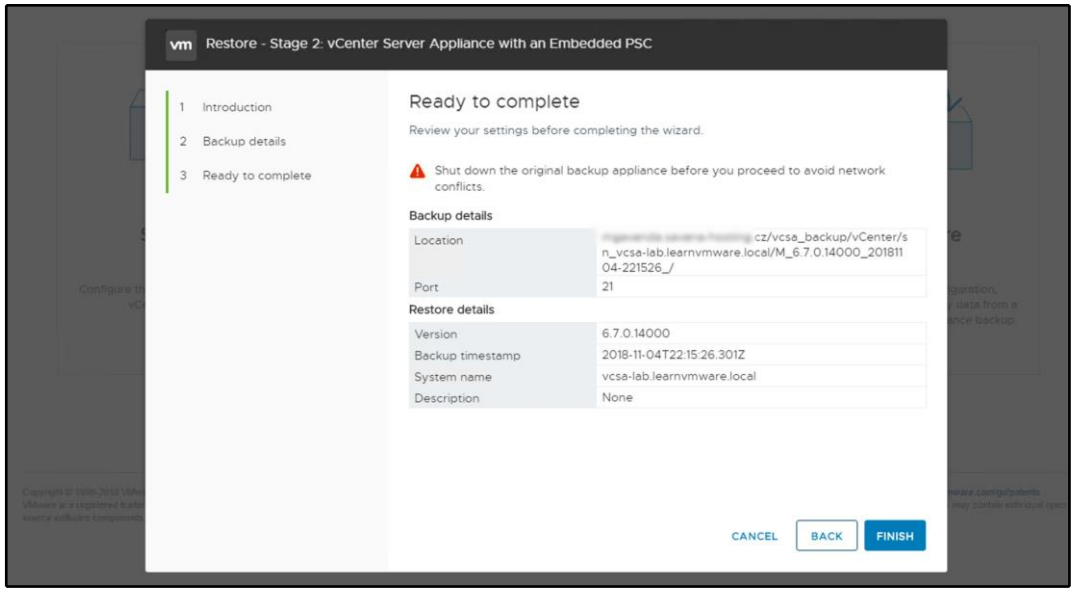
To restore your vCSA from a backup, you need the installation image of the vCenter Server containing the same vCSA version that you are about to restore. The restoration process installs a fresh new vCSA server and then performs a restoration from the backup location. It does not repair the existing vCSA, as follows:

1. Based on your operating system, launch the installation wizard.
2. From the main menu, select **Restore**.
3. The wizard follows the same steps as the installation itself. In the third step, you need to specify the backup location. You do not need to provide the full path to the backup. A first-level backup folder will do the trick.
4. Once you click **Next**, the backup browser will appear. Select the folder containing the backup you want to restore.

5. Once you have selected the backup folder, you can confirm the selection in the backup review.
6. The next steps are the same as for the clean installation. Select the deployment target, configure the VM name and the root password, and select the deployment size, and the datastore where the vCSA will be deployed.
7. You do not need to configure the network settings since the information from the backup file will automatically populate them.
8. In the last step, you may review all the settings and, once ready, click **Finish**.

Once the first stage of the installation is complete, you can continue with the second stage. For this stage, you need to provide a password. If you have selected **Encrypted Backup**, everything else is already preconfigured.

If the original vCSA is still available, shut it down as the wizard suggest to avoid any network conflicts, as shown in the following screenshot:



Once the restoration is done, you will be able to access your original vCSA at the configured FQDN with all the hosts in the inventory and all the performance metrics until the time that the backup is available.

Please note that if you have vCSA HA configured, it won't be configured after the restoration and you will need to deploy passive and witness vCSA again using the vCSA HA configuration wizard.

Managing data centers, clusters, and hosts

vCenter Server is a core component of the infrastructure that allows a centralized administration of hosts and VMs for your environment. To ensure the maximum efficiency of the infrastructure, you need to consider how to administer VMs and their resource demands.

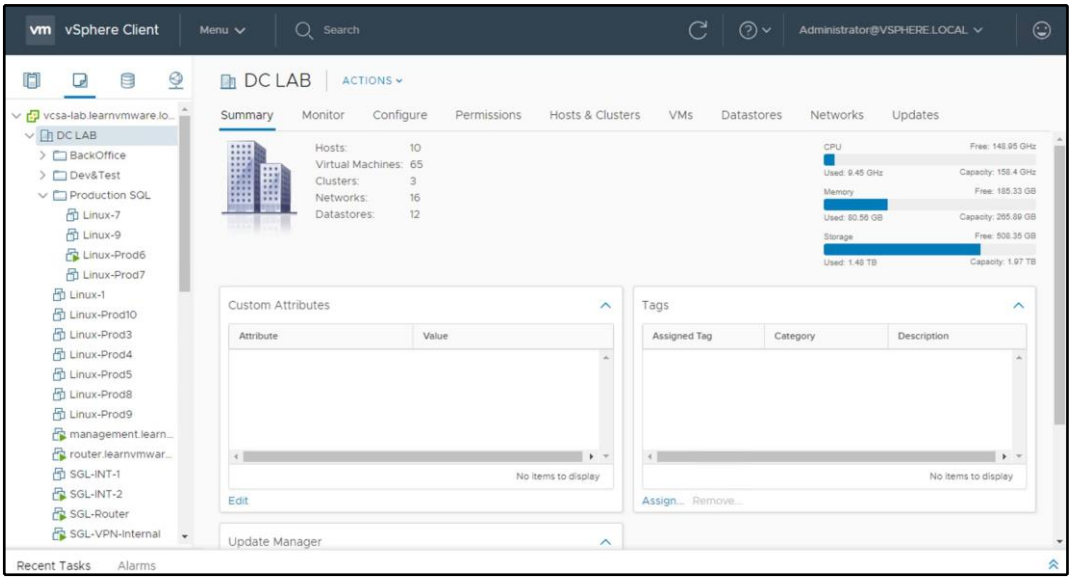
For an optimal organization of the inventory, you need to create some virtual objects in the vCenter Server to define a logical structure. The organization of the inventory requires the following tasks to be performed:

1. Create data center(s)
2. Create cluster(s)
3. Add hosts to the cluster
4. Build a logical infrastructure using folders
5. Set up networking (vSS and vDS)
6. Configure the storage system (the datastore and the datastore cluster)
7. Create clusters (resource consolidation, vSphere HA, and vSphere DRS)
8. Create a resource pool (flexible management of resources)

In vCenter Server, there are four main views available to manage the inventory:

- **Hosts and clusters**: Clusters, hosts, resource pools, and VMs. From this view, you can manage the resource allocations of VMs and their locations.
- **VMs and templates**: Folders, VMs, and templates. This view can be used to group VMs in a logical structure (by role, location, department, and so on) using folders. You can also manage the templates from which you can deploy new VMs.

- **Storage**: Datastore and datastore cluster. From this view, you have an overview of installed datastores in your virtual infrastructure, regardless of the data center membership. You can configure and manage all device configurations, including the datastore clusters.
- **Networking**: **vSphere standard switch** (**vSwitch**) and **vSphere distributed switch** (**vDS**). The setup of services, such as vMotion, vSphere FT, vSAN, and so on are managed from this view.



Creating a data center

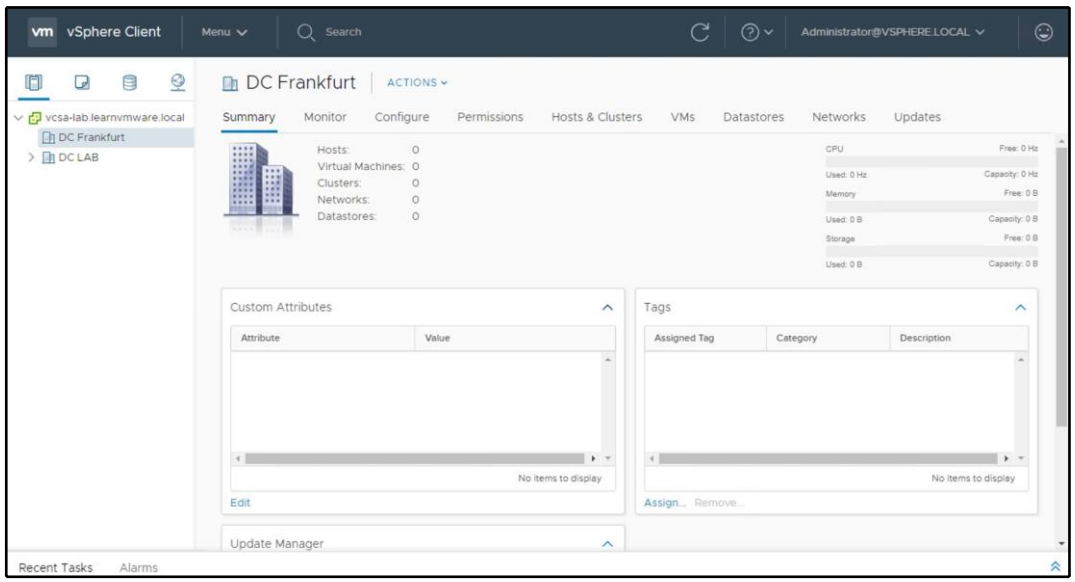
The vCenter Server is composed of a data center object that acts as a core container for all other objects. To add hosts and VMs to the vCenter Server, at least one data center object must be created.

You can configure more than one data center per vCenter Server since multiple data center objects within a single instance are supported. Data center objects are shared among the four views, allowing for a better organization of the view based on the corporate policies, therefore simplifying management.

A data center object usually represents either a physical location in your infrastructure, such as the name of the physical data center, a physical location in the data center, or even an individual data center room.

To create the data center object, proceed with the following steps:

1. From the vSphere Client, right-click on the connected vCenter Server and select the **New Datacenter** option.
2. Enter a name in the **Name** field and click **OK** to create the data center object. Once the data center has been created, you can add the hosts to the inventory, as demonstrated in the following screenshot:



Adding a host to the vCenter Server

Since the vSphere environment strongly relies on DNS, before configuring the vCenter Server, make sure that the name resolution is working correctly in your network.

Verify that the vCenter Server can resolve ESXi hostnames added to the inventory and that the added ESXi hosts can resolve the vCenter Server hostnames used to manage them. Ensure that all the hypervisors added to vCenter can resolve the hostnames of other ESXi hosts.

To add ESXi hosts to vCenter Server, follow these steps:

1. From the vSphere Client, log in to the vCenter Server.
2. In the **Hosts and Clusters** view, right-click on the configured data center object and select the **Add Host** option.
3. Enter the **ESXi hostname or IP address** and click **Next**. In this step, it is suggested to use **FQDN** instead of the IP address.
4. Enter the root credentials and click **Next**. When prompted, click **Yes** to trust the host and to accept the host's certificate.
5. On the host summary page, click **Next** to continue. On this page, information related to the added host is displayed.
6. Select an **available license** to assign to the host. A 60-day evaluation license can be assigned if no license keys have been entered previously. Click **Next**. If you haven't purchased a vSphere Enterprise Plus license, using the 60-day evaluation license, you have all the vSphere features available to complete the configuration of the environment, taking advantage of some automation and functionalities included in the Enterprise Plus license only. For example, you could take advantage of the Storage DRS feature to optimize the performance and resources across different storage devices in your vSphere environment.
7. Configure **Lockdown** mode. By default, Lockdown mode is set as **Disabled**. Select **Normal** to manage the host through the local console or the vCenter Server, or **Strict** to allow access to the host only through the vCenter Server, stopping the DCUI service. You might use Lockdown mode for hardening access to your ESXi servers. Select the option you want to use and click **Next**.
8. Specify the location to which you want to move existing VMs running in the selected host and click **Next**.
9. Review your settings and click **Finish** to add the host to the vCenter Server. Repeat the same procedure to add all other required hosts to this vCenter Server instance.

You can add a host to the data center object itself. In this case, the ESXi hypervisor will be considered as a standalone host, or you can add the ESXi host directly to the cluster as a member of the cluster. You can quickly move your ESXi hypervisors between different clusters or data centers; it's not a fixed configuration.

When you enter the root password to add the host to the vCenter, the password is used to establish a connection with the host and to install the vCenter agent. The process sets different credentials that maintain the communication and authentication between ESXi and vCenter, even if ESXi's root password is changed.

Disconnecting a host from vCenter Server

Once ESXi is connected to vCenter Server, you can always disconnect or remove the host later on. It's important to understand that disconnecting the host from vCenter Server is different to removing the host. Disconnecting a managed host from vCenter Server doesn't remove ESXi from the vCenter Inventory as well as the VM registered in the host. When the managed host is disconnected, vCenter Server suspends monitoring and management activities for that host. If you disconnect a host and then connect it again, all the performance metrics will be kept.

It's a different story if you remove the host from the vCenter Server. Removing a managed host from the vCenter Server means the host and its VM are removed from the vCenter Inventory. If you remove a host from the inventory and then add it again, it will be considered a new object and no historical performance metrics will be available.

To disconnect a managed host from the vCenter Server, proceed with the following steps:

1. From the vSphere Client, log in to the vCenter Server that manages the host to disconnect.
2. Right-click the managed host, select the **Connection** | **Disconnect** option and click **OK** to confirm. Once the host is disconnected from the vCenter Server, in the inventory view, the ESXi and all the VMs associated are marked as disconnected.

To reconnect the host, you operate from the vCenter Server, as follows:

1. Right-click on the disconnected host and select the **Connection** | **Connect** option, then click **OK** to confirm.
2. Click **Next** in the **Name** and **Location** tab, then enter the root credentials of the host to reconnect. Click **Next**.
3. In the host summary, click **Next**, then specify where to locate the VM in the **VM Location** tab. Click **Next** to continue.

4. Click **Finish** to reconnect the host. When the procedure has completed, the disconnect label is removed from the host and its VM. The host is available to the vSphere environment again.

Removing a host from vCenter Server

Removing a host from vCenter Server stops all the vCenter Server monitoring and managing activities. You should remove the managed host while still connected to remove the vCenter agent as well.

To remove a managed host, follow this procedure:

1. From the vSphere Client, log in to the vCenter Server that manages the host to remove.
2. Power off all running VMs, right-click on the host, and select **Maintenance Mode** | **Enter Maintenance Mode**. Click **Yes** to confirm.
3. Right-click on the host once again and select the **All vCenter Actions** | **Remove from Inventory** option. Click **Yes** to confirm the removal. The host and its VM are removed from the vCenter Inventory; the license assigned to the host is removed from the vCenter list and retained by the host.

When the host has been removed, the vCenter Server is no longer able to manage the host. To access the VM, you need to access the host directly.

Creating a cluster

A vSphere cluster is a configuration that manages the added hosts pooling the available resources. Once the cluster has been created, you can move the hosts to the cluster. When the hosts are added to the cluster, the cluster manages the available resources and allows you to enable the vSphere HA, vSphere DRS, and vSphere FT features that are only available with clusters. These features will be covered in *Chapter 15*, *Availability and Disaster Recovery* .

To create a cluster, proceed as follows:

1. From the vSphere Client, log in to the vCenter Server.
2. In the **Hosts and Clusters** view, right-click on the configured data center object and select the **New Cluster** option.

3. Enter the name of the cluster and select the features you want to enable. Click **OK** to create the cluster. To better understand the available features displayed in the wizard, the HA feature provides business continuity, while DRS is used to balance the workload across the hosts.

Enhanced vMotion Compatibility (**EVC**) is a feature that allows VMs to vMotion across hosts with different processors in the same cluster. The caveat is that all processors must be from the same vendor (Intel or AMD) since a mixed cluster is not supported. Pay attention when you install a new ESXi server in the same cluster.

To add hosts to the created cluster, the easiest way is to drag and drop the ESXi hosts into the cluster. Alternatively, you can right-click on the hosts, select the **Move To** option, select the target cluster, then click **OK**. You can also use scripts to automate the process of adding hosts to the cluster. If you are prompted about resource pool management, leave the default option and click **Yes**.

Removing a host from a cluster

When you remove a managed host from a cluster, the cluster loses the resources provided by the removed host, reducing the total capacity. All historical data remains in the vCenter Server database. Before removing a host from a cluster, make sure that the cluster has enough resources to provide to the workloads to avoid performance issues or service disruption.

If the vSphere DRS feature is not enabled in the cluster, make sure to migrate all running VMs to a new host using vMotion before putting the host in **Maintenance Mode**. If not migrated, powered off, or suspended, VMs will remain associated with the removed host:

Follow these steps:

1. From the vSphere Client, right-click on the host you want to remove from a cluster.
2. Right-click the host to remove and select the **Maintenance Mode | Enter Maintenance Mode** option, then click **OK** to proceed. If DRS is enabled, powered-off (you need to enable the option), and running VMs are migrated to other hosts in the cluster.
3. When the host enters **Maintenance Mode**, the host icon changes. Right-click on the host and select the **Move To...** option.
4. Select the destination (data center, folder, or a different cluster) of where you want to move the host to and click **OK**.

5. When the host has moved off the cluster, right-click on the host and select **Maintenance Mode** | **Exit Maintenance Mode**.

If you want to move a host in a cluster from one vCenter Server to another, you can disconnect the host and move it without putting the host in **Maintenance Mode**.

Managing hosts

vCenter Server is a core component of VMware vSphere that centralizes host administration, offering some powerful features that simplify the management process. vCenter Server provides a single pane of glass for your environment and allows access to the installed hosts and their configurations.

To access the ESXi management area, select a host from the vSphere Client. If you navigate from the available tabs, you can access the different configuration areas to set up the host matching the business requirements.

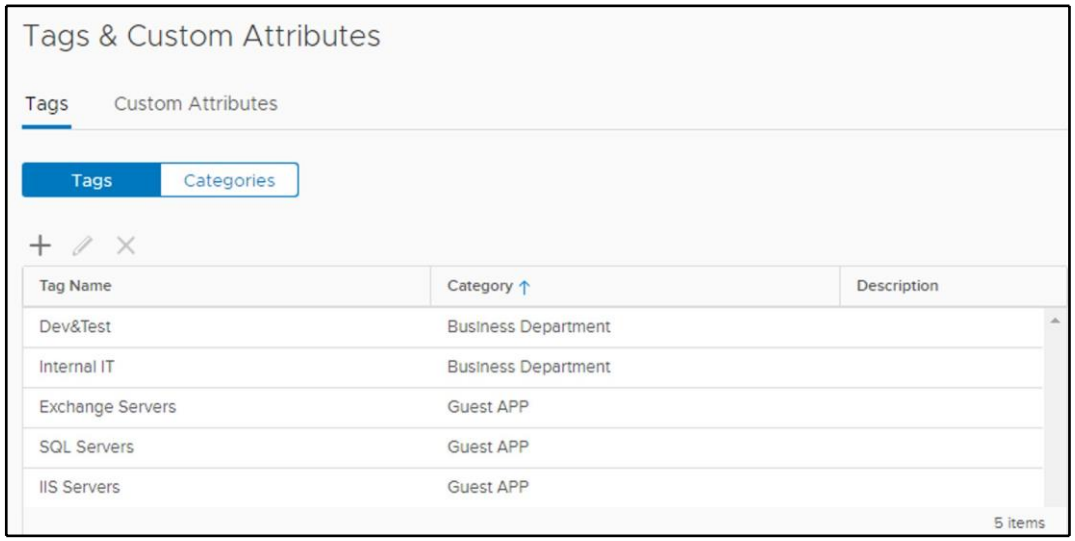
Let's have a look at the main areas:

- **Summary**: This displays information related to the ESXi, such as the resources in use, the tags, the global configuration, and more.
- **Monitor**: From this tab, you can track issues and alarms related to configuration problems and check the host's performance information, tasks, and events related to the selected host and the hardware health, to keep the status of the hardware components under control.
- **Configure**: From this tab, you can modify the host configuration. Storage configuration (such as storage adapters, and storage devices), network settings (such as vSwitches, VMkernel adapters, and TCP/IP), system components (such as host profile, firewall, security profile, and more), and hardware changes can be done in this section.
- **Permissions**: This tab is used to add permissions that specify users and roles.
- **VMs**: This shows the list of VMs and VM templates registered in the selected host. Double-click on an object to access its configuration area.
- **Datastores**: This displays the list of attached storage, showing details such as status, type, storage capacity, and free space.

- **Networks**: This displays a list of virtual switches and distributed virtual switches configured in the selected host.
- **Updates**: This displays a list of attached baselines from the **Update Manager** and the object compliance with the baselines.

Using tags

A tag is a label that you can apply to vCenter Server objects (such as datastores, VMs, and hosts) to simplify searches, allowing for a better sorting process. When a tag is created, it must be assigned to a category that groups related tags. A category also specifies whether you can assign one or multiple tags to an object. The creation and management of tags and categories are done through an intuitive configuration area, which can be reached by going to the menu, and then **Tags & Custom Attributes**, as follows:



For instance, you can use tags to classify your application type or the business department that is responsible for a VM.

To assign a tag, right-click on an object in the vCenter Inventory and select **Tags & Custom Attributes** | **Assign Tag**.

Some software backup solutions make use of tags to group VMs with the same backup policy (for example, RTO), making the administration process more straightforward.

Tasks

Tasks are activities performed by the system that occur on an object of the vCenter Inventory (to power on or power off a VM, for example) and can be executed in real-time or be scheduled.

The task list can be viewed in the vSphere Client by selecting the **Tasks** option from the menu. The list displays all tasks that occurred to a specific object, detailing information, such as the target, status, initiator, and more. By default, tasks listed for a single object also include tasks executed on the child objects. The list can be filtered by typing the keywords in the search field on the right.

Scheduling tasks

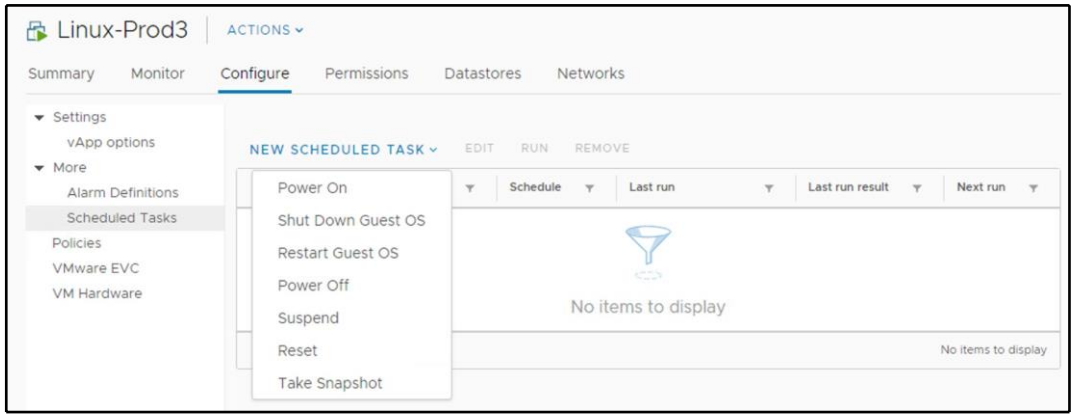
Tasks can be scheduled to run at specific times or recurring intervals. A task cannot be scheduled to run on multiple objects. The available tasks you can schedule from the vSphere Web Client are the following:

- Add hosts and check the compliance of a profile
- Change the power state, clone, create, deploy, migrate, make a snapshot, or edit the resources of VMs
- Change the cluster power settings
- Scan for updates
- Remediate the object

Tasks not included in the list can be scheduled using the vSphere API. To schedule a task, proceed as follows:

1. From the vSphere Web Client, select the object on which you want to schedule a task and select **Monitor** | **Tasks & Events**. Click **Scheduled Tasks** to create a new schedule.
2. From the **Schedule**, in the **New Scheduled Task** drop-down menu, select the task to schedule (available tasks depend on the selected object).

3. Configure task-related options and the scheduling settings. Enter an email address to be notified when the task is complete and click **OK** to save the task schedule, as demonstrated in the following screenshot:



Managing host profiles

Host profiles are a feature of vSphere that allow you to include the ESXi configuration in a profile (template) to ensure that all hosts installed across the infrastructure have the same configuration and are compliant with the setup policy you have in your organization.

Generally, after completing the deployment of an ESXi host, there are several settings you should configure to ensure the host's services match your infrastructure:

- **Network configuration**: Creating VMkernels and VM port groups, assigning IPs to VMkernels, setting up NIC-teaming, and mor
- **Storage configuration**: Configuring software iSCSI adapters, port bindings, and CHAP.
- **Time synchronization**: Configuring and enabling the NTP service
- **Enable services:** Enabling services such as SSH and shell
- **Firewall**: Opening specific ports that are required by some services

If you have just a few hosts to install, you can quickly and easily use the interactive ESXi installation method and once, completed, manually set up the required host's parameters. If the environment to build is large and you have 100-1,000 hosts to set up, manually performing the configuration for each hypervisor is a tedious and time-consuming task, and human error can occur at any moment: incorrect IP addresses assigned, wrong NIC to a VMkernel port group mapping, and so on.

Using host profiles mean we can avoid these kind of configuration errors. You profile a host by creating a template containing the configuration extracted from a reference host, and then you apply this template to any host of the infrastructure to ensure consistency. Once created, the host profile can be edited to change, enable, or disable properties. If a host profile is applied to a cluster, all the member hosts are affected, ensuring a consistent configuration.

Host profiles can also be used together with the *Auto Deploy* feature (*Auto Deploy* was covered in *Chapter 10*, *Deployment Workflow and Component Installation*) to automate the provisioning process fully.

The overall process can be summarized as follows:

1. Set up and configure the reference host. Because the configuration will be saved to the host profile, make sure the ESXi setup is correct and verified.
2. Create the master host profile, extracting the configuration from the reference host.
3. Attach the created host profile to a host or cluster to apply the standard configuration.
4. Check the compliance of processed hosts to the host profile to ensure they all have the same configuration.
5. Remediate the host to apply the settings. The ESXi host attached to the selected host profile modifies its configuration only at this stage.

To create a host profile, perform the following steps:

1. From the vSphere Web Client, right-click on the hypervisor used as a reference host and select **Host Profiles** | **Extract Host Profile**.
2. Enter a profile name and a description. The description is useful to identify the scope of the profile but is optional. Click **Next** when done.
3. Click **Finish** to start the host profile creation. When the process has completed, the created profiles can be found in the **Home** | **Policies & Profiles** | **Host Profiles** area of the vSphere Web Client.

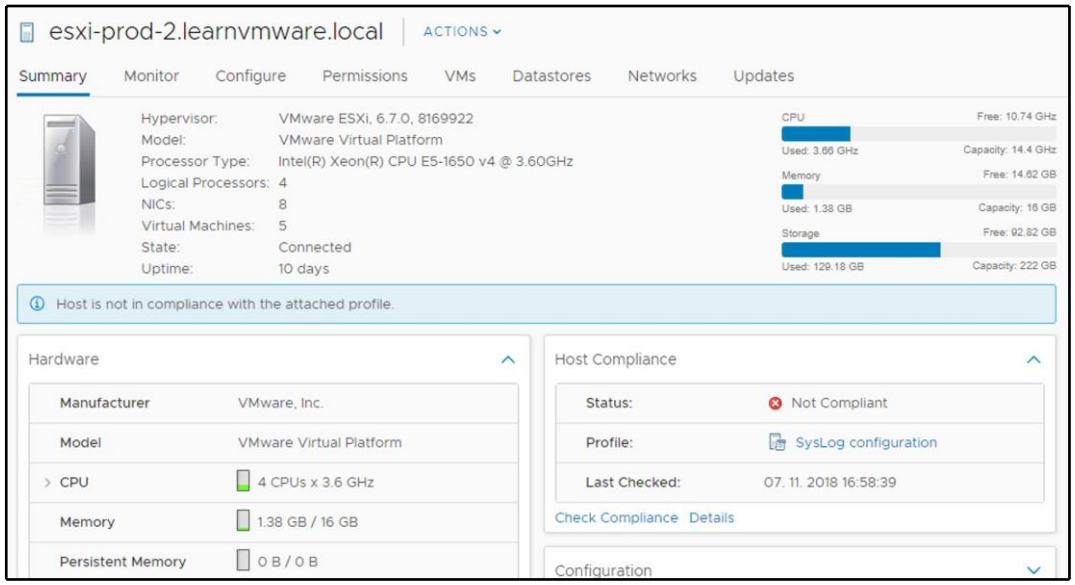
To apply settings saved in the profile to a host or cluster, you need to attach the created host profile. To attach the host profile, proceed as follows:

1. From the vSphere Web Client, right-click on the host to process and select the **Host Profiles** | **Attach Host Profile** option.
2. Select the host profile to attach and click **OK**. At this stage, you can customize the host (for example, configure the IP address and DNS server) or enable the **Skip Host Customization** option to avoid host customization during the process.

Once a host profile is attached to a host, the configuration is not automatically applied, but you must perform a compliance check first to compare the current host configuration with the configuration stored in the profile.

To run the compliance check, follow these steps:

1. Right-click on the host to check and select **Host Profiles** | **Check Host Profile Compliance**. If the checked host is found to be non-compliant, a warning message is displayed in the **Summary** tab, as demonstrated in the following screenshot:



2. If you click on **Details**, you will see which configurations are done differently compared to the host profile. In this case, the ESXi server, `esxi-prod-2`, does not have the syslog server configured (advanced configuration option `Syslog.global.logHost`).
3. You can remediate the host to match the Host Profile (the configuration from the host profile will be applied to the host) using the **Remediate** button.

Host profiles can be modified to change some settings by editing the desired profile. To modify a host profile, proceed as follows:

1. From the vSphere Web Client, go to **Home** | **Profiles & Policies** | **Host Profiles**
2. Select the profile you want to modify from the list and click on the **Edit Host Profile**

Automating tasks with scripts

The administration of the vSphere environment often requires you to perform repetitive tasks that can be time-consuming, involving the same activities to be done for each component of the infrastructure. Examples of these kinds of activities include migrating VMs or deploying new VMs from a template.

The chance to automate some tasks will allow you to optimize your time, improving efficiency and ensuring consistency. Manually modifying the configuration of thousands VMs, for example, will require a lot of time, with the risk of missing some steps or making some errors. Automation can perform the same tasks in seconds with no errors and ensure consistency within the network, reducing the workload of IT staff.

VMware offers some tools to automate tasks, such as PowerCLI, vCLI, **vRealize Orchestrator** (**vRO**), and the vSphere Web Services SDK. **vSphere Management Assistant** (**vMA**) has been deprecated, and version 6.5 is the final release.

Perhaps the most popular tool for system administrators is PowerCLI, but of course, the optimal solution is only what is suitable for your environment and needs.

Automating with PowerCLI

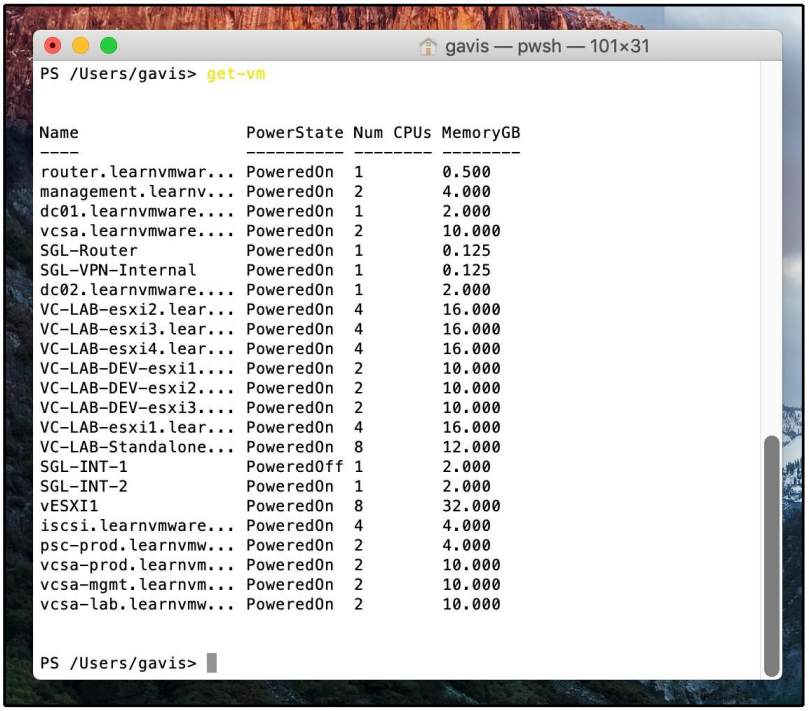
The most common automation tool provided by vSphere is PowerCLI, a command-line and scripting tool built on Windows PowerShell that provides cmdlets used for managing and automating vSphere and other VMware products.

Today, two versions of PowerShell and PowerCLI exist:

- **PowerCLI 6.5**: This version is based on the standard **Windows PowerShell**
- **PowerCLI 10.0**: This version is based on **PowerShell Core**, which is a multi-platform implementation of PowerShell

PowerShell Core can be installed on Windows, Linux, and macOS and it is the recommended version of PowerShell to use. Microsoft is going to deprecate old Windows PowerShell, so you should switch to PowerCLI 10 if you have not done so already. As mentioned, PowerShell Core can be installed even on macOS. You can use PowerCLI directly from your macOS, and you do not need to use Windows to jump hosts anymore!

The following screenshot shows PowerCLI running directly on macOS X:



If you have not install PowerShell Core yet, check the official manual of PowerShell Core at the following link: `https://docs.microsoft.com/en-us/powershell/scripting/setup/installing-powershell?view=powershell-6`.

Once the PowerShell Core is installed, you can easily install PowerCLI directly from the PowerShell Core.

The installation process of PowerCLI 10 has been simplified and requires us to run a command from the PowerShell console.
To install PowerCLI, follow this procedure:

1. Open the PowerShell console and run the following command:

```
Install-Module -Name VMware.PowerCLI
```

2. To see the installed modules, run the following command:

```
Get-Module vmware* -listavailable
```

3. Before you start to use PowerCLI, don't forget to change the default behavior of PowerShell Core if you are using self-signed certificates. By default, they are not trusted:

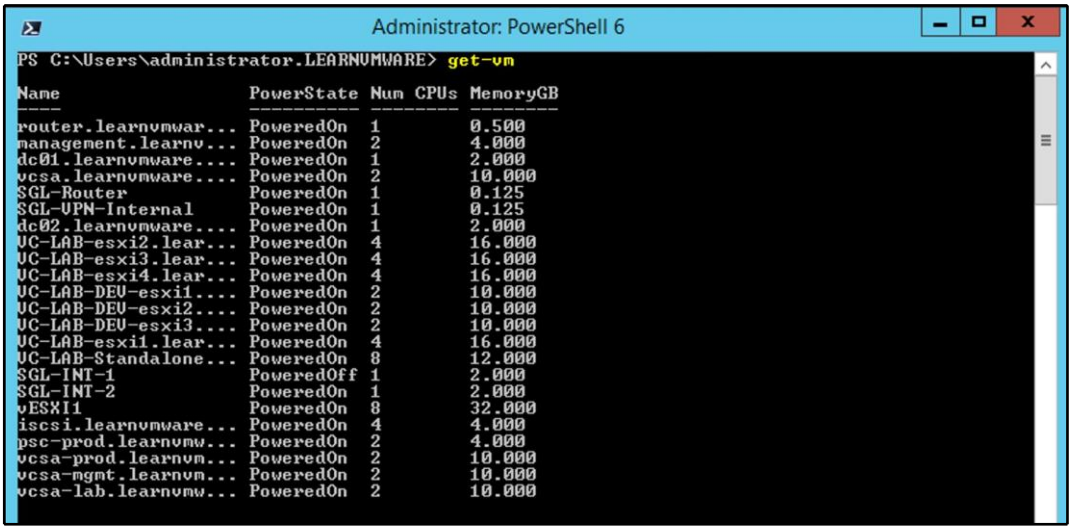
```
Set-PowerCLIConfiguration -InvalidCertificateAction Ignore
```

To try PowerCLI, open the PowerShell Core console where the VMware PowerCLI module has been installed. From the console, connect the vCenter Server instance to query, then run the following command (enter the login credentials when prompted):

```
Connect-VIserver –server VCSA_fqdn
```

To get a list of VMs running in the selected vCenter Server instance, enter the following command:

```
Get-vm
```



Remembering all PowerCLI commands and the correct syntax is not easy for most people, and the documentation is not always available to help. A useful cmdlet is available in PowerShell that provides information about a specific command is `Get-help`. To find the correct syntax to use with a specific cmdlet, `Get-help` helps you to find the information you need. For example, to find which parameters can be used with the `Get-VM` cmdlet to retrieve the list of running VMs, you can enter the following:

```
Get-help Get-VM
```

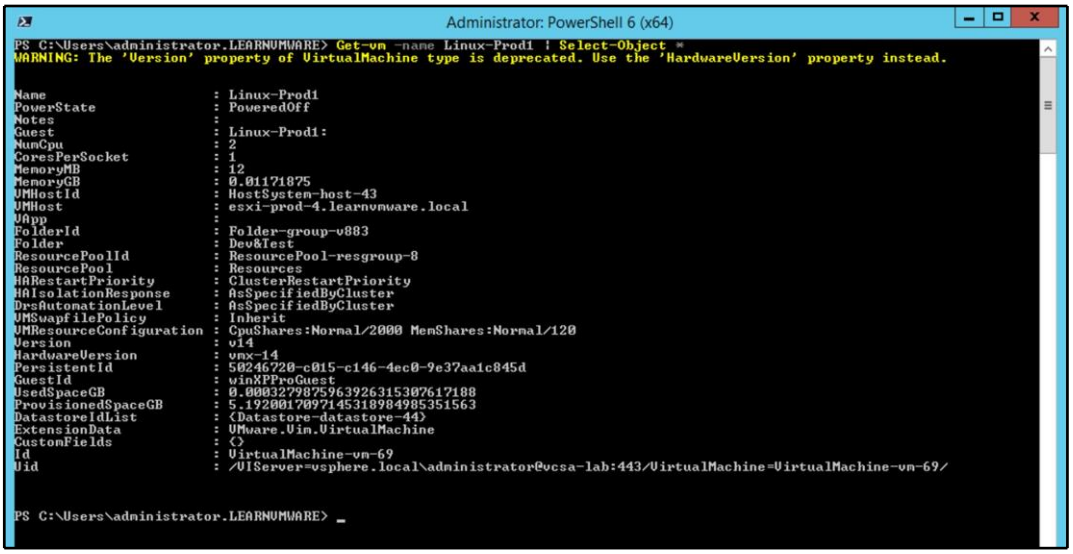
You get a brief explanation of the command, the syntax to use, and the description. If you append `-example` at the end of the command, the system also displays examples of how to use the command. You can pipeline multiple PowerShell cmdlets to build a script in a single line of code.

Sometimes, the `Get-*` command will retrieve only the necessary information about the object, such as PowerState, number of CPUs, and memory size, but there may be more properties of the object.

To get the full list, you can append `Select-Object *` on the command, as follows:

```
Get-vm -name Linux-Prod1 | Select-Object *
```

The output of the command is displayed in the following screenshot:



PowerCLI script examples

Here are some of the examples of PowerCLI scripts that are used to perform some tasks in the vSphere environment, as follows:

- **To move VMs to another host**, use the following script:

```
get-vmhost -name esxi-prod-3.learnvmware.local | get-vm |
Move-VM -Destination (Get-VMHost-name esxi-
prod-1.learnvmware.local)
```

- **To move a single VM to a different host**, use the following script:

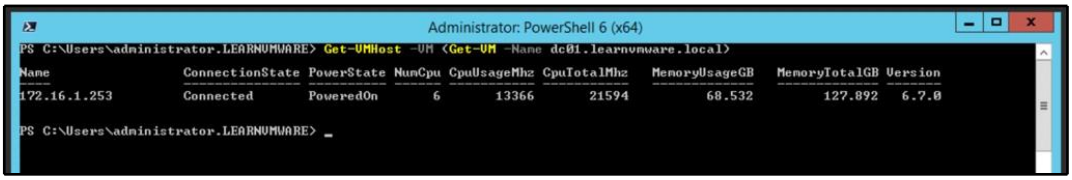
```
Move-VM -VM VM_name -Destination esxi-prod-1.learnvmware.local
```

- **To get information about the VMs**, previously, we used the `Get-VM` command to retrieve a list of running VMs in the vCenter Server instance. You also have the option of exporting the list of VMs in a `.csv` file, including some properties you want to specify:

```
Get-VM | Select-Object Name,NumCPU,MemoryMB,PowerState,Host |
Export-CSV VMinfo.csv -NoTypeInformation
```

- **To find out on which host a specific VM runs**, use the following script:

```
Get-VMHost -VM (Get-VM -Name VMname)
```



- **Configuring NTP**: We have already discussed the importance of having the hosts time-synced to avoid authentication issues. The following cmdlet configures the NTP server for the specific host, as well as setting the service to automatically start with the host:

```
Add-VmHostNtpServer -VMHost $vmhost -NtpServer
172.16.1.1,172.16.1.2
Get-VmHostService -VMHost $vmhost | Where-Object {$_.key -eq
"ntpd"} | Start-VMHostService
Get-VmHostService -VMHost $vmhost | Where-Object {$_.key -eq
"ntpd"} | Set-VMHostService -policy "on"
```

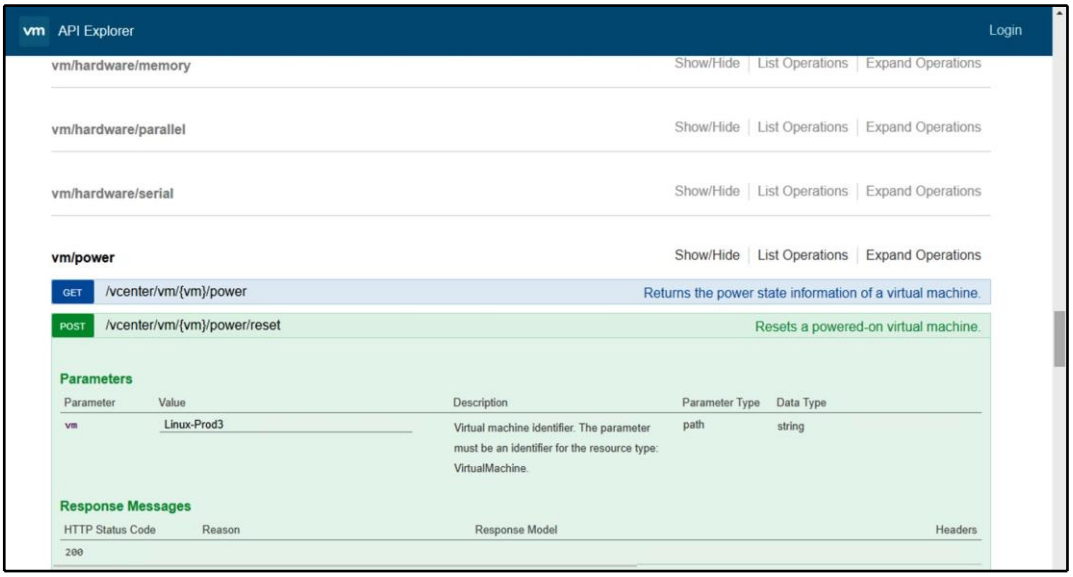
As you can see, with PowerCLI you can automate everything, and I would strongly suggest getting used to PowerCLI, because it is a powerful tool that can save you a lot of time, especially with repetitive tasks.

vCenter REST API

A new feature introduced in vSphere 6.5 is a REST API, which is a more modern, more straightforward-to-use, and more developer-friendly vSphere API. Compared to the capabilities provided by the vSphere API, at the time of writing, not all functions are supported by the REST API. However, compared to vSphere 6.5, the range of supported functions has been extended significantly.

Embedded in the vCSA, there is an API Explorer that allows you to access the documentation of the new REST APIs.

Access your vCSA at the address `https://<VCSA_IP>/apiexplorer` to reach the API Explorer and click on the **Select API** drop-down menu to select the available endpoints, as shown in the following screenshot:



To get the complete documentation on a specific API (description, required fields, request body, and more), click the **Show/Hide** option to expand the available sections. For example, to reset a running VM, I can use the following URL:

```
https://vcsa-lab.learnvmware.local/rest/vcenter/vm/Linux-Prod3/power/reset
```

The whole command for CURL is as follows:

```
curl -X POST --header 'Content-Type: application/json' --header
'Accept: application/json' --header 'vmware-api-session-id:
7d884f11981fbd6e7b383ca737277834'
'https://vcsa-lab.learnvmware.local/rest/vcenter/vm/Linux-Prod3/power/
reset'
```

12
Life Cycle Management, Patching, and Upgrading

vSphere 6.7 simplifies and enhances the capabilities for patching and upgrading ESXi hosts and the **vCenter Server Appliance** (**vCSA**). In addition, vSphere 6.7 introduces many new features and improvements, such as the vCSA with the integrated **vSphere Update Manager** (**VUM**), vCenter HA, and more. Thanks to those features, migration to the latest release is highly recommended.

The VUM service is now fully integrated into the vCSA and no longer requires an additional external Windows server. The embedded VUM can also benefit from the vCenter HA feature for redundancy. VUM is enabled by default, and only a minor configuration is required so that you have a system that's ready to handle patches and upgrade tasks. You will also learn how to patch ESXi hosts using the command line in situation where you do not have vCenter Server.

vSphere 6.7 includes a migration tool that allows administrators to easily and quickly migrate from vCenter for Windows to VCSA.

In this chapter, we will cover the following topics:

- Patching a vSphere 6.7 environment
- Upgrading workflow and procedures
- Upgrading the vCSA
- vCenter 6.5 for Windows to vCenter 6.7 for Windows
- vCenter 6.5 for Windows to vCSA 6.7 migration
- Upgrading standalone ESXi servers
- VUM
- Updating the vCSA

Patching a vSphere 6.7 environment

Keeping ESXi hosts and vCenter Servers up-to-date is not only an essential best practice, but it's strongly recommended to ensure the correct functionality of the virtual platform and protection from bugs. Several methods are available for patching ESXi hosts via the use of VUM (this will be discussed later in this chapter) to update all hosts automatically. Alternatively, if no vCenter Servers are present in the network, the command line of the ESXi server can be used as well. Also, the vCSA can be patched in different ways, all of which will be analyzed later on.

There are two different upgrade types:

- **Minor updates**: From one build to a higher one, but still within the same major version. For example, from ESXi 6.5 U1 (build 5969303) to ESXi 6.5 U2 GA (build 8294253).
- **Major updates**: From one major version to a higher major version. For example, from ESXi 6.5 U2 GA (build 8294253) to ESXi 6.7 GA (build 8169922).

You can check versions and corresponding builds at `https://kb.vmware.com/s/article/2143832`.

If you are performing a minor update of your ESXi servers, it is not necessary to upgrade your vCenter Server. If you are performing a major update, vCSA must be updated before you update your ESXi servers, otherwise it will not be able to manage the newer hosts.

Feel free to check out the **VMware Product Interoperability Matrices** at `https://www.vmware.com/resources/compatibility/sim/interop_matrix.php` to get a better understanding of compatibility requirements between different products.

Upgrade flow to vSphere 6.7

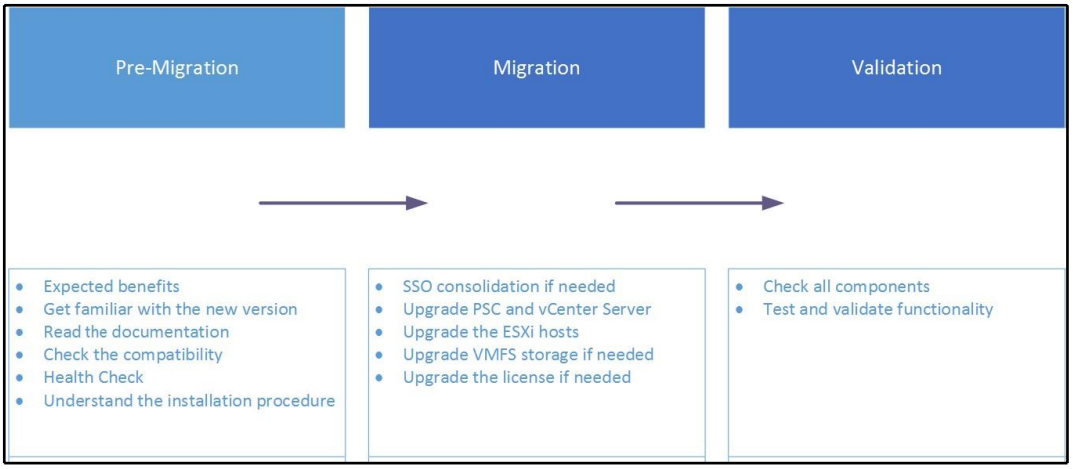
The latest version of the VMware virtual platform, vSphere 6.7, comes with exciting new features and improved capabilities that bring tremendous benefits for the network regarding improved functionality, management, security, and more that were not available in previous versions.

The entire upgrade process needs to follow a specific sequence and flow; first, all PSCs, then all vCenter Servers, then all ESXi hosts. However, if you have more VMware products, the entire sequence could be much more complicated.

For more information, see **KB 53710**: **Update sequence for vSphere 6.7 and its compatible VMware products** at `https://kb.vmware.com/s/article/53710`.

Upgrading the workflow and procedure

For a successful migration to the new version, the overall procedure must be carefully planned with a precise and well-executed workflow to avoid potential issues, such as service disruption or compatibility issues with running components. The migration procedure plan can be split into three main steps, as shown in the following diagram:



Let's look at these steps in more detail.

Step 1 – pre-migration

The pre-migration step includes a plan of the tasks that should be done before starting the actual migration. An analysis of the expected benefits of the new features should be done to determine the added value to your business and justify the investment to management. Try to obtain as much documentation as you can, such as guides, release notes, and tips of the new release, to limit possible problems with the upgrade. Explaining how the new release and new features should be implemented and configured is also an essential point for a successful migration.

Make sure that running programs in the current vSphere environment are also supported in the release you are going to install. If other VMware products are used in your network, validate the compatibility of each product by using the **VMware Product Interoperability Matrices** that are available on the VMware website at the following URL: `https://www.vmware.com/resources/compatibility/sim/interop_matrix.php`.

Ensure that the backup solution in use supports the new version to ensure the protection of your workload. In the event of incompatibility, take all the necessary actions (upgrades, replacement) to ensure full support before migrating.

Performing a health assessment for your current vSphere environment is useful for detecting objects that are no longer needed but are still consuming resources. They also help fix misconfigurations to avoid issues during migration and network cleanup.
There are several tools available on the market that allow you to perform a healthcheck on the vSphere environment. In addition to solutions that require a license, some free tools can also be used to health check your virtual platform.

The following are some examples of commercial and free products that you can use to health check your environment:

- **Licensed**: VMware vRealize Operations Manager, **vSphere Optimization Assessment** (**VOA**), Runecast Analyzer, or Opvizor Health Analyzer
- **Free**: Turbonomic Virtual Health Monitor, Veeam ONE Free Edition, or RVTools

Step 2 – migration

Plan all the involved steps accordingly, evaluating the impact of new features and the improvements that can be applied to the current environment. An upgrade order of the virtual components should be established to avoid potential problems. For example, the vSphere platform requires first upgrading the vCenter Server and then the ESXi hosts to avoid communication issues within vSphere components.

Since VMware has deprecated the Windows-based version of vCenter Server, it's worth migrating directly to the Linux-based vCSA. This takes advantage of the new features that were introduced in version 6.5, such as embedded VUM, **vCenter High Availability** (**VCHA**), and built-in file-based backup restore, which is available in the vCSA only. If you have vCenter Servers with external **Platform Services Controllers** (**PSCs**), get both components on the same version to take advantage of new features.

You can perform an upgrade of vCSA 6.5 to 6.7 using the following tools:

- **Graphical interface**: Using this, you can insert the new vCSA ISO image into your management station as well as by using guided installation
- **CLI interface**: On the vCSA ISO file, you can also find the CLI that allows you to upgrade vCSA in unattended mode

When PSCs and vCenter Servers have been upgraded, you can start migrating ESXi hosts. ESXi 5.5 is not supported in version 6.7 at all, so you must upgrade those ESXi hosts before managing them through vCenter 6.7. ESXi 6.0 or 6.5 can be managed by vCenter 6.7, although new features won't be available for such hosts.

For large environments, you could schedule the upgrade in different maintenance windows by upgrading the vSphere environment in stages. You can start by upgrading all PSCs, followed by the vCenter Servers, and then the ESXi hosts. Planning the upgrade process in different steps reduces the maintenance of the environment in three shorter time frames, thus limiting downtime. To avoid issues, upgrade each vCenter **Single Sign-On** (**SSO**) or PSC one at a time.

Step 3 – validation

Ensure that the upgrade has been completed successfully and that all components work as expected. Verify the full functionality of the vSphere infrastructure and integration with third-party products (such as backup software) according to plan. Once the validation has succeeded, the migration procedure is complete.

Upgrading vCSA 6.5 to vCSA 6.7

VMware made a significant effort to simplify the migration process to vSphere 6.7. They did this by introducing direct upgrades from the installation media of vCenter Server Appliance from an existing vCSA and PSC Appliance 6.5 or a Windows-based vCenter Server to the new version. The tool supports vCenter Servers running version 6.0 and higher.

The upgrade process comprises two stages:

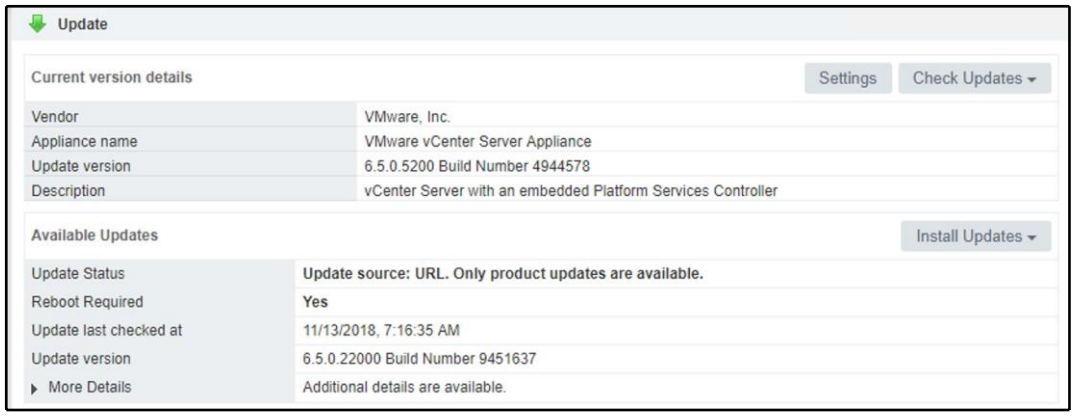
1. vCSA deployment
2. Making a copy of the configuration from the vCenter Server source

The automated upgrade process requires the **Distributed Resource Scheduler** (**DRS**) feature in the cluster, in which the source vCenter Server is installed but not set to fully automated mode.

The upgrade procedure is straightforward and guided through a simple and clear UI in which you must specify source and target network parameters in the upgrade wizard when requested.

Before you start with the update, don't forget to create a backup of an existing vCSA appliance from the **vCenter Server Appliance Management Interface** (**VAMI**).

You can't upgrade to the new major version directly from the VAMI interface of vCSA. Those upgrades are between minor versions. As you can see, on my vCSA 6.5, I only have the option to upgrade to a newer build of vCSA 6.5, not 6.7:

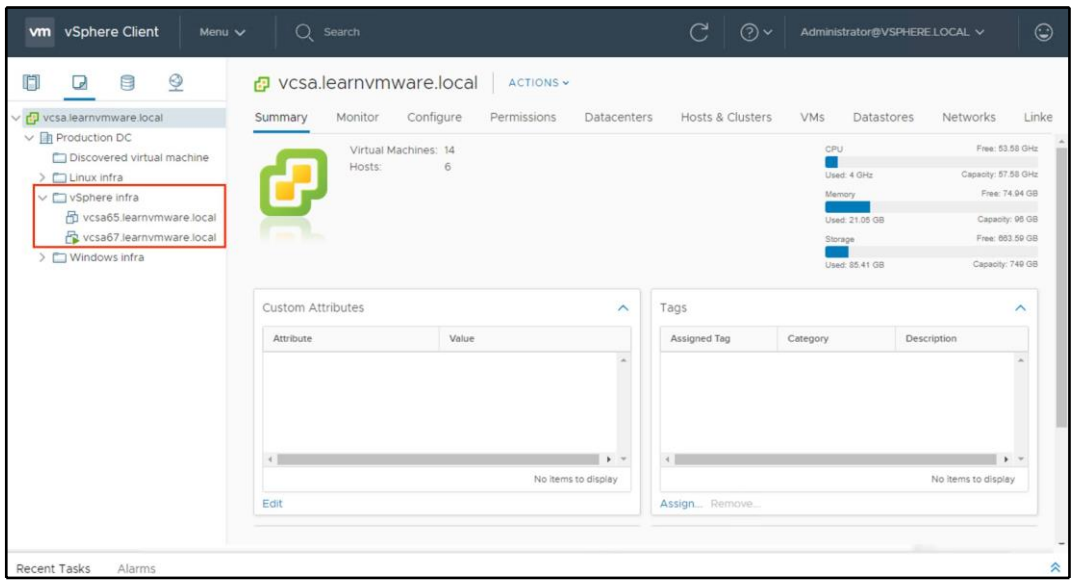


The upgrade itself is not an in-place upgrade. The original vCSA won't be touched. Instead, a new vCSA will be deployed, and the data will be migrated.

To perform an upgrade of vCSA 6.5 to 6.7 with an embedded PSC, the following steps are required:

1. Mount the ISO installation media of vCSA to your management station.
2. Launch the GUI installer located at `CDROM:\vcsa-ui-installer\win32\installer.exe`.
3. Select the **Upgrade** option.
4. The **Upgrade** wizard looks similar to the installation one, and only a few steps are different. After you accept the end user agreement, you need to connect to the existing vCSA and fill in the required password for vCSA itself and the ESXi host that is being managed by the current vCSA.
5. In the next steps, you need to provide a destination ESXi host or vCenter Server where the new vCSA will be deployed. You also need to provide the new vCSA VM name and root password.
6. After you have selected the deployment size based on your inventory size and datastore, which will be used to host new vCSAs, you need to specify networking of the new vCSA. Keep in mind that you need to assign a different IP address to the new vCSA that will be used during migration. Once the migration is done, the IP will be changed to the original one.
7. Once stage 1 is complete, stage 2 configuration (the actual migration) will start.
8. Stage 2 is slightly different from the new vCSA installation. First, the pre-check is performed, and you will see the output of the pre-check in the wizard.
9. Once you resolve the warnings from the pre-check, you can select what data you need to migrate to the new vCSA. This decision will significantly affect the overall duration of the migration.
10. Once all the information is gathered, you can double-check everything before you start with the actual upgrade.
11. In the last step, you need to confirm that the original vCSA will be shut down during the process. Once the data is migrated, the old vCSA will be powered off, and a new one will get an original IP address.
12. Once the whole procedure has finished, you can see different messages based on your vSphere infrastructure and its configuration.
13. After the process has completed, you can log in to the VAMI interface of the vCSA (at the original IP address or FQDN), and you should see that the new version is running.

Now, if you log in to the vSphere client (HTML5 or Flex), you will see that the new VM has been deployed that hosts vCSA 6.7, but you will also see that the old VM is still available but powered off, as shown in the following screenshot:



The migration process doesn't delete the old vCenter Server and its configuration, but copies data to the new vCSA and then powers the source vCenter off. This allows you to quickly restore the old vCenter Server if the upgrade process fails.

Upgrading vCenter 6.5 for Windows to vCenter 6.7 for Windows

vCenter for Windows is a fully supported deployment type in VMware vSphere 6.7, but keep in mind that this is the last version that supports vCenter for Windows. In the next release, vCSA will be the only supported deployment.

The upgrade consists of two steps:

1. Upgrade PSC (the embedded version is not used)
2. Upgrade vCenter Server

PSC upgrade

Before you upgrade vCenter for Windows to vCSA, the PSC must be migrated to the corresponding vSphere version, which in our case is vSphere 6.7.

If you are migrating vCenter for Windows with an embedded PSC, you can skip this step. You can follow these steps to upgrade your PSC:

1. Perform a backup of the Windows Server that hosts the PSC.
2. Mount the installation image of vCenter Server for Windows to the external PSC and start the installation wizard of vCenter Server.
3. The installation will realize that it has been launched on a system that contains a previous version of the vSphere component and the upgrade wizard will be launched.
4. Accept the license agreement and provide an SSO user account.
5. You can change the default ports that PSC will be bound to as well as the installation directories.
6. At the summary windows, you must check that the Windows server was backed up. If not, the installation will not be permitted.
7. Only after the upgrade is completed can you continue with the next steps.

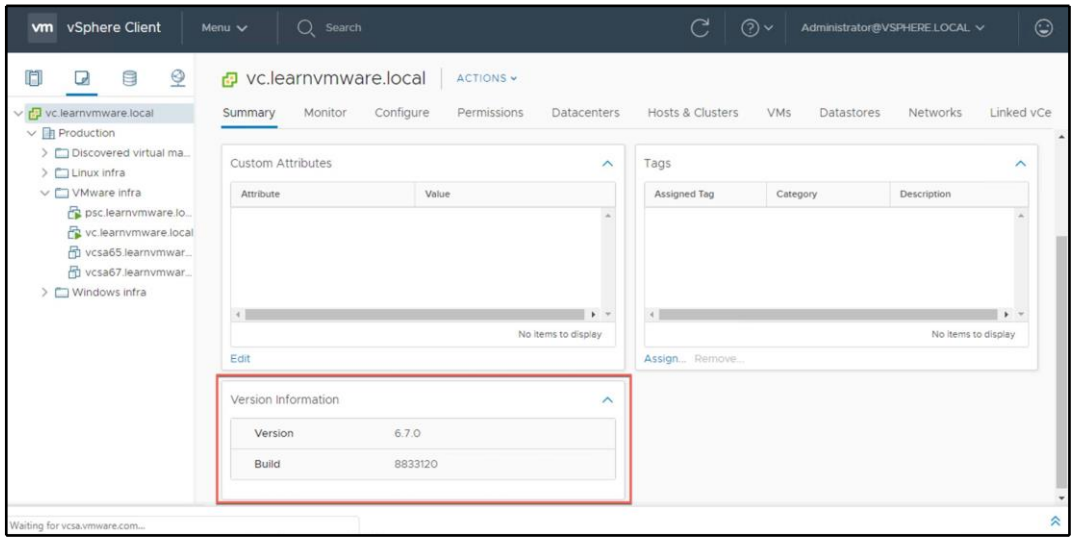
Upgrading vCenter Server

The upgrade of vCenter Server for Windows consists of the same steps as the PSC upgrade:

1. Perform a backup of the Windows Server that hosts the vCenter Server as well as the SQL database (if external is used).
2. Mount the vCenter Server for Windows 6.7 installation image.
3. Start the installation of the vCenter Server for Windows.
4. The installation will discover that the previous version of vCenter is running and the upgrade will be performed instead.
5. Follow the installation wizard, provide the SSO administrative password, and customize the installation directories if necessary.

6. In the summary of the upgrade, you need to explicitly confirm that the backup was taken. Otherwise, the **Upgrade** option will be grayed out.

After you log in to the vCenter Server, you will see that the configuration is intact (for example, custom virtual machine folders) and that you are running vCenter 6.7:



Migrating vCenter 6.5 for Windows to vCSA 6.7

Historically, many infrastructures were based on vCenter for Windows, especially those that were not deployed after vSphere 6.5 was announced. Before vSphere 6.5 was introduced, the vCSA was an appliance with a limited set of features and many companies preferred to use a full-feature vCenter Server running on Windows.

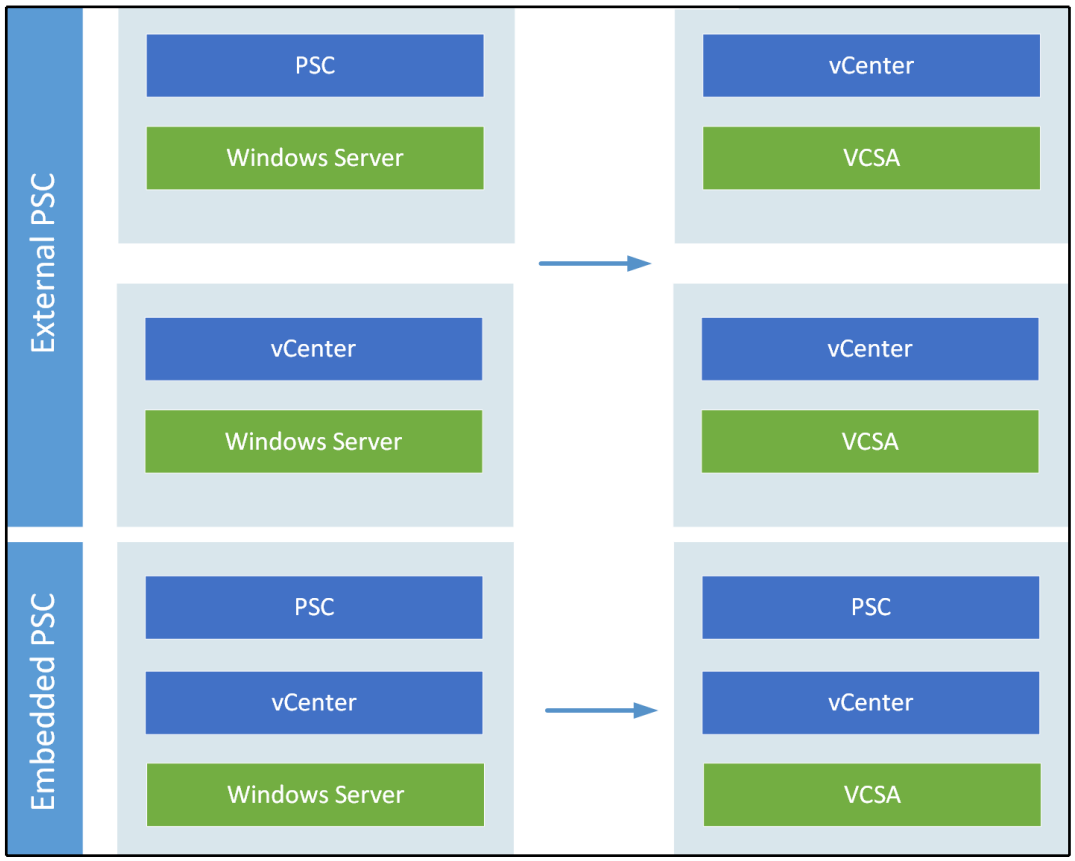
From vSphere 6.5, vCSA is the preferred vCenter deployment option, and vSphere 6.7 is the last release that supports vCenter for Windows. Customers are encouraged to perform the migration from vCenter for Windows to vCSA.

VMware introduced the vCenter for Windows to vCSA migration wizard in vSphere 6.5 as an integrated part of the vCSA installation.

Migration procedure

The migration from the vCenter Server for Windows to vCSA has several steps, depending on your vSphere deployment:

- With an external PSC, migrate to the dedicated vCSA machine that will be used as an external PSC. Migration from an external PSC to an embedded one is not supported.
- Migrate vCenter for Windows 6.5 to vCSA 6.7. If you have installed a dedicated vCenter server and PSC, the result will be two appliances, one running vCenter Server and the second one running the PSC. With the embedded Windows deployment type, the result will be a single vCSA appliance holding both the vCenter and PSC roles:



Migrate in this context means migrate and upgrade in a single step. Check the following documentation for additional information about vCenter for Windows to vCSA migration: `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vcenter.upgrade.doc/GUID-9A117817-B78D-4BBE-A957-982C734F7C5F.html`.

The migration itself is a straightforward process, as all you need to do is launch the migration wizard.

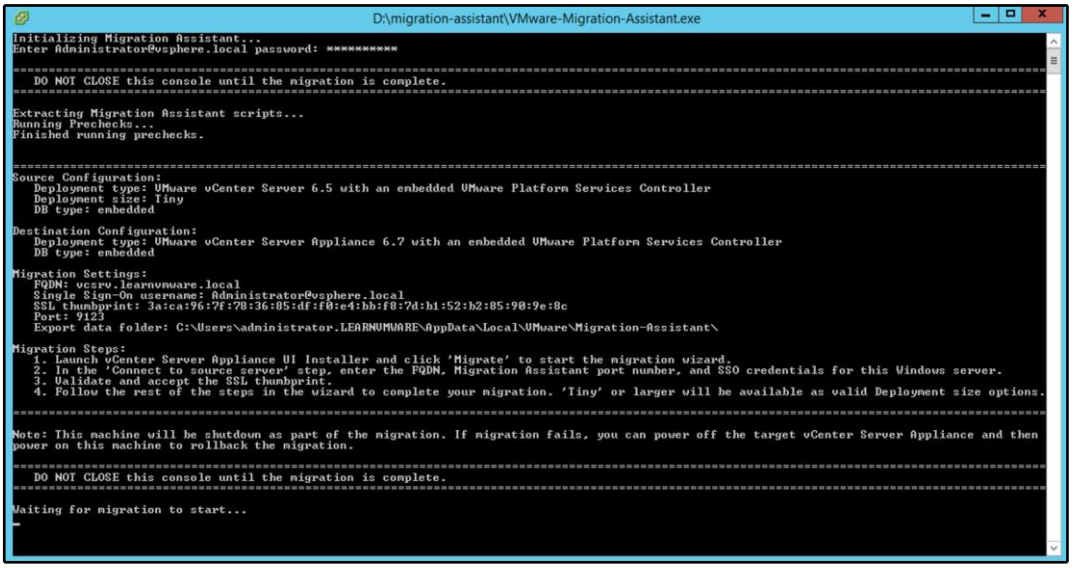
The migration wizard supports all kinds of vCenter for Windows deployments:

- vCenter for Windows with embedded PostgreSQL database and embedded PSC
- vCenter for Windows with embedded PostgreSQL database and external embedded PSC
- vCenter for Windows with an external database and embedded PSC
- vCenter for Windows with an external database and external embedded PSC

If you have an external PSC, perform the migration of the PSC first, then continue with vCenter Server. If you have vCenter Server with an embedded PSC, only migrate the vCenter Server.

To migrate vCenter for Windows with an internal PostgreSQL database and an embedded PSC to vCSA with an embedded PSC, follow this procedure:

1. From the source Windows vCenter Server, launch the migration assistant from the installation ISO image of the vCSA 6.7. The migration assistant is located at `CDDRIVE:\migration-assistant\VMware-Migration-Assistant.exe`:



2. Plug in the vCSA installation image to your management station and select **Migrate**.
3. After you have agreed with the license terms, you need to connect to the source—vCenter Server for Windows.
4. The **Appliance deployment target** step can be an ESXi server or a vCenter Server, as you already know. The next steps are the same as for a new vCSA installation. Here, you need to provide the VM name of the new vCenter Server, root password, deployment size, and datastore.
5. In the **Configure network settings** step, you need to provide a temporary IP address that will be used during the migration process. Once the migration has finished, the new vCSA will be reconfigured with the old IP address that's currently configured in the vCenter for Windows.
6. In the last step, you can check everything and if the configuration is correct, start with the deployment.

Once stage 1 is complete (initial VCSA deployment), you can continue with the actual migration:

1. Based on your vSphere environment, you might see different warnings and notices once the stage 2 migration wizard connects to the source vCenter for Windows.
2. If your vCenter Server for Windows is a member of an **Active Directory** (**AD**), you will be prompted for AD credentials so that you can join the new vCSA to the AD.
3. You have an option to select which data will be migrated to the new vCSA appliance.
4. Next, you can join the **Customer Experience Improvement Program** (**CEIP**), and in the final step, you have to confirm that the backup of the source vCenter for Windows was done.
5. During the migration, the source vCenter Server for Windows will be powered off.
6. Now, the actual migration will be performed based on your configuration settings.
7. If the migration was successful, you should see a screen with the results of the operation.

Once the migration has finished, you can log into the web interface of the new vCSA appliance (running on your original IP address) and verify that the configuration was moved, as well as that the version installed matches the targeted version.

At this stage, vCenter 6.7 should be running in your environment (either upgraded from the previous vCSA version, migrated from vCenter for Windows, or running as a Windows service in the case of vCenter for Windows). It is now time to upgrade your ESXi hypervisors.

Upgrading standalone ESXi servers

Once you have successfully upgraded your vCenter server to version 6.7, you can start with the ESXi servers.

It is crucial to check that your hardware is supported in the targeted version of VMware vSphere by using the **Hardware Compatibility List** (**HCL**). Over time, some older hardware platforms will no longer be supported in the new vSphere releases.

The **ESXi Compatibility Checker** Python tool can be used to verify such compliance for you in an automated way.

ESXi compatibility checker

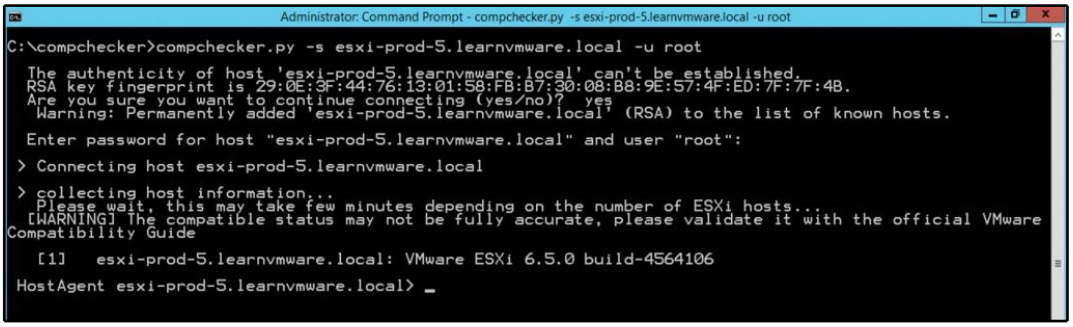
The ESXi compatibility checker tool which is available as a Fling from VMware labs at `https://labs.vmware.com/flings/esxi-compatibility-checker`, can save you from taking the time of going through the HCL manually. To use this tool, you have to install Python on your management station, as described in the **Requirements** tab of the tool in the preceding link.

Flings are small projects that are being developed by VMware employees in their spare time. Why Flings? A Fling is a short-term thing, not a serious relationship but a fun one.

Once the tool has been successfully installed, you can quickly launch it, which will connect you to the ESXi server when you use the following command:

```
Compchecker.py -s IP or FQDN of ESXi -u root
```

The output of the preceding code is as follows:



Once you have connected to the ESXi server, you can check what versions are available for upgrade based on the currently installed version:

- `host 1`: Selects the host (you might connect to the vCenter Server as well, and so multiple ESXi servers will be available. In the case of the standalone host, only one host, `host 1`, is available).
- `comp s`: Verifies the compatibility of the server hardware.

- `up`: Displays all versions that are available for upgrade.

Once you have decided which version you would like to upgrade to, you can verify the compatibility of the physical hardware using the following command:

```
upto 6.7.0 -s
```

The following screenshot shows the output of the preceding command:

As you can see, in this case, the server might be upgraded to ESXi 6.7, but the current driver version of **VMXNET Generation 3** (**VMXNET3**) is not supported.

Once you have confirmed that your hardware is compatible with the targeted version of VMware vSphere, you can decide how to perform the upgrade:

- **Using ISO installation media**: The ESXi server boots directly to the installation of the ESXi server and you can perform the upgrade from there
- **Using CLI:** Through SSH, you can connect to the running ESXi server and perform the upgrade from the running hypervisor

Updating or patching ESXi hosts through the installation ISO

You can boot into the installation image of ESXi server and perform the upgrade from here:

1. Insert the installation ISO image of the ESXi hypervisor to the physical CD-ROM (or the virtual one by using IPMI, iLO, or iDRAC).
2. Reboot the server and boot from the virtual CD-ROM.
3. The installation wizard is exactly the same one that appeared for installing the new ESXi servers on the physical hardware.

4. During installation, you can select where you can select a disk. When installing the ESXi servers, you will see an asterisk mark (*) that indicates that the previous version is already installed on the disk.
5. If you select a disk, you will have following options:
    - **Upgrade and preserve VMFS datastore**: Configuration will be preserved as well as the local datastore.
    - **Install and preserve VMFS datastore:** Configuration will be set to default, but the datastore will be preserved.
    - **Install and overwrite VMFS datastore**: Configuration will be set to default, and the datastore will be re-written. Be careful with the last option. If you have any existing virtual machines on such a datastore, they will be lost.
6. If you do not have any existing datastores on the disk, only the **Upgrade** or **Install** options will be available.

Once the installation or upgrade is done, reboot the server and verify the installed version in DCUI.

Updating or patching ESXi hosts through the command line

Patches and updates for ESXi are combined in a bundle provided by VMware in the `.zip` format that includes some **vSphere Installation Bundle** (**VIB**) ESXi software packages containing fixes and updates.

To proceed with the update, you need to obtain the latest available patches from the VMware website at `https://my.vmware.com/group/vmware/patch`. Patches and upgrades are cumulative, and the patch bundle is provided, which includes all past security and critical updates.

Once the patch bundle has been downloaded, proceed with the following steps:

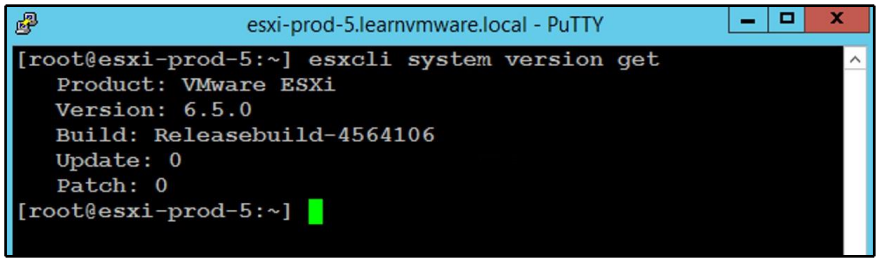
1. From vSphere Client, log in to the ESXi host to upload the downloaded bundle to a local datastore that's reachable by the host. In the navigator area, select **Storage** and then select **Datastores** on the right-hand side.
2. From the available datastores, select the location on which you want to upload the patch and click **Datastore browser**.
3. Create a new folder or select an existing folder, and then click the **Upload** button to upload the patch. Click **Close** when the upload has completed.

Alternatively, you can use a tool such as WinSCP to copy the patch bundle directly to a local datastore on the ESXi host. This may be an option if ESXi hosts to be patched into the network don't have access to the same shared storage.

4. SSH the host using a tool such as PuTTY and log in to the host by entering the root credentials. If the SSH shell is not enabled, from vSphere Client, right-click the **Host** item and select **Services | Enable Secure Shell (SSH)** or enable the SSH service directly from the DCUI.
5. Before patching the host, it can be useful to identify the currently installed build version by running the following command:

```
esxcli system version get
```

The following screenshot shows the output of the preceding command:



6. Before applying the patch, the host must be put in maintenance mode to migrate the running VM off the host and preventing new VMs to be placed in the hypervisor. To enter the host in maintenance mode, run the following command from the command line:

```
esxcli system maintenanceMode set --enable true
```

7. Make sure that the ESXi is in maintenance mode, and then proceed with the update procedure by running the following command:

```
esxcli software vib update -d
/vmfs/volumes/datastore/folder/patch_bundle.zip
```

8. When the patch has been applied successfully, you may need to reboot the host. Run the following command to do this:

```
reboot
```

9. When the ESXi host has been rebooted, exit the host from maintenance mode with the following command:

```
esxcli system maintenanceMode set --enable false
```

10. Check the host version after the update to confirm that the update was successful by running the following command:

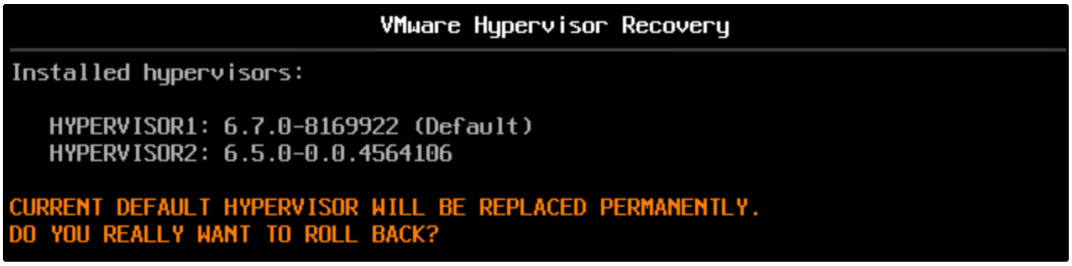
```
esxcli system version get
```

Rolling back to the previous version

If something went wrong, thanks to the two independent boot banks, you have the option to roll back to the previous version.

To perform the rollback, follow these steps:

1. Reboot your ESXi server.
2. When the hypervisor progress bar starts loading, press *Shift* + *R*. You will see the following warning:



3. Press *Y* to roll back the build.
4. Press *Enter* to boot.

For more information, visit the official **KB1033604 – Reverting to a previous version of ESXi** at `https://kb.vmware.com/s/article/1033604`.

VUM

The VUM service is a tool that allows you to efficiently manage patches and updates for VM, hosts, and vApps that are installed in the virtual environment. In comparison to previous versions, VUM no longer requires the installation of an additional external Windows server. This is because, since vSphere 6.5, the Update Manager server and client components have become part of the vCSA.

VUM is installed during the vCSA installation, and it's enabled by default. VUM uses a PostgreSQL database that is bundled with the appliance to store its data. Although both vCenter Server and VUM share the same PostgreSQL database, they use a different database instance.

In vSphere 6.7, a Windows-based Update Manager 6.7 instance cannot be connected to vCSA 6.7 during the installation procedure because it will fail with an error.

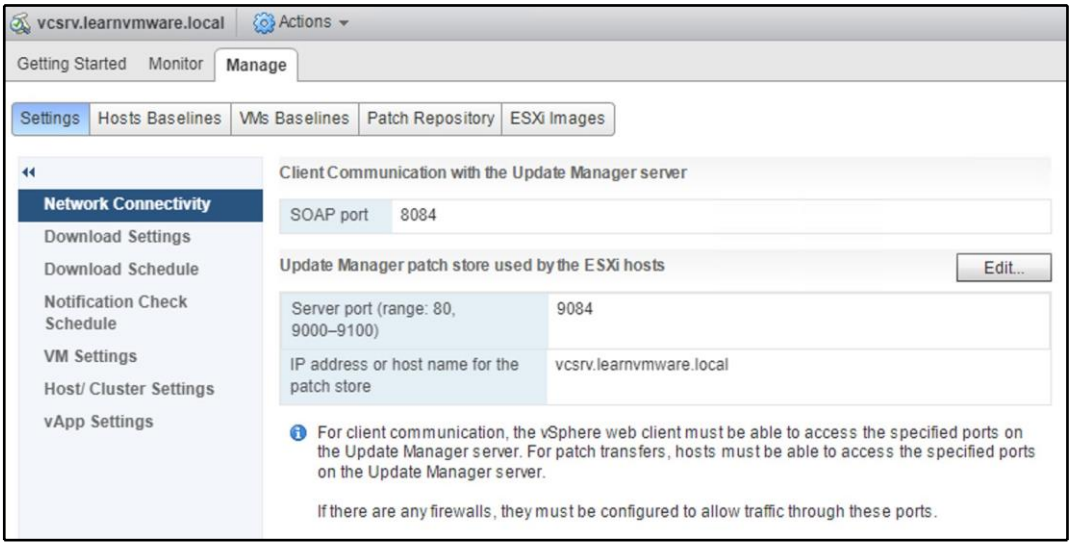
If Update Manager doesn't have access to the internet, you can install the optional **Update Manager Download Service** (**UMDS**) module to download virtual appliance upgrades, patch binaries, patch metadata, and notifications. UMDS must be installed on a machine with internet access and, in version 6.7, it's available for both Windows and Linux-based OSes.

Although VUM is integrated into the vCSA, the UI of the VUM is not yet fully supported in the HTML5 client. To use all of the features of VUM, you need to use an old FLEX Client. In VMware vSphere 6.7U1, it is fully integrated.

Configuring VUM

Default settings configured during the vCSA installation can be modified from the Update Manager administration area, and these changes are only applied to the Update Manager instance that's specified. If you have multiple Update Manager instances in your SSO domain, changes are not propagated to other instances in the group. To specify the Update Manager instance to work with, you have to select the name of the vCenter Server on which the Update Manager instance is registered.

To configure VUM from the vSphere Web Client, select **Home** | **Update Manager** and specify the vCenter Server instance to edit. In the **Manage** tab, you can specify the following VUM configuration settings:



You can see the following tabs in the configuration settings:

- **Network Connectivity**: From this tab, you can only change the IP address or the hostname for the patch store.
- **Download Settings**: Edit this area to specify the patch type to download and additional custom URLs to specify third-party patch repositories. By enabling the **Use a shared repository** option, you can specify the URL of the UMDS instance that's used to centralize the downloads. If a proxy is used to access the internet, edit the **Proxy Settings** area to specify the correct parameters. Patches in ZIP format can also be imported into the repository by using the **Import Patches** button.

Since the service strongly relies on DNS resolution, use an IP address whenever possible to avoid any potential DNS resolution problems. If you use a DNS name, make sure that the specified DNS name can be resolved by vCenter Server and from all ESXi hosts managed by Update Manager.

- **Download Schedule**: In this area, you specify the frequency of patch downloads with the option of sending a notification email (SMTP settings must be configured in vCenter Server by accessing **Configure** | **General area**).

- **Notification Check Schedule**: This is used to specify the frequency that's used by Update Manager to check the VMware repository for notifications about patch recalls, new fixes, and alerts.
- **VM Settings**: You can specify whether a snapshot of the VM should be taken before applying the patch. In case the remediation fails, you can quickly roll back to the snapshot taken before the remediation. Since snapshots affect VM performance, it's strongly recommended that a snapshot retention policy is defined to delete the created snapshots, saving precious storage space.
- **Host/Cluster Settings**: This area allows you to control the operations that are required during remediation. To apply the updates, the target host must be put in maintenance mode, and the running VM must be migrated to other hosts of the cluster to ensure availability. The operation can be automated if vSphere vMotion is configured and DRS is enabled in the cluster. You can also specify that you wish to install patches on PXE booted hosts, but updates are lost after the host reboots if the patch is not included in the host image as well.

The configuration of these settings may vary, depending on the setup of your infrastructure. However, as a general guideline, you can configure host and cluster settings as follows:

- **Disable any removable media devices**: Removable devices may prevent the host from entering in maintenance mode
- **Disable admission control**: It is suggested that this parameter is disabled to make additional resources available to the cluster, especially if you have a few hosts
- **Disable FT**: This is required if you have only two hosts

When a new ESXi patch is available, ensure that you update the image that's used for PXE booted hosts as soon as possible to have the patch applied persistently.

- **vApp setting**: Enabled by default, this allows you to specify the use of the smart reboot feature to reboot the virtual appliances, thus maintaining the correct startup dependencies.

Working with baselines

To upgrade objects in your vSphere environment, you can use predefined hosts and VM baselines that are created during the installation of the vCSA. Baselines are used during the scan of the VM to determine the compliance level of scanned objects (**hosts, VM**, and **virtual appliances**).

While host baselines can be customized, you cannot create custom VM or VA baselines.

In vSphere 6.5 Update 1, VUM was integrated into vSAN, providing an automated update process to ensure a vSAN cluster is up-to-date with the best available release to keep your hardware in a supported state.

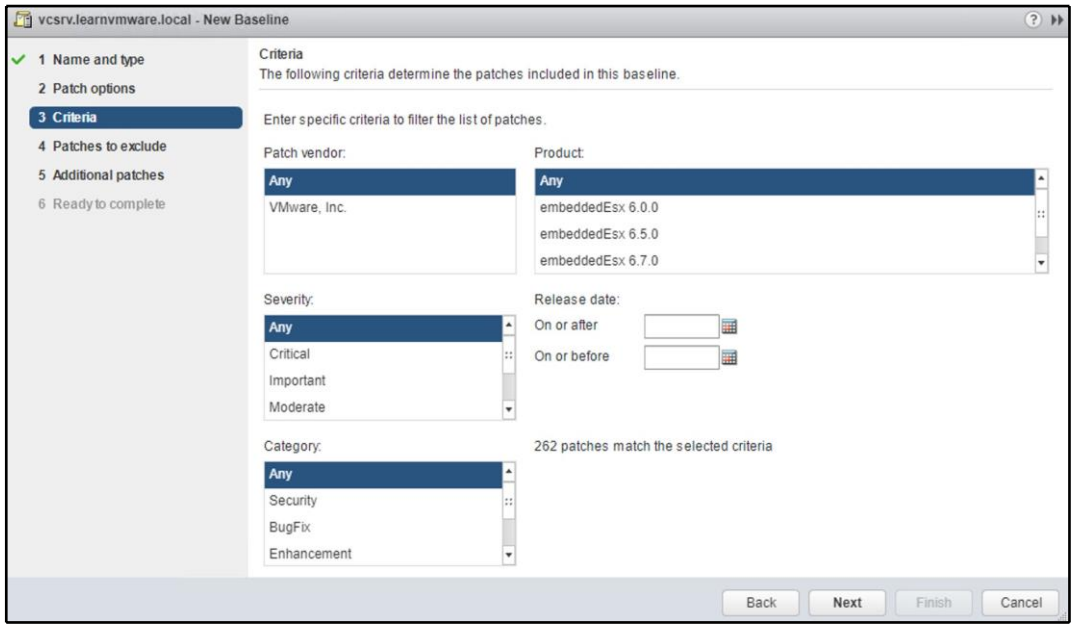
VUM provides some predefined baselines that can only be attached or detached to the inventory objects, without the ability to edit or delete them:

- **Hosts baselines**: This provides critical host patches and non-critical host patches options
- **VMs/VAs baselines**: This provides a VMware Tools Upgrade to Match Host, a VM Hardware Upgrade to Match Host, and VA Upgrade to Latest

To create a new host baseline, proceed as follows:

1. From vSphere Web Client, select **Home** | **Update Manager** and select the vCenter Server instance on which Update Manager is registered.
2. Select **Manage** | **Hosts Baselines** and click **New Baseline**.
3. Enter a name in the **Name** field and description in the **Description** field for the new baseline and specify the baseline type area from the three available options. Click **Next**.
4. Specify the type of baseline patch you want to use and click **Next**. You have two baseline types to choose from:
    - **Static baseline**: The baseline doesn't change, even if new patches are added to the repository. You can create a static baseline to ensure that a specific patch is applied to all the hosts of your environment.
    - **Dynamic baseline**: This is useful to keep systems current as patches change over time. Dynamic baselines specify a set of patches that meet the criteria specified during the configuration, thus adding or removing some specific patches.

5. If a dynamic baseline type has been specified, you have to define the criteria to determine what patches to include in the baseline. For example, to create a host baseline specifically for critical bug fixes, you can select the parameters that meet your needs in the **New Baseline** wizard. You can also specify a release date range to restrict patches that will be included in the baseline:



6. Select patches to exclude from the baseline and click **Next**.
7. Specify additional patches, if any, to include in the baseline and click **Next**.
8. When the parameters have been defined, click **Finish** in the **Summary** window to create the new baseline.

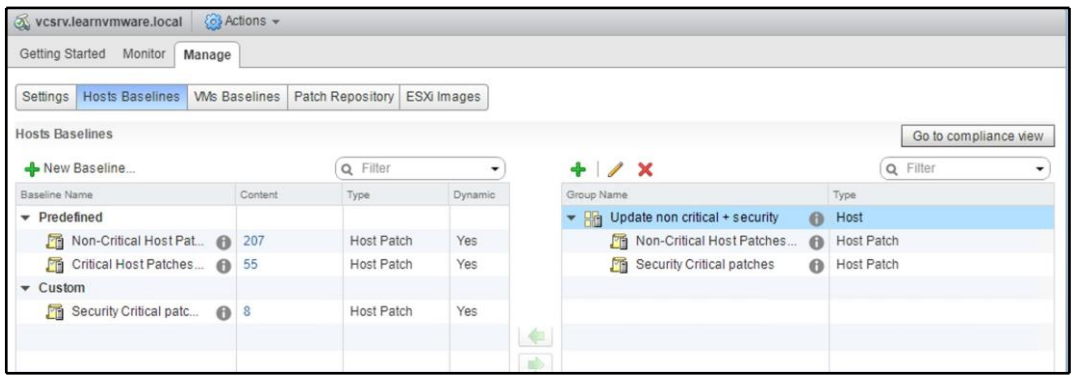
To verify which hypervisor is not compliant with the parameters that were configured in the created baseline, the new baseline must be attached to the ESXi hosts executing the **scan** procedure (host scans will be discussed later in this chapter in the *Scanning VMs and hosts* section).

Baseline groups

In addition to baselines, you can define baseline groups that are used to put together different existing baselines to meet specific needs for your environment. For example, baseline groups can be used if you want to update ESXi hosts in your environment, ensuring that a specific patch is applied during remediation. You can create a baseline group by combining a dynamic baseline for patches and a static baseline with the specific patch you want to apply. To install the latest updates and host extensions to ESXi hosts, using a baseline group allows you to combine different baseline types, performing the task in a single step and simplifying the overall procedure.

To create a new baseline group, from vSphere Web Client, select **Home** | **Update Manager** and click the **Manage** tab. You need to follow these steps:

1. Select **Hosts Baselines** and click the new baseline group button. Enter a name and description and then click **Next.**
2. Specify the upgrades to apply, if any, and then click **Next.**
3. Select the patch baseline to use and click **Next.**
4. Select the extension to apply to the hosts and then click **Next.**
5. In the **Summary** window, click **Finish** to create the new baseline group:



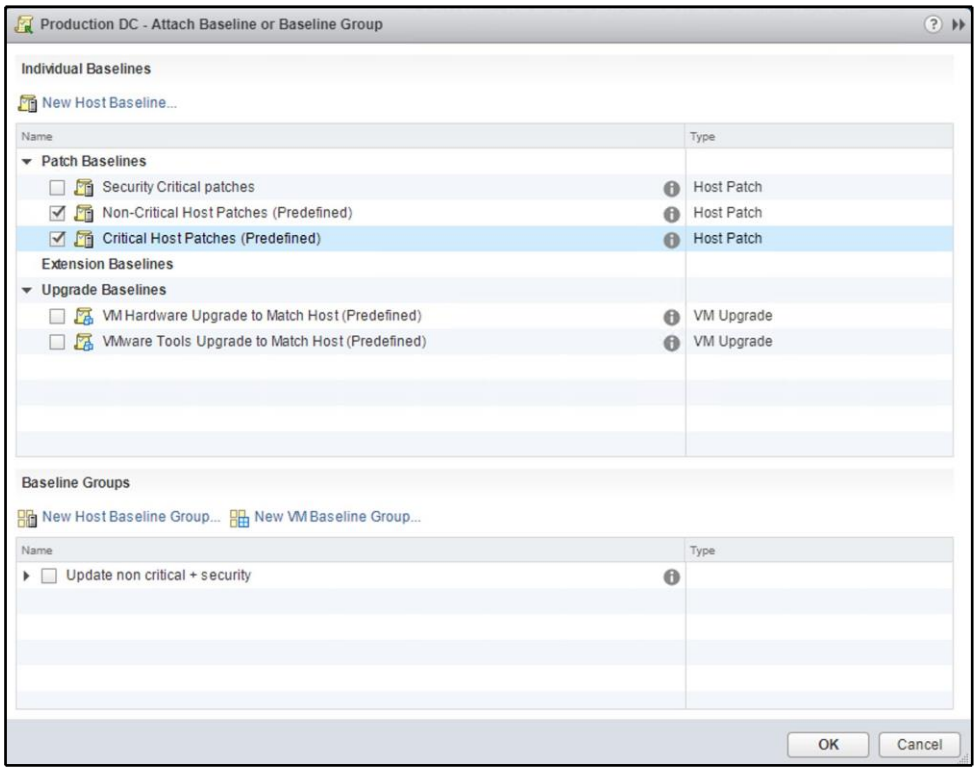
To edit or delete a baseline, right-click the baseline to process and select the **EDIT** or **DELETE** baseline option accordingly.

Attaching or detaching baselines

Once a baseline or a baseline group has been created, you must attach it to a host or VM to scan and determine whether the object is compliant. By attaching a baseline at a higher level in vCenter Server, it will also be applied to child objects. You can attach different baselines at different levels if you need to apply specific baselines to specific objects.

To attach or detach baselines or baseline groups to hosts or VMs, proceed with the following steps:

1. Using vSphere Web Client, from the inventory, select to view the object level on which you want to attach the baseline or baseline group and select the **Update Manager** tab.
2. Click the **Attach Baseline...** button to select the baselines to use from the list and then click **OK.**
3. To detach a baseline, right-click the baseline or baseline group to remove it and click the **Detach Baseline...** button:



Use the **Hosts and Clusters** view to attach baselines to ESXi hosts, and the **VMs and Templates** view to attach baselines to VMs.

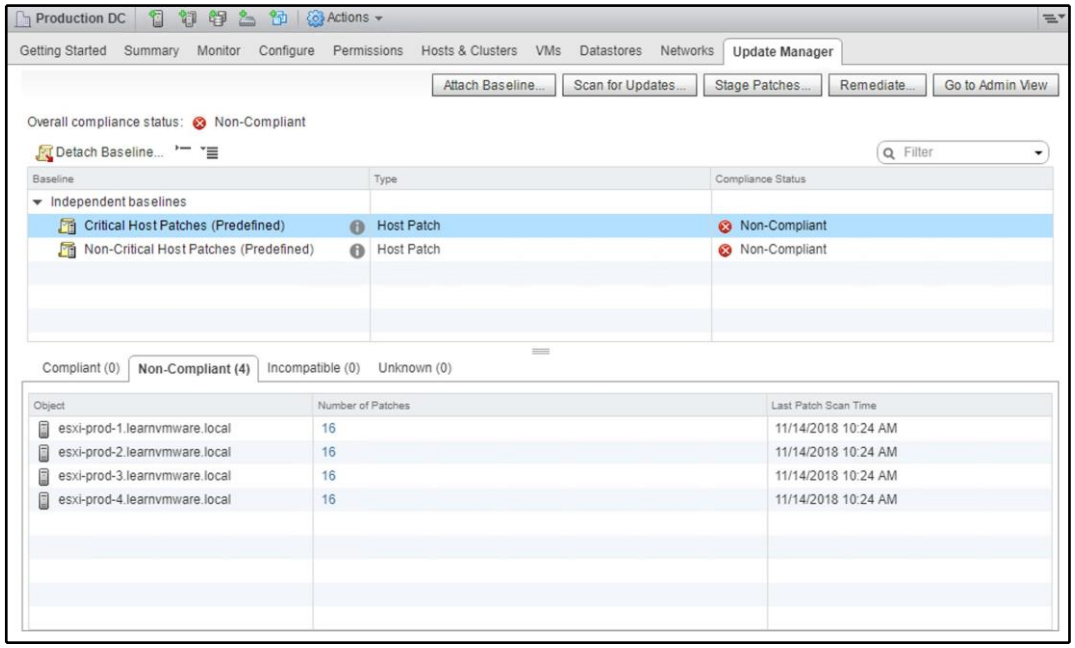
Scanning VMs and hosts

The scanning process allows for the identification of hosts, VMs, or virtual appliances that are not compliant with the attached baselines and baseline groups.

Object scans can be initiated manually or scheduled. To perform a manual scan, select the vCenter Server, data center, cluster, or the host object and then select the **Update Manager** tab:

1. To scan hosts, click the **Scan for Updates...** button to open the dialog box. Select **Patches** and **Extensions** and **Upgrades** as types of updates to scan for, and then click **OK**.
2. To scan VMs and vApps, the procedure to follow is similar to what is performed for hosts. Click the **Scan for Updates...** button and select any of the three available options – **Virtual appliance upgrades**, **VMware Tools upgrades**, and **VM Hardware upgrades**. Click OK to proceed with the scan.

The **Compliance Status** column indicates whether the scanned objects are **Compliant** or **Non-Compliant** against the attached baselines and baseline groups. If the value reported is **Non-Compliant**, you need to perform the remediation of missing patches or updates:



Staging and remediating patches

If the scanned hosts are marked as non-compliant, you need to remediate them to apply missing patches or updates. You have the option to stage or remediate patches and updates.

Let's see the differences between the two processes:

- **Staging**: Patches are copied from Update Manager to the ESXi hosts across the network. This allows you to reduce the remediation time. Staging host patches is not a required step, and it is not necessary to put hosts in maintenance mode while patches are staged. If you have hosts connected to Update Manager over slow WAN, staging patches can reduce the ESXi outage that's required for remediation.
  To proceed with staging from vSphere Web Client, click the **Stage Patches...** button in the **Update Manager** tab and follow these steps:
  1. Select the baselines to attach and click **Next.**
  2. Specify the hosts on which you want to stage patches and then click **Next.**

3. Select the patches and extensions to be staged in the selected hosts and click **Next.**
4. Review the settings selection and then click **Finish** to begin the staging process.

- **Remediating**: The remediation process applies patches and upgrades to the objects that are non-compliant with the attached baseline. To remediate hosts from vSphere Web Client, click the **Remediate...** button in the **Update Manager** tab and follow these steps:
    1. Select the baseline to apply to the hosts and click **Next**.
    2. Select the hosts to remediate and click **Next**.
    3. Select the patches and extensions to apply to selected hosts and click **Next**.
    4. In the **Advanced options** step, you can schedule the remediation task by specifying the name of the task, description of the task, and remediation time. You can also choose to ignore warnings for unsupported hardware devices that may stop the remediation procedure. When you are done with this, click **Next.**
    5. In the **Host remediation options**, be sure to leave the VM power state as set to **Do Not Change VM Power State** to avoid VM downtime, allowing the system to vMotion the VMs to other hosts. Also, tick the **Disable any removable media devices connected to the virtual machines on the host** option. After doing this, click **Next**.
    6. Specify the **Cluster remediation options** to apply to the selected cluster during remediation. Disable DPM, FT (if you have only two hosts in the cluster), and HA admission control (if you have a few hosts in the cluster), and click **Next**.

By default, the remediation process runs sequentially for host members of a cluster. You can enable the remediation in parallel by ticking the appropriate option in the **Cluster remediation options** step.

7. After reviewing the settings selection, click **Finish** to begin the remediation procedure for the selected hosts. When the remediation process is complete, ESXi hosts will be patched/upgraded and ready to host a VM.

If you do not have a DRS license (for vSphere Standard or Essentials), the host won't switch to the maintenance mode automatically because DRS will not be able to migrate the VMs from the hosts. You have to perform the vMotion of running VMs manually. The same applies if you have your cluster DRS configuration set to **Partially automated**. If the DRS is set to automatic, migration will be invoked by the system user, just like any vMotion that is invoked by DRS.

Once the remediation process finishes, the scan will be invoked automatically, and you should see that all ESXi hosts are compatible with the attached baselines.

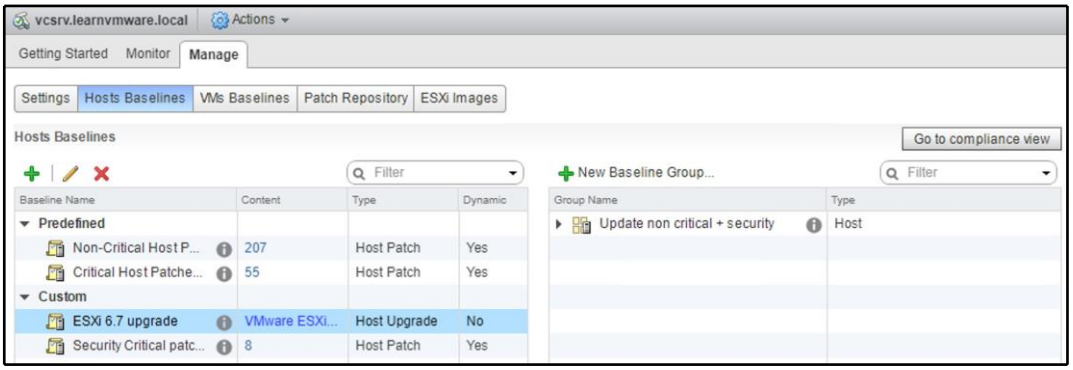
Upgrading hosts with VUM

VUM allows you to upgrade an ESXi host from a previously supported version to the current version.

The first step is the creation of the baseline so that you can attach the hosts to upgrade:

1. From vSphere Web Client, select **Home** | **Update Manager** and select the vCenter Server instance you wish to configure. Select the **Manage** tab.
2. Navigate to **ESXi Images** and click the **Import ESXi Image** button to import the image file that's used to upgrade the ESXi hosts.
3. From the wizard, click on the **BROWSE** button and select the ISO file to use for the upgrade, and then click **Next** to upload the image into VUM.
4. When the upload has been completed, a review of the ESXi image information is displayed. Click **Close** to exit the import wizard. The uploaded image listed in the **ESXi Images** tab will be used to create the baseline.
5. Go to the **Hosts Baselines** tab and click the **New Baseline** button. In the wizard, enter the name of the new baseline and, optionally, add a description. This is useful for identifying the baseline scope. Under **Baseline Type**, select **Host Upgrade** and then click **Next**.
6. Select the ESXi image to use for the upgrade and then click **Next**.

7. A review of the settings selection is displayed. Click **Finish** to create the baseline and exit the wizard:



8. Click the **Go to compliance view** button to attach the new baseline to the hosts to upgrade.
9. In the **Update Manager** tab, click the **Attach Baseline...** button to specify the baseline to attach to the object level (cluster) that contains the ESXi hosts you need to upgrade to the new version and click **OK**.
10. Click the **Scan for Updates...** button to verify the host's compliance against the attached baseline. In the **Confirm Scan** wizard, specify **Upgrades** as the scan for the option and then click **OK**.
11. Click the **Remediate** button and select the created baseline to apply to the ESXi hosts. Click **Next** to continue.

Before starting the remediation process of the hosts, back up the current ESXi configuration to quickly restore the hypervisor in case something goes wrong with the upgrade.

12. Select the target ESXi hosts to remediate and then click **Next**.
13. Accept the EULA and click **Next** to continue the upgrade procedure.
14. Follow the remediation steps, as we discussed previously.

As a result of the remediation procedure, the processed ESXi hosts will now be compliant against the applied baseline.

You can verify the installed version in the **Summary** tab of the ESXi server.

If you are upgrading host members of a cluster, it is suggested to upgrade one host at a time to prevent cluster failure in the event of problems during the remediation process.

Upgrading VM hardware

Another useful feature of VUM is the option to automate and schedule the upgrade of VM hardware version. A VM with an outdated hardware version cannot take advantage of new features that are introduced in the latest VMware vSphere releases. VUM allows admins to easily identify VMs that don't have a current hardware version and upgrade them automatically.

VUM comes with a predefined VM hardware baseline that you can't change or delete, but you should use to upgrade the VM hardware to the current version (vSphere 6.7 introduces hardware version 14).

Hardware upgrades can be performed only while the VM is powered off. If you plan a hardware upgrade of your VM, you should consider that this process will cause downtime.

The hardware upgrade steps are similar to what we have discussed already:

1. From vSphere Web Client, go to the **VMs and Templates** inventory view and select the object level (data center, for example) on which you want to attach the baseline.
2. Select the **Update Manager** tab and click the **Attach Baseline...** button. Select the **VM Hardware Upgrade to Match Host** option and click **OK**.
3. Click the **Scan for Updates** button to check the VM's compliance against the attached baseline. In the **Scan for Updates** wizard, specify **VM Hardware upgrades** as the scan option to allow VUM to detect outdated VM hardware. Then, click **OK**.
4. Now, click **Remediate** to configure the task and upgrade the hardware version for outdated VMs.
5. The remediation procedure requires the selection of the baseline to attach and the selection of the objects to remediate.

6. The remediation task can be scheduled to be executed in the correct maintenance window where the downtime is due to the upgrade having a minor impact on the production environment. By defining a task name and a task description, you can remediate the VM on power cycle, or you can specify three different schedules depending on the state of the VM: powered on, powered off, or suspended. By default, all three options are configured to run the action immediately, so you should pay attention before confirming the remediation execution. Click **Next** to continue.
7. Specify the remediation **Rollback options** step to revert the VM to the state before the remediation if something goes wrong during the upgrade process. It's recommended to configure snapshot retention so that you can delete the snapshot after a specified time. This will help you avoid performance issues. Click **Next** to go to the final step.
8. Review your settings selection and click **Finish** to execute or schedule the remediated task.

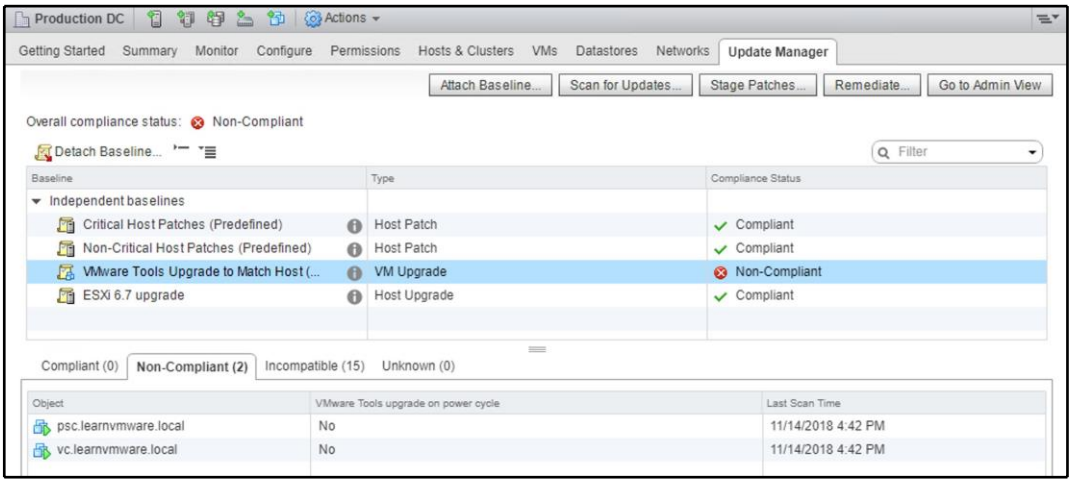
Upgrading VM Tools

VUM can also be used to automate the VMware Tools upgrade process for the VM in the inventory. During the powering on or the restart of a VM, Update Manager can be configured to check the VMware Tools version that's installed in the VM and perform the upgrade to the newest version that's supported by the host that is running the VM. The upgrade of VMware Tools can be scheduled to avoid VM downtime during working hours.

The procedure is the same as what's used to upgrade the hardware version of a VM:

1. From vSphere Web Client, go to the **VMs and Templates** inventory to attach the requested baseline.

2. Once the **VM Tools Upgrade to Match Host** baseline has been attached (through the **Attach baseline...** button), click **Scan for Updates...**, selecting VM Tools upgrades as the option to check the VM's compliance against the attached baseline. After doing this, click **OK:**



If the virtual machines do not have VMware Tools installed, they will be listed in incompatible objects.

3. To remediate a non-compliant VM, click the **Remediate** button to upgrade the VM Tools for an outdated VM.
4. Now follow the steps we used previously to upgrade the hardware version to complete the remediation procedure.
5. Once the upgrade has been performed, you will see that the previously non-compliant objects are now in a compliant state.

Updating the vCSA

You should check for minor updates of the vCSA to keep it up-to-date. For minor upgrades on the same major version, you do not need to use a vCSA installation ISO image – the upgrade can be performed from the VAMI interface of the vCSA or CLI.

Updating the vCSA through the command line

Since vSphere 6.5, upgrading the vCSA has also been simplified. There are two ways to patch the vCSA—through VAMI, which was introduced in vSphere 6, or by using the command line. From the VMware website, download the latest vCenter Server update that's provided in ISO format and save it anywhere on your computer. The ISO image containing the patches must be uploaded to shared storage that's accessible from the vCSA that's present in the network. The ISO image can also be attached to the CD/DVD drive of the vCSA.

Before proceeding with the update, take a snapshot of the vCSA to quickly revert to a working state in case something goes wrong during the patching process.

Staging and remediating patches

Patches from the ISO file that were previously downloaded from the VMware website can be staged to the vCSA for updates. This can be done by attaching the ISO image to the CD/DVD drive of the vCSA or by specifying a datastore ISO file.

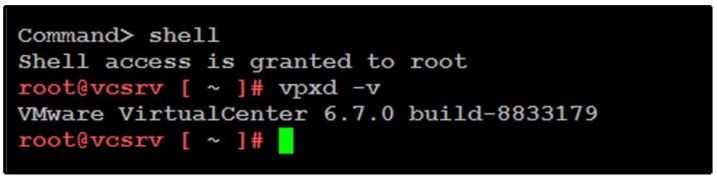
Staging patches is a useful procedure for speeding up the remediation process because patches are already available locally on the vCSA and the downtime during remediation is reduced.

Mount the ISO image patch to the vCSA VM and SSH the vCSA using a tool such as PuTTY. Enter the root credentials to log in to the vCSA and proceed with the upgrade procedure of the vCSA.

To check the current vCSA version from the CLI, you can use the following command:

```
vpxd -v
```

The following screenshot shows the output of the preceding command:



To stage packages to the vCSA, run the following command:

```
software-packages stage --url
```

Staged patches information can be checked with the following command:

```
software-packages list --staged
```

If a mistake is made during the patch staging procedure, you can always unstage the staged patches by running the following command:

```
software-packages unstage
```

To install staged patches, run the following command from the command line:

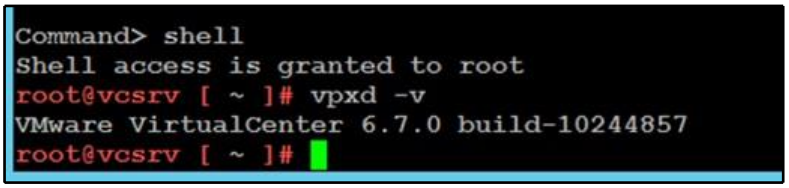
```
software-packages install --staged
```

Staging patches is not a requirement and updates can be installed directly from an attached ISO image. To install patches directly from the ISO image, run the following command:

```
software-packages install --iso
```

The patch installation process requires a few minutes to complete, and a reboot may be necessary.

After the reboot, check the installed version again with the `vpxd -v` command:

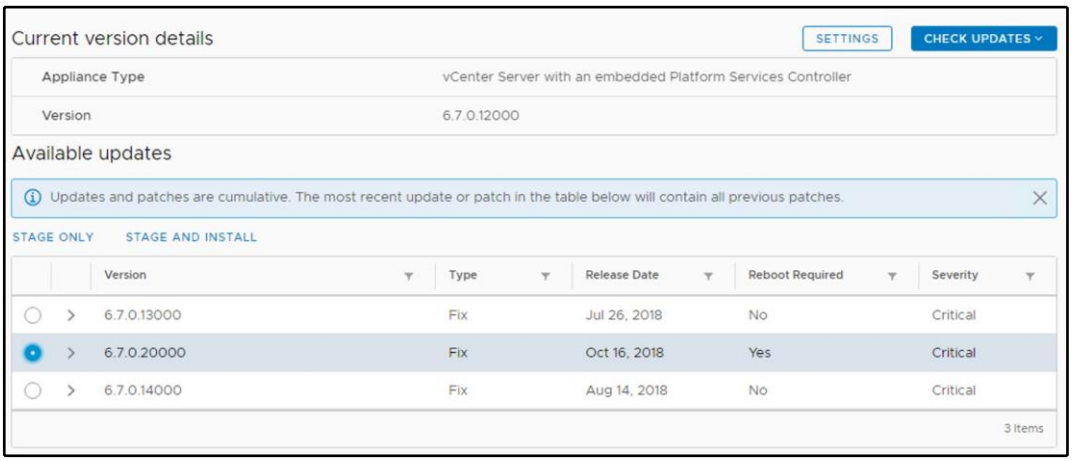


Updating the vCSA with VAMI

To make the overall procedure simpler, the vCSA can also be updated using the UI through the VAMI. To access the VAMI, type `https://<VCSA_IP>:5480` into your favorite browser and enter the root credentials.

Perform the following steps to install the available updates from the repository or the ISO image:

1. Access the **Update** tab and select the **Check Updates** | **Check Repository** option to check the available updates from the default VMware repository:



2. If the ISO image has been mounted to the vCSA VM, from the **Update** tab, select the **Check Updates** | **Check CDROM** option to install the available updates directly from the ISO image.
3. If a new update has been detected, the **Available updates** area displays the information related to the available update, but the current release doesn't provide a list of patches that will be installed in the system. Click **Install updates** to proceed with the update installation.
4. The installation requires a few minutes to complete, and a reboot of the vCSA may be required to complete the update process. Click **OK** to proceed with the upgrade.

If a vCSA in your network is configured with an external PSC, patches or updates must be applied to the PSC and its replicating partners first, and then, if installed, in the vCenter SSO domain.

If a proxy is enabled in the network configuration of your vCSA, you might experience a generic download failed error (both PSC and vCenter Server are affected) when you try checking for updates online through VAMI.

To quickly fix this issue, perform the following steps:

- **From the vCSA UI**, enable the **Proxy Settings** in the **Networking** area.
- From CLI, SSH the vCSA, edit the `/etc/sysconfig/proxy` file, and manually enter a valid HTTP proxy address in the `HTTPS_PROXY` line, as in this example:

```
HTTPS_PROXY=”https://proxy.domain.com:3128/”
```

Once the appliance has been upgraded, you should see the correct version installed in the **Summary** window in the VAMI interface.

13
VM Deployment and Management

Once the setup of the vSphere environment has been completed, the final step involved in firing up your virtual infrastructure is deploying the virtual machines. When the hosts and vCenter Server are in place, they provide physical resources to the VMs that physically reside on the storage device shared within the environment.

This chapter will look closely at the structure of VMs and their configuration to better understand how they work and which options we should configure in order to obtain the best performance. The use of templates is a key point in the management of VMs since it simplifies the management of the environment, allowing VMs to be created easily and deployed quickly. Compared to physical machines, VMs deployed from a template don't need to be installed from scratch, which saves you time.

We will also look at the content library, which allows us to centrally store all our ISO images and templates. We can easily replicate its content over the network to vCenter servers in different sites.

We will cover how to work with virtual machines and which operations can be performed on VMs, such as snapshots. We will also have a look at different physical-to-virtual and virtual-to-virtual conversions.
In this chapter, we will cover the following topics:

- The components of a virtual machine
- Deploying VMs
- The content library
- Managing VMs

- The content library and its features
- Importing and exporting VMs
- Converting VMs

The components of a virtual machine

A VM behaves in the same way as a physical computer, but it's a software computer that runs an OS and applications supported by the host's provided resources. A VM supports all the same functionalities and devices as a physical machine, but it's easier to manage and more secure.

Typically, a VM can be configured to run on ESXi hosts, data centers, clusters, or resource pools, and includes three main components:

- Virtual and hardware resources
- Virtual machine tools
- Guest operating system

Virtual hardware

When you create a VM, the ESXi host presents the hardware as a specific set of resources to the VM. The hardware type provided by the configuration wizard is selected by VMware to ensure the highest level of compatibility with the supported OS.

Every VM has a CPU, memory, and disk resources. Virtual devices in the VM perform the same functions as the hardware on a physical computer. You can configure most of the virtual devices present in the VM, but certain virtual hardware cannot be modified or removed, such as the VMCI device.

When you create a VM, specific virtual hardware is presented to the VM. Sometimes you need to adjust the default hardware to meet the requirements of the guestOS or the applications that will be installed inside the VM. To access the virtual hardware configuration, right-click on the VM and select the **Settings** option.

Let's walk through the main components you need to configure in a VM.

vCPUs

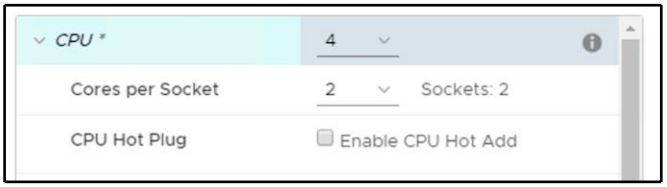
One or more virtual processors can be defined in the VM, but the amount cannot exceed the number of logical processors (sockets x cores x 2 if hyperthreading is enabled) present in the host. The number of vCPU sockets specified in the configuration determines the number of cores available. One VM could have virtual sockets and virtual cores. By default, for each vCPU, a single socket with a single core is assigned as virtual hardware. You can change the default behavior and assign multiple CPU cores per single socket. You might need to do this for licensing reasons, for applications running inside the guest OS (for example, the latest SQL Express can work with more cores, but not with more sockets).

If you have more than eight vCPUs, **virtual NUMA** (**vNUMA**) is enabled, and ESXi distributes the VMs in more NUMA nodes if it is not possible to fit them in just 1.

For more information about NUMA and vNUMA, visit `https://blogs.vmware.com/performance/2017/03/virtual-machine-vcpu-and-vnuma-rightsizing-rules-of-thumb.html`.

If you are running a vSphere Enterprise Plus license, you can also enable Hot Plug for CPU. This option will enable you to add more vCPUs even if the virtual machine is running. You are usually not allowed to change the number of vCPUs while the virtual machine is powered on.

Hot Plug for CPU can be easily enabled by selecting the **Enable CPU Hot Add** checkbox:



The maximum number of supported vCPUs per VM is 128.

Memory

A default amount of RAM is configured based on the selected guest OS. The specified RAM is the memory the OS will present to its system; it's also the maximum amount of RAM the VM can claim from the physical memory installed on the host.

Assigning memory to the virtual machine does not automatically mean that the amount of memory will no longer be available on the ESXi hypervisor level. If the virtual machine is not accessing the memory (see the active memory used performance metric), memory over-commitment techniques can be used to spin more virtual machines.

Note that if you are using over-commitment techniques for memory, you should carefully monitor the active memory used, so it does not exceed the total available amount of memory on the ESXi hypervisor. Otherwise, swapping might occur, affecting the performance of the virtual machines.

Memory optimization techniques are mostly the same as they were in previous versions of VMware vSphere, but with one significant difference. From vSphere 6.0, in **Transparent Page Sharing** (**TPS**), page sharing is enabled by default within VMs (intra-VM sharing), but is enabled between VMs (inter-VM sharing) only when those VMs have the same salt value. This change was made to ensure the highest security between VMs.

For more information, see **KB 2080735: Security considerations and disallowing inter-Virtual Machine Transparent Page Sharing** at `https://kb.vmware.com/kb/2080735`.

In vSphere 6.7, a VM supports a maximum of 6,128 GB of RAM (with the latest virtual hardware versions).

Network adapter

During the VM configuration, you must select the adapter type and the network it will connect to. Depending on the VM compatibility and the guest OS, the supported NIC types are the following:

- **E1000E**: This is the default adapter for Windows 8 and Windows Server 2012. It emulates the Intel 82574 Gigabit Ethernet NIC.
- **E1000**: This driver is available in most newer guest OSes and emulates the Intel 82545EM Gigabit Ethernet NIC.
- **Flexible**: This identifies itself as a Vlance adapter, an emulated version of the AMD 79C970 PCnet32 LANCE NIC. Most 32-bit guest OSes have the driver for this NIC type. When installing Virtual Machine Tools, the adapter changes to the higher-performance VMXNET adapter.

- **VMXNET**: This is optimized for VM performance. To provide the driver, it requires Virtual Machine Tools to be installed.
- **VMXNET 2 (Enhanced)**: Based on the VMXNET adapter, this provides high-performance features, such as jumbo frames and hardware offloads. A limited set of guest OSes support it.
- **VMXNET 3**: This offers all the features available in VMXNET 2 and it's a paravirtualized NIC designed for performance. Multiqueue support (known as RSS in Windows), IPv6 offloads, and MSI/MSI-X interrupt delivery are some of the additional features offered by this adapter. It requires VM hardware version 7 or later, and a limited set of guest OSes support it.

By default, the MAC address is automatically generated from the vCenter Server MAC address pool, but you can set a manual MAC address if you need to.

You can't change the adapter type once the adapter is created. The option to set the adapter type is only available when creating a new network adapter.

Virtual disks

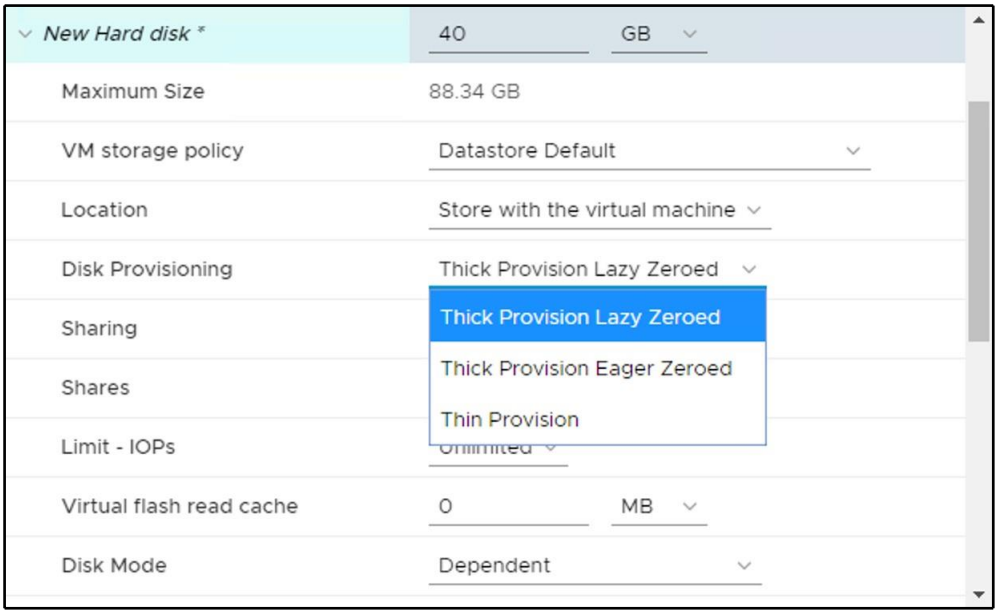
A virtual disk stores the actual VM data and the VM can be configured to use a new disk, attach an existing disk, or map a SAN LUN. A LUN to a VMFS map is referred to as a **Raw Device Mapping** (**RDM**) that points to the raw LUN. In this case, the `.vmdk` file (`.vmdk` files will be discussed later in the file structure) doesn't store data, as the data is stored in the LUN, but it contains the mapping to the LUN disk information.

Virtual disks can be moved across different data stores connected to the host on which the VM runs. When a new disk is created, it can be provisioned in three different formats, depending on the requirements:

- **Thick provision lazy zeroed**: This is the default format; space on the datastore is allocated when the VM is created, and data on the physical device is not erased.
- **Thick provision eager zeroed**: This is the format used to support specific configurations, such as vSphere FT or some SQL installations; it allocates space on the datastore when the disk is created. Compared to the lazy zeroed format, data on the physical device is zeroed out at creation time. The thick provision eager zeroed format takes longer to be provisioned.

- **Thin provision**: This is used to save space; it's the fastest method to create a new disk. This format doesn't allocate all the requested disk space upon creation. At first, it only uses the space required by the initial operations of the disk, growing in size until the maximum configured size is reached.

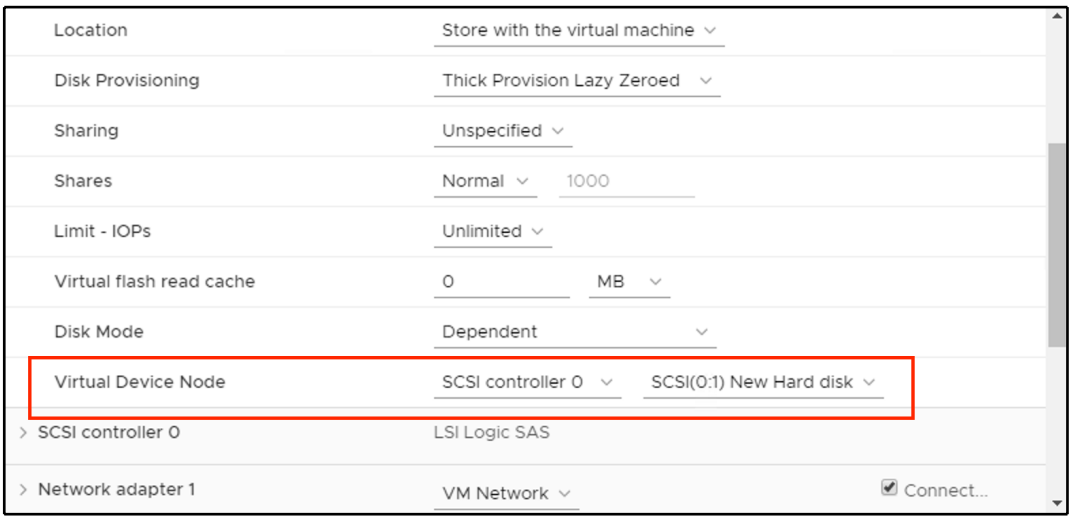
The following screenshot shows the different **Disk Provisioning** types:



Each virtual disk has a disk mode, as follows:

- **Dependent**: Dependent disks are included in snapshots. Snapshots will be discussed in the *Managing VMs* section.
- **Independent-Persistent**: Independent disks act the same as dependent disks, but the writes are committed to the disk immediately and the disks are not affected by the snapshots. Even if you create a snapshot, the data will be directly written to the disk.
- **Independent-Nonpersistent**: Any writes made to non-persistent disks are discarded when you power off or reset the virtual machine.

Virtual disks can only be connected to a single SCSI controller. You can connect an existing virtual disk to a different SCSI controller. For example, once the VM tools are installed, you can assign a new PVSCSI controller, as shown in the following screenshot:



Storage controller

Added by default during VM creation, the storage controller is used to access virtual disks, CD/DVD devices, and SCSI devices. Storage controllers are presented to VMs as different types of storage controllers, such as BusLogic Parallel, LSI Logic Parallel, LSI Logic SAS, VMware Paravirtual SCSI, AHCI, SATA, and NVMe.

Generally, the default controller optimized for best performance is assigned to the VM based on the guest OS selection, the device type, and VM compatibility. A maximum of four SATA, four SCSI, and four NVMe controllers are supported for each VM. If, during the creation of the VM, the Windows Server 2008 or 2012 guest OS is selected, for example, the LSI Logic SAS controller is assigned by the system.

The following list contains the options that are available today (visit `https://blogs.vmware.com/vsphere/2014/02/vscsi-controller-choose-performance.html` for more information):

- **BusLogic**: This was one of the first emulated vSCSI controllers available in the VMware platform.
- **LSI Logic Parallel (formerly known as just LSI Logic)**: This was another emulated vSCSI controller available initially in the VMware platform.
- **LSI Logic SAS**: This is an evolution of the parallel driver to support a new future-facing standard.
- **VMware Paravirtual (also known as PVSCSI)**: This has been designed to support a very high throughput with minimal processing costs. The vSCSI controller is virtualization-aware and is, therefore, the most efficient driver.

When a VM is created, two storage controllers are assigned by default:

- **SATA**: This controller is assigned to access CD/DVD devices and supports up to 30 devices. If you have multiple disks, to distribute the load and improve performance, you can add up to four controllers per VM. An AHCI SATA controller is supported for VMs with ESXi 5.5 and later compatibility. A SATA controller is supported by most guest OSes and is assigned by default to CD/DVD devices.
- **SCSI**: Depending on the guest OS, many VMs have this controller configured by default. A single controller supports up to 15 devices. If you have multiple disks, to distribute the load and improve performance, you can add up to four controllers per VM. In the new SCSI controller, you can enable SCSI bus sharing to allow the virtual disk to be shared by the VM, for example, for building a guest cluster. There are three options available:
    - **None**: The virtual disk cannot be shared
    - **Physical**: The virtual disk can be shared by a VM on the same host
    - **Virtual**: The virtual disk can be shared by a VM on any host

File structure

A VM is composed of several files that typically reside on a datastore in the VMs folder. The VM settings are managed through vSphere Client, but you can also use the command line using PowerCLI, vCLI, or the vSphere Web Services SDK.
The core files that compose a VM are as follows:

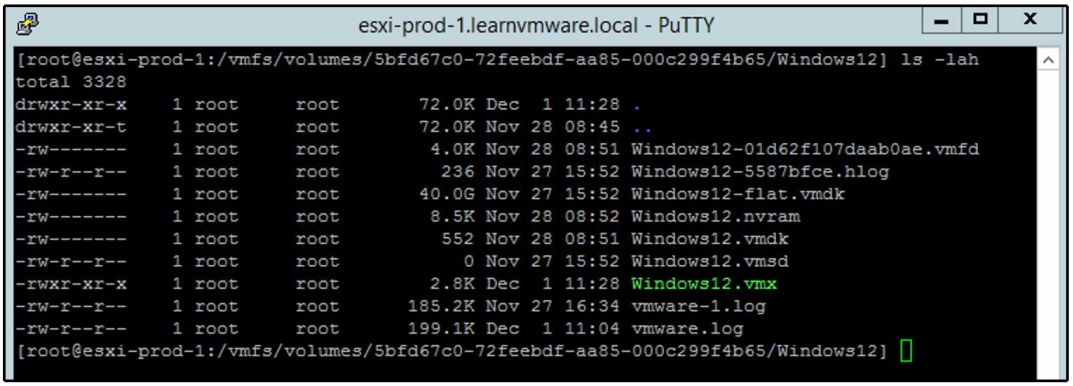
- `.vmx`: This is a plaintext file that stores the configuration of the VM. The file contains information related to the hardware that resides in the VM, such as the processor number, the amount of RAM, the disks, the MAC address, the virtual hardware version, the number of NICs connected, the virtual disk location, and other configurations of the virtual machine, as shown in the following example:

```
config.version = "8"
virtualHW.version = "14"
nvram = "Windows12.nvram"
pciBridge0.present = "TRUE"
svga.present = "TRUE"
floppy0.present = "FALSE"
svga.vramSize = "8388608"
memSize = "4096"
powerType.powerOff = "default"
powerType.suspend = "default"
powerType.reset = "default"
...
```

  The `.vmx` file contains a list of keys and related values that identify the components configured in the selected VM. To determine, for example, the configured RAM or the installed OS in the VM, you need to scroll down the list and identify the keys, memSize, and guest OS, which indicate the requested information. The `.vmx` file is only the configuration file of the VM and doesn't store any data from the guest OS. The virtual hard disk file with a `.vmdk` extension is responsible for storing the actual data of the VM.

- `.vmdk`: Identifies the virtual hard disk of the VM that holds the data of the guest OS instance. A VM can have one or more `.vmdk` files depending on the disks configured in the `.vmx` file. For instance, if you configure disks **C:** and **D:** in a VM running Windows OS, you will have two `.vmdk` files, one for each configured drive.

If you browse the datastore where the VM resides, you can see only a single `.vmdk` file (if the VM is configured with a single drive). Technically, the virtual hard disk is composed of two files with the same extension: a VMDK descriptor and a `flat.VMDK` file. Let's take a look at the roles of these files. The `.vmdk` file is the descriptor file, a plaintext file that contains the configuration information and pointers to the flat file. Generally, the `.vmdk` descriptor file is a small file. `-flat.vmdk` is generally a large binary file that contains the actual data of the VM. Its size is defined in the `.vmx` configuration file. The `.vmdk` file can start from a few GB in size and can grow up to 62 TB (the maximum size supported in vSphere 6.7). To see both the `.vmdk` and `-flat.vmdk` files, you need to access the command line, navigate to the datastores folder where the VM resides, and run the `ls -lah` command, as shown in the following screenshot:



- `.nvram`: This is a binary file that cannot be edited and contains the VM BIOS or EFI configuration. If you delete this file, it will be automatically recreated when the VM is powered on.
- `.log`: This is saved in the same directory as the VM configuration files and contains the logs of the VM activities. It can be used for troubleshooting if
- you encounter a problem. A new `.log` file is created every time the virtual machine experiences a power cycle.

  `.vswp`: For each powered-on VM, two files are used as swap files in case of RAM contentions. The biggest is usually the size of the vRAM of the VM minus the vRAM reservation.

Snapshot-related files will be described in the *Managing VMs* section.

Changing the default file position

By default, all VM-related files are in a single folder with the original VM name (or the VM name after a VM storage migration). You can place the different files in different datastores based on your needs. The following files are part of the virtual machine:

- **VMDK files**: Having virtual disks in different datastores allows you to choose the proper type of disks with the proper performance and service level. You can choose a new location when you add a new virtual disk or choose different locations for each VMDK when you apply a storage migration.
- **Swap file**: Migrating VM swap (`.vswp`) files to a different datastore is possible and described in **KB 2003956**: **Migrating virtual machine swap (.vswp) files from one datastore to another** (`https://kb.vmware.com/kb/2003956`). You can also use an SSD datastore for this purpose, but usually, the need for a different position occurs when storage array replication is used, and you need to avoid swap file replication.
- **Log files**: By default, ESXi/ESX hosts store VMs specific logging in the same directory as the VM configuration files. VM logs can be reconfigured to archive at different intervals, with different names, in different volumes, or when the log reaches a specific size. For more information, see **KB 1007805**: **Locating virtual machine log files on an ESXi/ESX host** (`https://kb.vmware.com/kb/1007805`).
- **Snapshot files**: All files with snapshots are created in the VM working directory, which, by default, is the same directory as that of the VM. The working directory can be changed with **KB 1002929**: **Creating snapshots in a different location than default virtual machine directory for VMware ESXi and VMware ESX** (`https://kb.vmware.com/kb/1002929`).

Virtual machine tools

Virtual machine tools is a set of utilities installed on the guest OS that improves the overall performance and provides better control of the VM, making administration easier. Virtual machine tools is not installed by default.

Although a guest OS can run without virtual machine tools, the management of power controls and other features is not available unless you install virtual machine tools. Shutdown or restart options, for example, are not available without virtual machine tools. An improved graphics interface, better mouse control, and the ability to copy and paste files are some of the main benefits you will notice after installation.

There are three types of virtual machine tools that you can install:

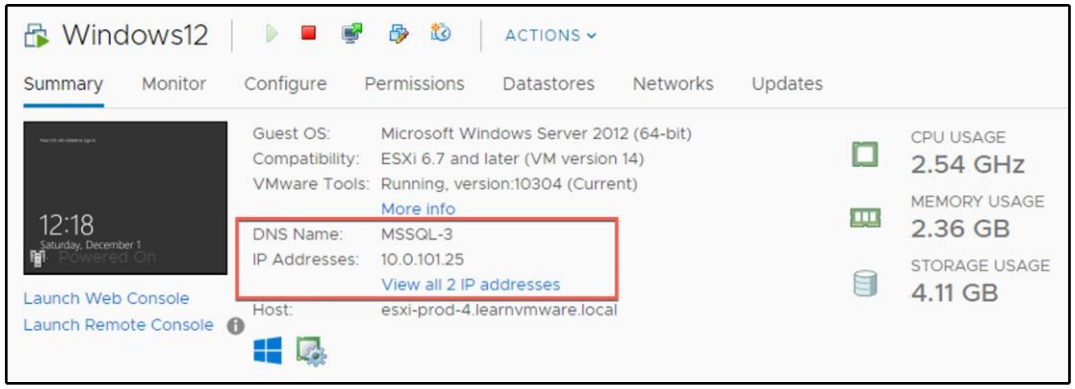
- ISOs (containing installers)
- **Operating System Specific Packages** (**OSPs**)
- **open-vm-tools** (**OVT**)

You can find more information at `https://docs.vmware.com/en/VMware-Tools/10.2.0/com.vmware.vsphere.vmwaretools.doc/GUID-5D9177F3-A098-42F7-B87F-551F61BA434E.html`.

The following list includes some of the features of virtual machine tools:

- Integration with the vSphere suite as a DNS and IP propagation
- Improved network adapter performance (VMXNET driver)
- Improved storage controller performance (PVSCSI driver)
- Smooth mouse experience
- Copying, pasting, and dragging and dropping files
- Improved video resolution
- Improved sound
- The ability to take quiesced snapshots of the guest OS

The **IP Addresses** and **DNS Name** propagated to the vCenter client is shown in the following screenshot:



Virtual machine tools include the following components:

- **VMware device drivers**: This gives you drivers for virtual hardware, including network adapters. Drivers provide smooth mouse operations and improved sound, graphics, and performance.
- **VMware user process**: This gives you the ability to copy and paste text between the VMware Remote console and the host operating system.
- **VMware services**: This handles communication between the guest and host operating system.

For more information on **Virtual Machine Tools**, see the blog post at `https://blogs.vmware.com/vsphere/2017/11/every-vsphere-admin-must-know-vmware-tools.html`.

A recommended practice is to upgrade to the latest Virtual Machine Tools version included in your ESXi. VMware vSphere 6.7 includes Virtual Machine Tools version 10.2.0.

OVT

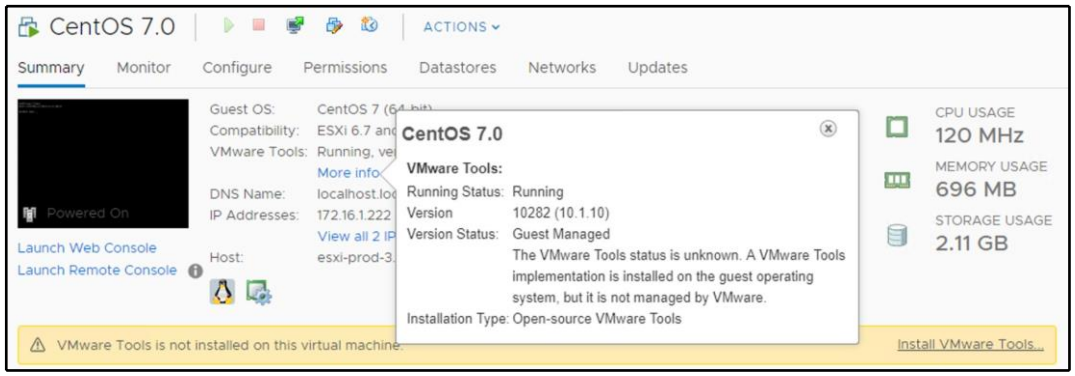
OVT is an open source implementation of Virtual Machine Tools specific for Linux that allows you to bundle the tools into the guest OS, avoiding the management of the Virtual Machine Tools life cycle. OVT is delivered with RPM packages or with `yum` or `apt`. To install OVT in a VM, perform the following steps.

1. Access the system console and run the following command:

```
yum install open-VM-tools
```

   The preferred option is to use the ISO image installation type for Virtual Machine Tools. With Open VM tools, you cannot use the vSphere Update Manager to upgrade the version of the VM tools.

2. To check which version of VM tools is installed quickly, click on **More info** under VMware Tools in the virtual machine overview:



Deploying VMs

The creation of VMs in vSphere 6.7 is a core task, and different methods are available for deployment. The most suitable deployment method to use depends on the goal of the VM, the configuration, and the type of infrastructure the VM will run on.

You can create a VM using the following methods:

- **Creating a VM from scratch**: This is used if you need a VM with a specific configuration, OS, or application, and it's not already present in your environment.
- **Using templates**: If a VM has the requirements you need and is deployed frequently, the use of a template (a master copy of a VM) is a good option to consider. This option requires a minor setup stage after the deployment and allows you to save time. Templates also allow you to further customize the system using guest OS customization templates for supported guest operating systems.
- **Cloning**: If similar VMs are deployed in your environment, the cloning option requires less time than creating and configuring a VM from scratch.

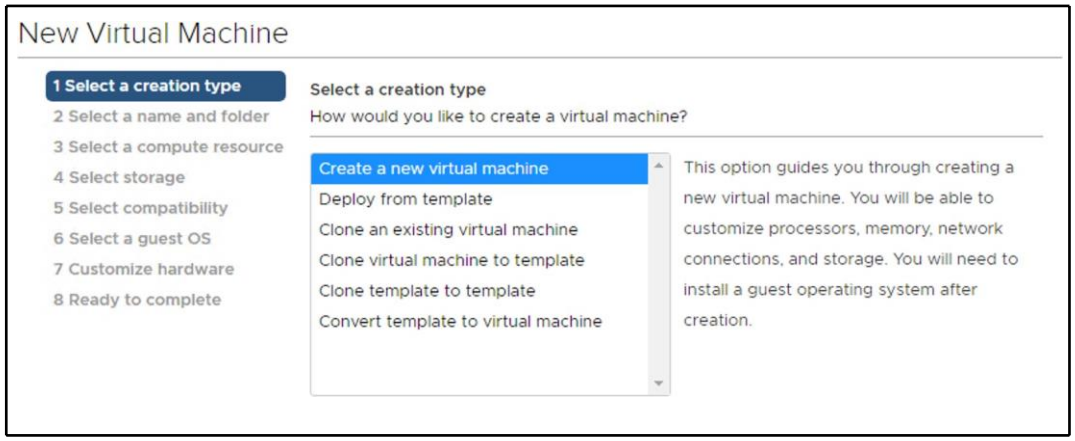
To identify which method is suitable for your environment, let's have a closer look at these options to understand the differences between them.

Creating a new VM

You create a new VM when you need a VM with a specific configuration and a specific OS, and it is not already installed on your virtual infrastructure. When the VM is created from scratch, you can define the virtual hardware to use (CPU, RAM, or a hard disk). The default disk assigned to the VM can be removed and you can add new one, either selected from an existing disk or a new RDM device.

To create a new VM, follow this procedure:

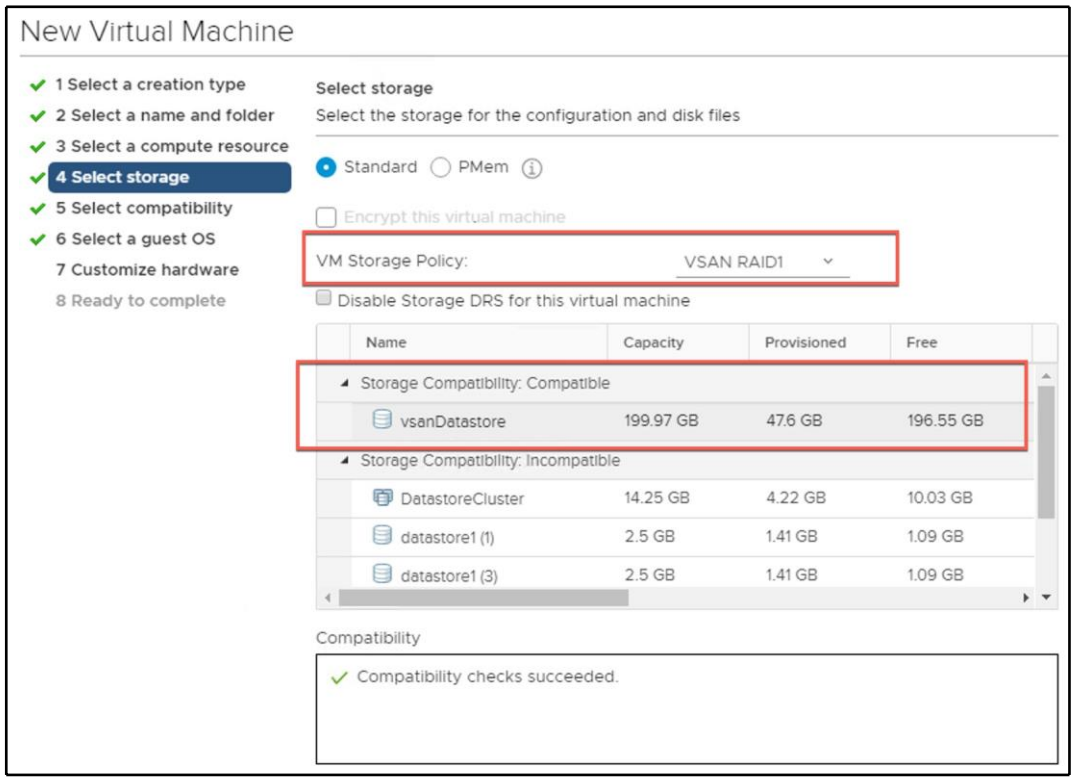
1. From the vSphere Client, access the vCenter Server and right-click a valid parent object from the inventory (it can be a datacenter, cluster, resource pool, or host), then select the **New Virtual Machine** option.
2. From the **New Virtual Machine** wizard, select the **Create a new virtual machine** option to proceed with a new installation:



3. Enter a VM name, specify the location for the VM, then click **Next**. If you place the VM into a cluster with DRS disabled or set in manual mode, you need to specify the host on which to create the VM.

4. Select a computer resource (cluster, host, or resource pool) the VM will access to take the resources and click **Next**. In this step, a compatibility check is performed against the selected location to avoid compatibility issues. If the checks succeed, you can proceed with the next step.

5. Select the datastore or datastore cluster to store the configuration and the virtual hard disk files that meet the VM requirements (performance, size). Make sure you have enough space for VM creation and the operations related to the VM operations (for example, snapshots). If you are using storage policy, only compatible datastores will be listed based on the selected storage policy. Click **Next**:



6. Select the VM compatibility. From the compatible with drop-down menu, specify the version of ESXi the machine can run on. This setting determines the virtual hardware (hardware versions are covered in the following section) available to the VM, such as the available virtual PCI slots, the maximum number of CPUs, and the maximum RAM configuration.
7. Select the OS family (Windows, Linux, or other) and the version of the guest OS the VM will run. The OS selection determines the supported devices and the vCPU number available for the VM.
8. On the **Customize hardware** screen, you have the option to customize the virtual hardware presented to the VM. You can adjust the number of vCPUs to use, specify the amount of RAM, add a new NIC, add a new virtual disk, or remove a device that is not needed (such as a floppy drive), and so on. The VM compatibility settings determine what virtual hardware is available and the configuration maximums. Click **Next** when done.
9. Review the VM settings and click **Finish** to create the VM. Keep in mind that you are creating just the configuration of the virtual machine and no OS has yet been installed. Once created, the VM will appear in the vCenter Server inventory.

Hardware version

The hardware version defines the virtual hardware available to the VM that corresponds to the physical hardware available on the host. vSphere 6.7 introduces hardware version 14 and supports VMs created with previous hardware versions. Each hardware version supports at least five major or minor vSphere releases. By default, the compatibility of the VM is given by the host version on which the VM is created, or by the inventory object on which the default VM compatibility is set.

You might be wondering which hardware version to use. This depends on which version of ESXi your environment uses. If you have multiple hosts with different versions, you should choose the correct hardware version to match the lowest version host used in the infrastructure. However, a lower version will have reduced functionality, and a VMware product won't support a VM with a higher hardware version that is configured with a lower version. If your environment runs vSphere 6.7, you should configure the running VM with the highest hardware version available to take advantage of the latest features.

Different VM versions can be created, edited, and run on a host if the host supports that version. Actions on a host are limited, or the VM might not have access to the host if the VM's configured hardware version is higher than the version supported by the host.
The VM hardware versions can be summarized as follows:

ESXi/ESX version	**Version 14**	**Version 13**	**Version 11**	**Version 10**	**Version 9**
ESXi 6.7	Create, edit, run	Create, edit, run	Create, edit, run	Create, edit, run	Create, edit, run
ESXi 6.5	Not supported	Create, edit, run	Create, edit, run	Create, edit, run	Create, edit, run
ESXi 6.0	Not supported	Not supported	Create, edit, run	Create, edit, run	Create, edit, run
ESXi 5.5	Not supported	Not supported	Not supported	Create, edit, run	Create, edit, run

The chosen version determines not only the hardware available to the VM but also the supported OS. During the deployment of the VM, the OS supported depends on the hardware version configured.

To run Windows Server 2016, you need at least virtual hardware version 10. Otherwise, the Windows Server 2012 guest OS option won't be available.

Setting the default hardware version

By default, the VM compatibility is configured to use the datacenter settings and the host version. In vCenter Server, you can define a default hardware version for VM creation on a host, cluster, or datacenter.

To configure the default hardware version, perform these steps:

1. From the vSphere Client, log in to the vCenter Server, right-click on the object to configure, and select **Edit Default VM Compatibility**.
2. In the **Compatible with** option, using the drop-down menu, select the hardware version to use and click **OK** to confirm. When a VM is created in this cluster, the default compatibility setting is used.

Installing the OS

Once the VM has been created, you need to install the OS as you would for a physical machine. There are two methods available to install the OS on a VM:

- **Using PXE**: You don't need any installation media for this installation type, but the guest OS you install must support PXE installation, and PXE infrastructure must be in place. The VMware vSphere suite does not include the PXE infrastructure, except for the Auto Deploy feature.
- **From media**: You install the guest OS from a CD/DVD media or an ISO image.

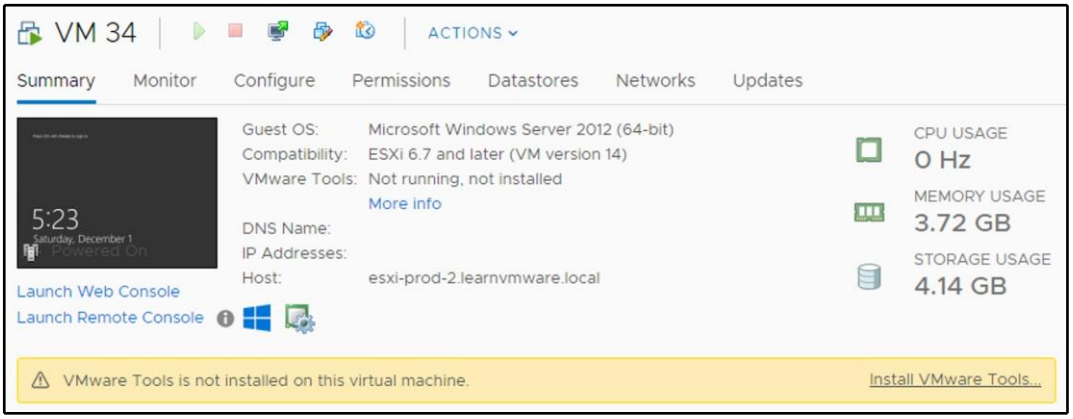
Using an ISO image is generally the fastest method to install a VM OS. Let's do so:

1. Download the ISO image file, then upload the guest OS media to install on a VMFS or NFS datastore that is accessible by the host. Alternatively, you can also use a content library (this will be discussed later in this chapter) to store the ISO image file.
2. From the vSphere Client, right-click on the virtual machine to install and select **Edit Settings**. Access the **Virtual Hardware** tab and expand **CD/DVD drive 1**.
3. From the **CD/DVD drive 1** drop-down menu, select the installation method you want to use from the available options:
    - **Client Device**: The CD-ROM of your machine will be accessed to install the guest O
    - **Host Device**: The CD-ROM of ESXi will be accessed to install the guest OS
    - **Datastore ISO File**: The ISO image file of the guest OS is selected from the datastore to which you previously uploaded the file
    - **Content Library ISO File**: Select the ISO image to mount from the content library (the creation of a content library is discussed in the *Content library* section)
4. Do not forget to check **Connect At Power On** under **Status**. Otherwise, the virtual machine will be equipped by the CD ROM with the associated ISO file (or the client or host device), but, from the perspective of the VM, the media will be ejected. Select the method to use and click OK to confirm.
5. Right-click on the VM to install and select **Power** | **Power On**. Make sure you have set the correct boot order. By default, the virtual machine will try to boot from the disk first. If the operating system is not installed, the second option to use is a CDROM boot followed by a PXE network boot.

Follow the installation options of the guest OS to complete the installation.

Installing Virtual Machine Tools

Although a VM can run without Virtual Machine Tools, VMware highly recommends installing the latest version to enable advanced features (graphic, networking, mouse, storage, and so on). If no VM tools are installed, you will see the following notification in the **Summary** tab:



The installation of Virtual Machine Tools can be performed in three ways:

- **Using vSphere Client**: You can install or upgrade Virtual Machine Tools on a single VM at a time.
- **Using VUM**: If more VMs need to install or upgrade Virtual Machine Tools, you can automate the process using VUM (VUM will be covered in *Chapter 12*, *Life Cycle Management, Patching, and Upgrading*).
- **Using other tools**: You can also use tools such as a Linux repository or a standalone version of Virtual Machine Tools, which is downloadable from the Driver and Utilities tab at `my.vmware.com`.

To install the Virtual Machine Tools, follow these steps:

1. From the vSphere Client, right-click the running VM to process and select **Guest OS | Install Virtual Machine Tools** to mount the disk image in the virtual CD/DVD of the VM
2. Access the guest OS and proceed with the installation

The installation takes a few seconds and may require a reboot of the VM. A quick way to perform the installation is by clicking the Install Virtual Machine Tools link from the warning message in the **Summary** tab of the VM.

Virtual Machine Tools is included in the ESXi distribution and the bundled tools' ISO image files are located in the `/locker/packages/` directory. If you want a central repository in a shared datastore, take a look at **VMware KB 2129825**: **Installing and upgrading the latest version of Virtual Machine Tools on existing hosts** (`https://kb.vmware.com/kb/2129825`).

Cloning a VM

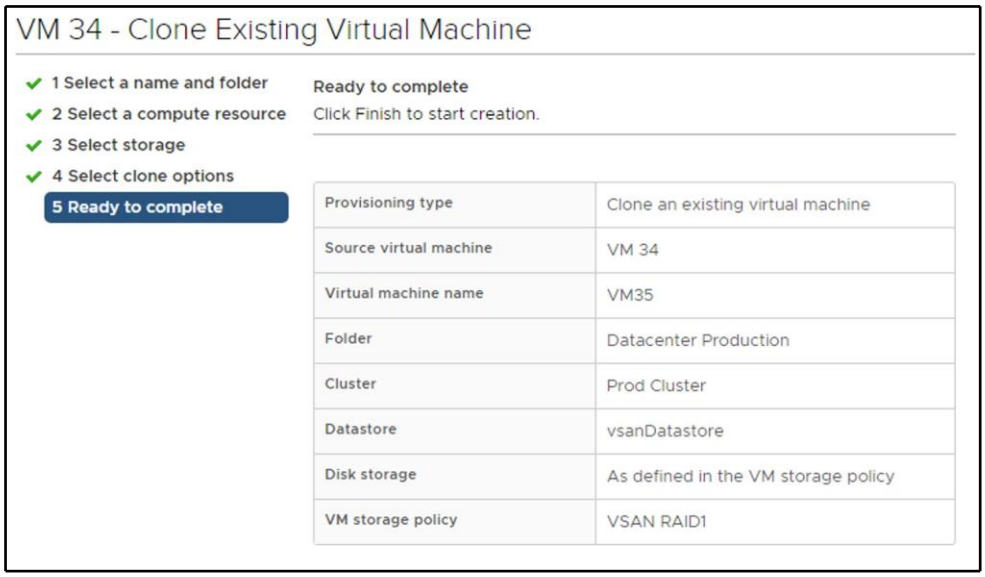
A VM deployed by cloning another VM creates an exact copy of the original VM. **Cloning** is the fastest method to deploy a new VM if an existing VM has the same features and applications you need for the new installation. When using the **Clone** option, the new VM will have exactly the same configuration as the source one (for instance, the same IP address will be configured within the guest OS or hostname). A new VM on the vSphere level will, of course, have a different UUID or MAC address.

You can clone both running or powered-off virtual machines, but a new virtual machine will always be powered off, since the clone operation clones only the virtual machine files, not the state (content of the virtual memory).

This procedure allows you to save time during deployment because you simply need to clone and configure a few parameters.
To deploy a new VM by cloning an existing one, follow these steps:

1. From the vSphere Client, log in to the vCenter Server and access the inventory view. Right-click the VM to clone and select **Clone | Clone to Virtual Machine** to create a new VM.
2. Enter the name of the virtual machine and select the location in which to deploy the VM, then click **Next**.
3. Select a computer resource to allow the VM to access the resources of the selected object. If the compatibility checks succeed, click **Next** to continue.
4. Select the storage in which to store the configuration and disk files. Make sure you have sufficient space in the selected datastore. Specify the virtual disk format and click **Next**.

5. In **Select clone options**, you can customize the guest OS to prevent conflicts (a duplicate computer name or IP address already in use) and automatically power on the VM once it is deployed. Click **Next**.
6. In the Summary window, click **Finish** to begin the cloning process of the selected VM:



Clone to Template will clone an existing virtual machine to the new virtual machine template available in the Virtual Machines and Templates view. Clone as Template to Library will clone the source virtual machine and place the template in the content library.

Deploying a VM from a template

The deployment of a VM from a template is performed by creating a new VM from a copy of a template configured with specific virtual hardware and software. Template deployment is the recommended option if you need to deploy several machines with the same requirements. Proceed with these steps:

1. Go to the **VMs and Templates** inventory view and right-click on the template from which to deploy the new VM.
2. Select **New VM from This Template** to create a new VM based on the selected template.

3. Specify a name of the VM and specify the location in which to place the VM by selecting a datacenter or folders, depending on your organizational needs. Click **Next**.
4. Select a computer resource to allow the VM to access the resources of the selected object. If the chosen location causes compatibility issues, a warning message is displayed in the compatibility area. If the checks succeed, click **Next** to continue.
5. Select the datastore in which to store the VM files. You can specify the format of the virtual disk (we talked about the disk format in the *Virtual disks* section) you want to configure, then click **Next**.
6. In Select clone options, you can customize the guest OS to prevent conflicts due to a duplicate computer name or IP address and automatically power on the virtual machine once deployed. The guest OS customization allows you to modify the computer name, license, and network settings. When the desired option has been selected, click **Next**.

In the **Summary** window, click **Finish** to deploy the new VM based on the selected template.

There are multiple options for how to create a template:

- **Convert an existing (powered-off) virtual machine to a template**: In this situation, a source virtual machine will be removed from the inventory and a new template will be available. If you operate on the **Hosts and Clusters** view, the template will not be visible. You need to switch to **VMs and Templates** to access your templates. The source virtual machine will be kept on the original datastore with all its data, the only difference is that the `.vmx` configuration file will be renamed as `.vmtx`.
- **Clone an existing VM (powered off or powered on) to a template**: The source virtual machine will be kept intact and the new virtual machine will be transformed to the template.

You cannot power on a template. If you need to upgrade your template, you need to convert the template back to the virtual machine, perform the necessary upgrade (for instance, patching the guest OS), and convert it back to the template.

VM customization Specifications

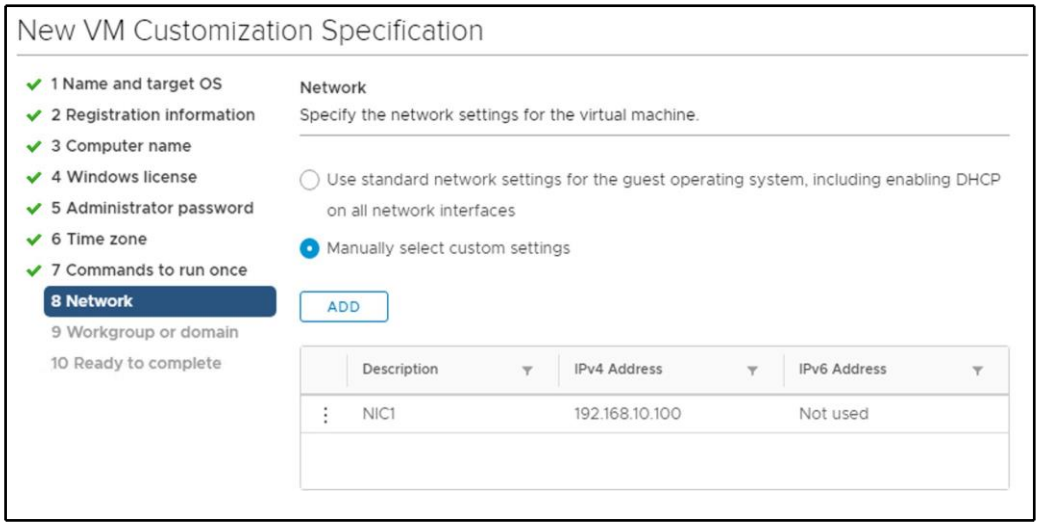
Usually, you do not need to run two copies of a virtual machine, instead you should perform additional customizations to the new virtual machine so that it has its own customized configuration.
By customization, we are referring to the following tasks (depending on the guest OS type):

- Changing the IP address
- Changing the hostname/computer name
- Changing the administrator/root password
- Setting the time zone
- Joining the computer to Active Directory Domain
- Running several initial scripts

To create a new VM customization specification, switch to **Policies and Profiles** and select VM customization specifications:

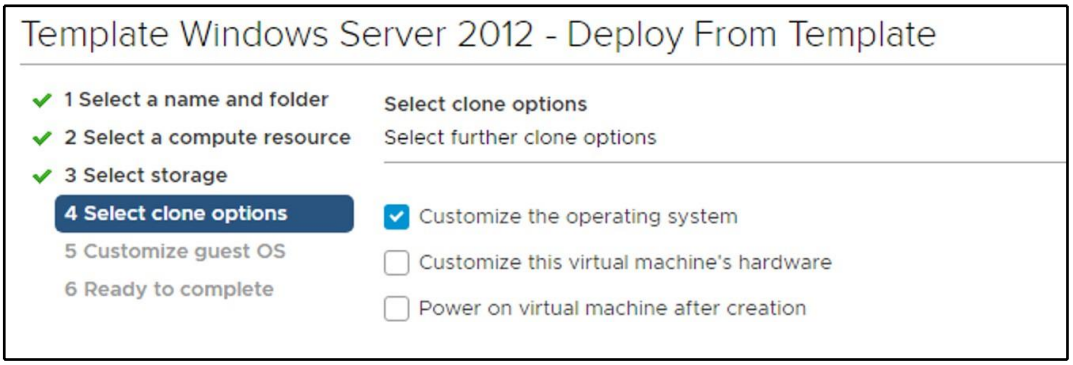
1. Configure the name of the customization specification and the target guest OS. Based on the selection, different options will be available for the customization. If you select Windows as the target guest OS, the following options will be available:
    - **Use custom SysPrep answer file**: The SysPrep answer file is used for the actual customization of the guest OS. If you do not select this option, you will manually configure the desired parameters in the **New VM Customization Specification** wizard. If you have an existing SysPrep file, you can use it instead of manual configuration. For more information about SysPrep, take a look at the official documentation: `https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/use-answer-files-with-sysprep`.
    - **Generate a new security identifier** (**SID**): With the change SID option, all of the deployed virtual machines can acquire a unique **security identifier** (**SID**). A unique SID is required when joining a VM to the active directory.
2. The owner name and the organization name will change the `RegisteredOwner` and `RegisteredOrganization` registry keys in `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion`.

3. The computer name can either be provided during the clone/deploy wizard, it can be a fixed name in the VM customization specification, or a new name will be assigned based on the name of the virtual machine on the vSphere level.
4. You can directly assign a Windows license if you need to. If you do not assign the license, you will need to activate the guest OS once deployed.
5. You need to provide a new password for the local administrator account. You can choose whether the virtual machine should be logged in automatically once deployed.
6. You might need to change the time zone of the guest OS.
7. Invoke several commands inside the guest OS once deployed. You can either use native commands of the guest operation system, such as `netsh` for disabling the network interface, or you might use a more complicated script located inside the source template that will be invoked, for example, to extend the disk size.
8. In the network section, you can assign an IP address to the network interface. Note that based on the customization specification, the target virtual machine needs to have the exact number of virtual network adapters, as defined in the customization specification. If your customization specification includes settings for two NICs, the new virtual machine that will be deployed and customized by such a customization specification must also have two virtual NICs. The **Network** section of the customization profile is shown in the following screenshot:



9. You have the option to automatically join this new virtual machine to the AD domain.
10. Once all inputs are filled in, you can review the settings and proceed with the creation of the VM.

Once the customization specification is created, you can choose the **Customize the operating system** clone option:



In the next step, all your available customization specifications will be displayed and, based on your selection, the new virtual machine will not only be cloned from a template but also customized with the configuration stored in the customization specification:

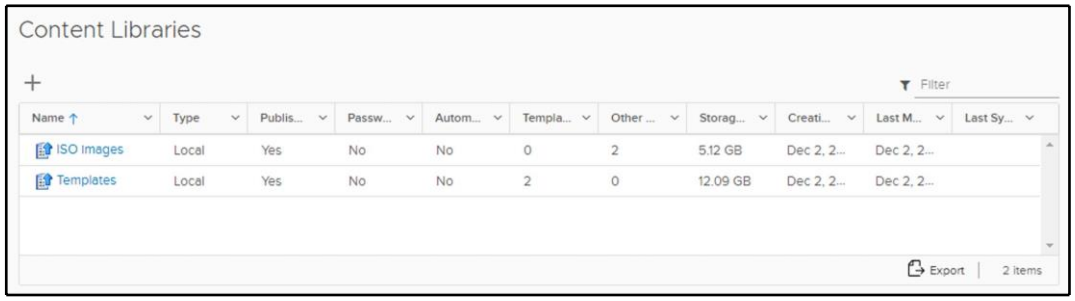
Content library

A content library is a container object to store templates, vApps, or other files that can be shared across multiple vCenter Server instances in the same or different locations to ensure consistency and compliance within the infrastructure.

vSphere 6.5 introduced new features and some enhancements that improve performance and recoverability. You can now mount an ISO directly from the content library, apply a guest OS customization during VM deployment, and update existing templates. The content library is included in the vSphere backup/restore service as well as the VCHA feature set (from VMware vSphere 6.5).

A VM template, a vApp template, or another type of file in a library is defined as a library item that can contain single or multiple files (ISO, OVF, and so on).

You can define multiple content libraries, and during the configuration, you specify on which datastore the content library will be stored. In this example, I have created two local content libraries. The first one is for ISO images and the second one holds all templates and external OVF files:



As you can see, each of the content libraries created its directory in the datastore, and the actual files are stored under folders with a UUID, not the name of the items.

Creating a content library

You can create two types of content library:

- **Local**: This is used to store items on a single vCenter Server instance that can be published to allow other users from other vCenter Servers to subscribe to it.
- **Subscribed library**: This is created when you subscribe to a published library and can be created in the same vCenter Server as the published library or a different vCenter Server instance.

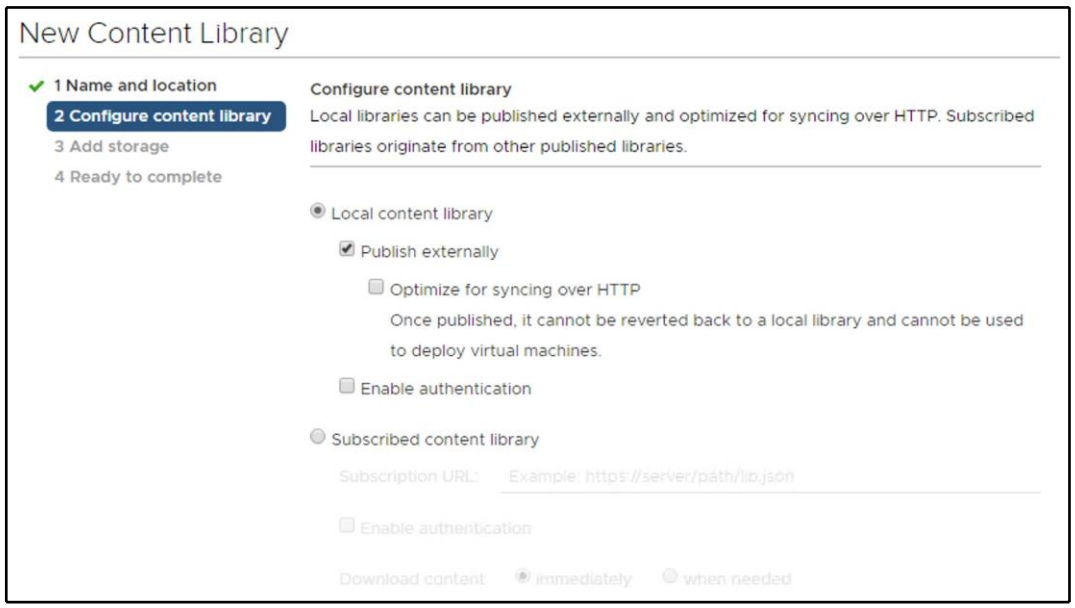
If the subscribed library is created in a different vCenter Server, the option to download all contents or metadata can only be configured in the **Create Library** wizard. To keep the content of a subscribed library up to date, the subscribed library automatically synchronizes to the source published library on a regular basis. Synchronization of the subscribed library can also be done manually.

Local content library

Before you can subscribe to an existing library, you need to define the Local Content Library.

To create a Local Content Library, proceed as follows:

1. From the vSphere Client, access the vCenter Server and, from the menu, select **Content Libraries** and click on the create a new library icon with the + sign to open the create library wizard.
2. Enter a name and a description in the note field, then click **Next**.
3. Specify the type of content library you want to create (local or subscribed), then click **Next**. Select **Publish externally** to make the content of the library available to other vCenter Server instances. If you want the users to use a password when accessing the library, select **Enable authentication** and set a password. Check the **Optimize for syncing over HTTP** checkbox to create an optimized published library. This library is optimized to ensure lower CPU usage and faster streaming of the content over HTTP. This library is used as a central content depot for subscribed libraries:

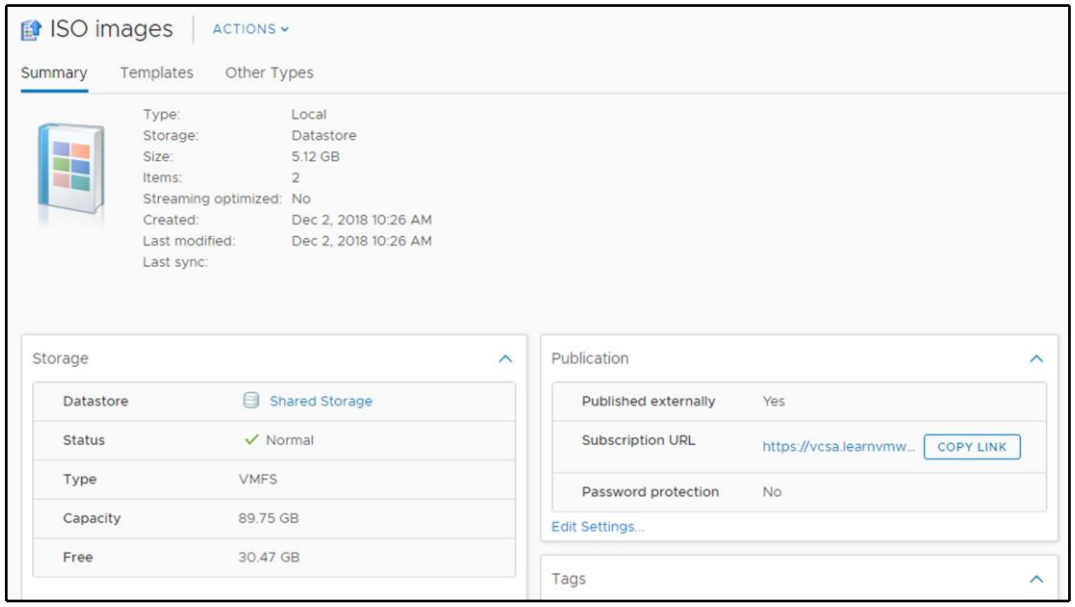


4. Select the datastore used to store the library's content and click **Next**.
5. Review the settings and click Finish to create the library.

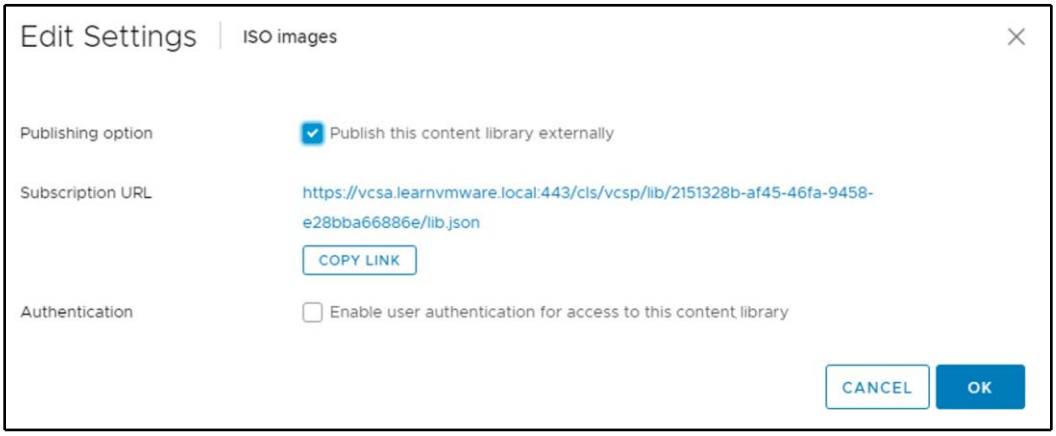
Subscribed content library

A subscribed content library can synchronize its content with other content libraries. The idea is that you maintain only one content library (for example, in HQ) and all subscribed libraries will synchronize their content from the HQ without additional manual operations:

1. To subscribe to a content library, you need to have at least one content library defined on another vCenter server with the **Published Externally** flag set to **Yes**, as seen in the following screenshot:



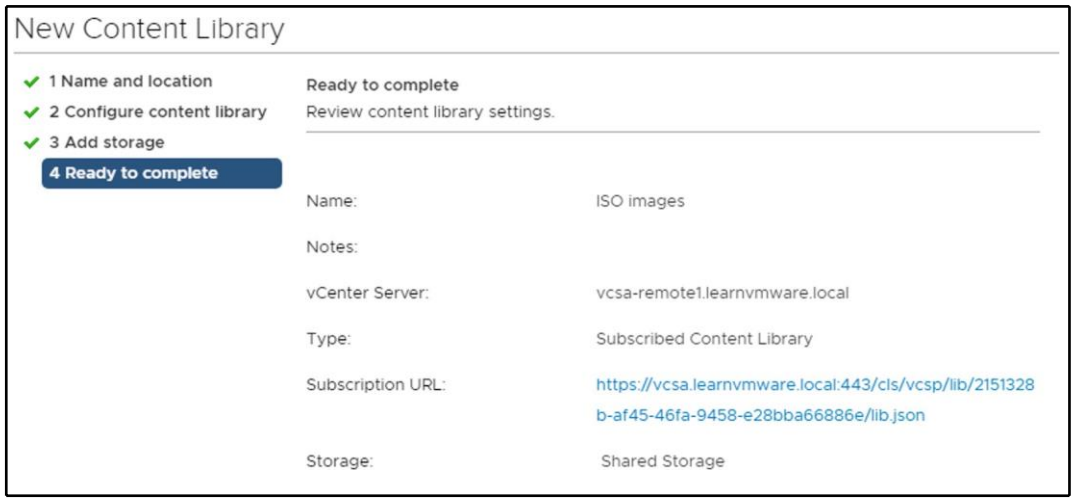
2. During the configuration of a subscribed library, you need the **Subscription URL**. You can quickly click the **COPY LINK** button to get the URL, or you can click on **Edit Settings** to access the information:



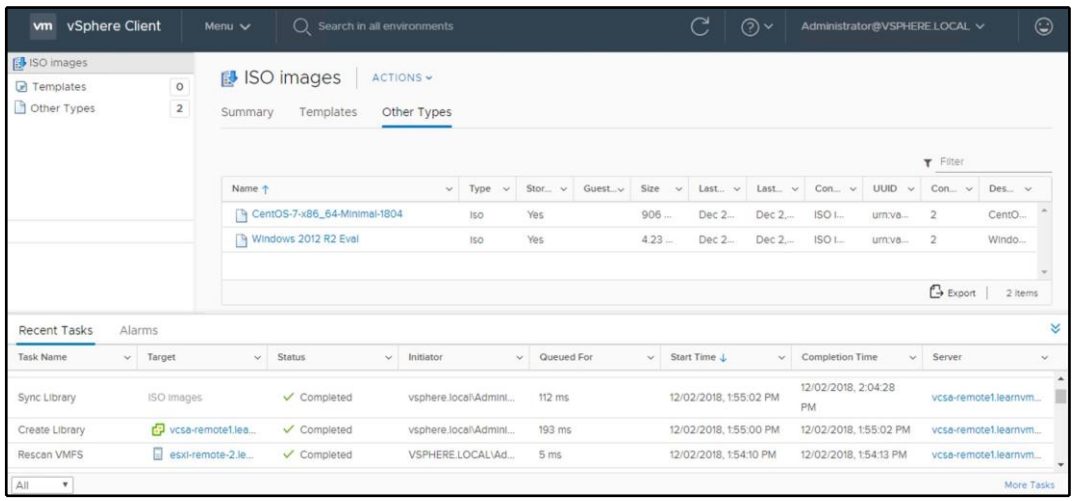
It is possible to have multiple Local and Subscribed Content Libraries in a single vCenter Server.

To create a new subscribed content library, perform the following tasks:

1. In a different vCenter server (vCenter does not need to be part of the Linked-Mode), switch to **Content Libraries** and create a new content library.
2. Provide a name for the new content library and click **Next**.
3. Select the subscribed content library option, provide a subscription URL, and provide authentication credentials if the authentication is configured. You have the option to either download all content automatically, or only download the metadata and download the item from the source content library once the item is accessed.
4. The files will be stored on a datastore, so select which datastore will back up the content library.
5. Confirm the configuration on the review screen:



Once the content library is created, it will synchronize automatically from the source content library, and once the synchronization is complete, you will see the same content in the subscribed content library as in the original one:



If you need to manually sync the content library (automatic synchronization is enabled by default) you can invoke manual synchronization from the **Actions** menu.

You can find more details about synchronization intervals and timeouts at `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-1A5A5387-0E5C-4158-9836-2544990EED00.html`.

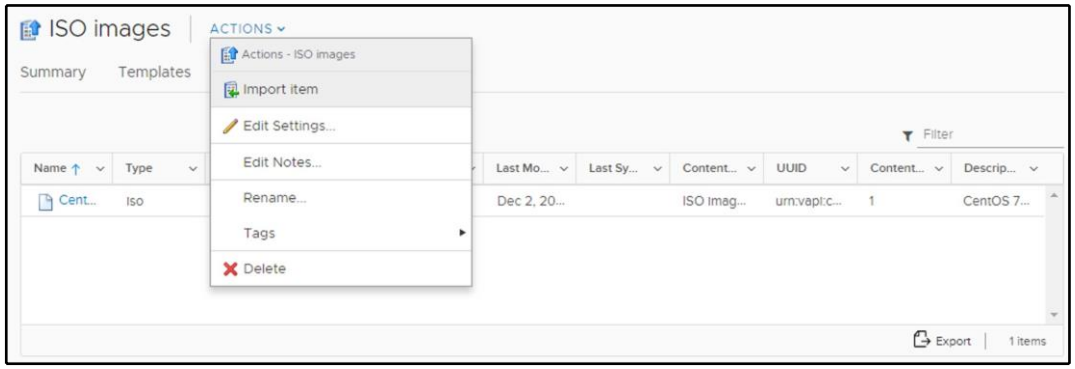
Working with the content library

Content Libraries provide centralized access to all your ISO files, OVF templates, vApps, and any other files. Several tasks can be performed with content libraries. Let's walk through a few options.

Uploading ISO images

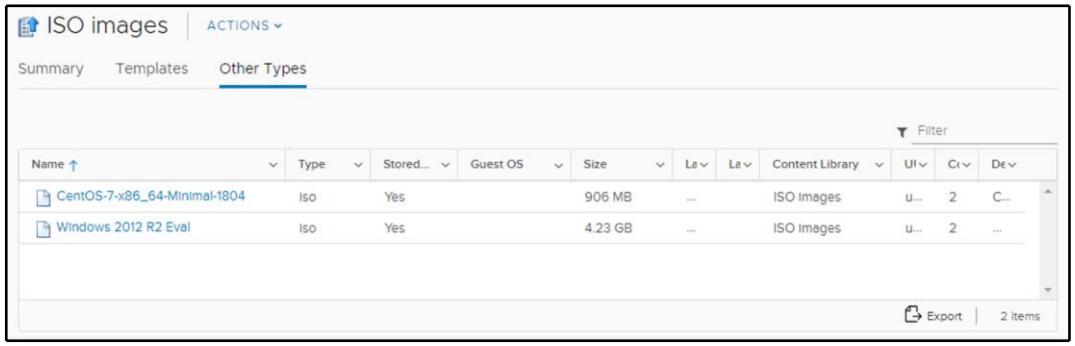
To upload ISO files, perform the following operations:

1. Right-click on the created content library and select the **Import item** option to import content to the library:



2. You can import content by specifying a URL or a local file. If you import the content from a local file, you locate the file using the **Browse** button. You can also edit the name of the item to identify the file better. Click **OK** to import the required item.

You can see what content is available in the library in the following screenshot. ISO images are stored under **Other Types**:

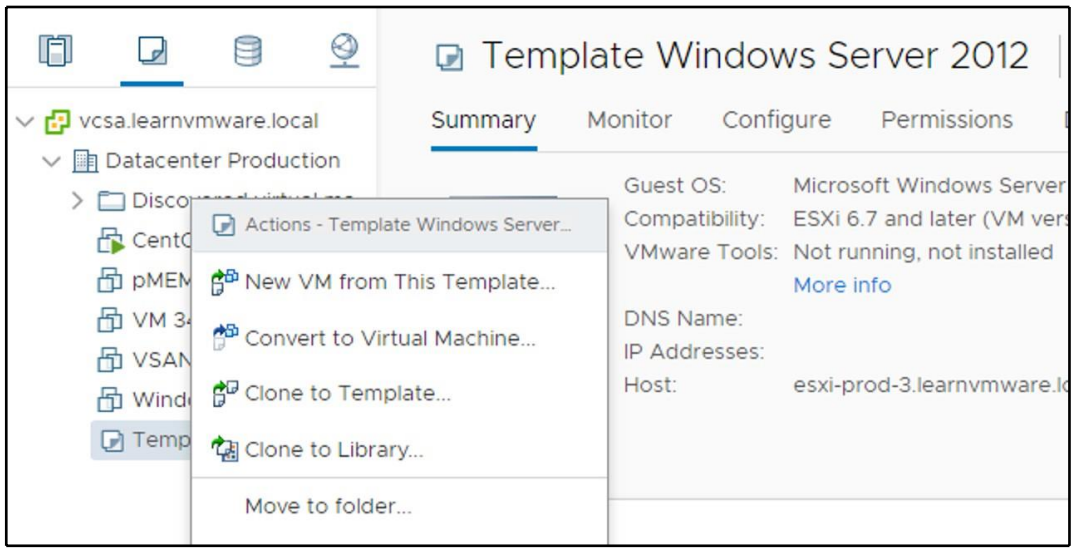


Uploading templates and OVF files

Existing templates can be cloned to the content library. You can also upload any OVF/OVA file, such as third-party appliances.

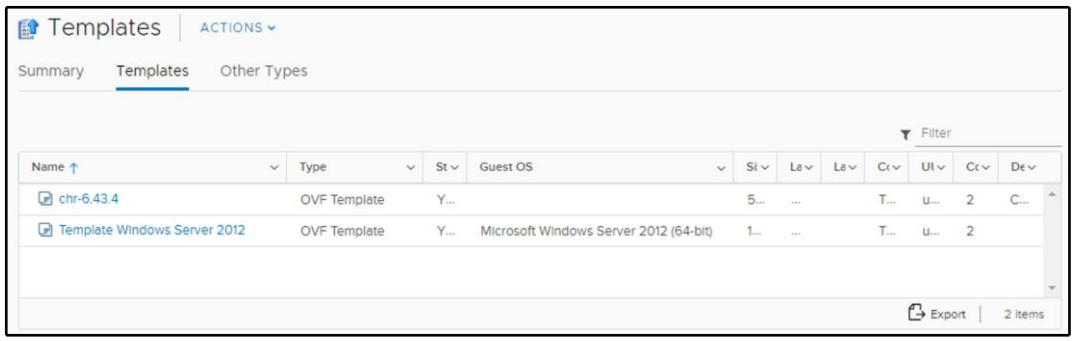
If you have an existing template in your inventory, you can clone it to the content library. The original template will not be removed from the inventory, and the clone will be available in the content library. Let's get started:

1. Right-click on the existing template and select **Clone to Library...**:



2. Specify in which content library the template should be cloned.
3. Once the wizard is closed, a new task will be launched. The existing template will be exported as an OVF file and uploaded to the content library.

Once the clone process is finished, you will see your new OVF template under the **Templates** tab in the content library. If the content is cloned from the template, you will also see the guest operating system version. For external OVF files uploaded to the content library, this information will not be available:



Deploying VMs from the content library

Once the OVF file is located in the content library, you can deploy a new VM from the content library. You cannot use guest OS customization scripts when the virtual machine is deployed from an OVF file in the content library:

1. Right-click on the datacenter or cluster and select **New Virtual Machine**. In the wizard, select **Deploy From Template**.
2. If the content library is configured, you will have the option to select the template from the content library.
3. Provide a name for the new VM and select in which datacenter it should be deployed.
4. As with any new virtual machine, select on which ESXi host or Cluster the VM will be deployed.

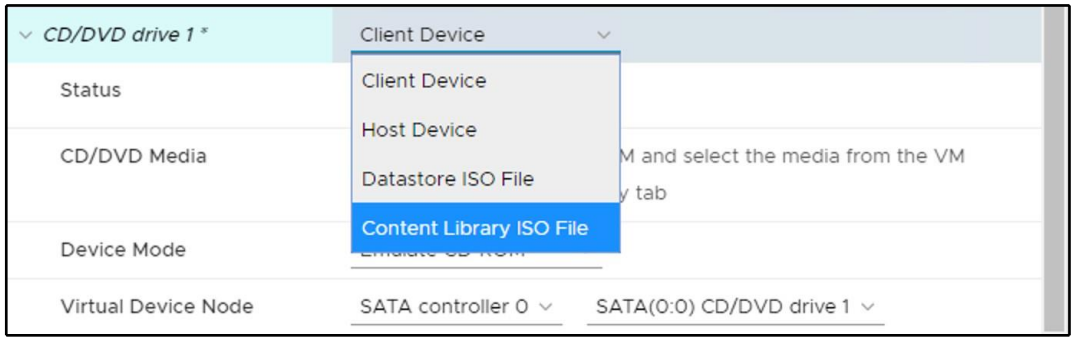
5. As items in the content library are OVF files, based on the selected item, you will see the details from an OVF manifest. If the template had been cloned from a template, no details would be provided, but if the template is a third-party OVF image, you might see different pieces of information here:

6. Choose on which datastore the new VM will be deployed
7. Change the mapping of the virtual network interface card. On the left, you have an original port group that was assigned to the template during creation. On the right, you can choose a target port group**.**
8. Review the settings and Confirm the deployment. Once the deployment starts, you will see deploy OVF template task running in the inventory.

ISO files from the content library

If you choose to use a content library to host your ISO installation images, you can easily access them when creating a new virtual machine:

1. Edit the settings of a virtual machine, select **CD/DVD drive 1,** and then select **Content Library ISO File**:



2. Select which ISO file you want to mount
3. Confirm the selection and close the edit settings to mount the ISO file to the virtual machine

Managing VMs

When VMs have been deployed in your infrastructure, you can start the administration using the available tools and features offered by vSphere Client. Several actions can be performed on VMs to keep a clean inventory and a healthy infrastructure. Let's take a look at some common procedures an administrator performs on a regular basis.

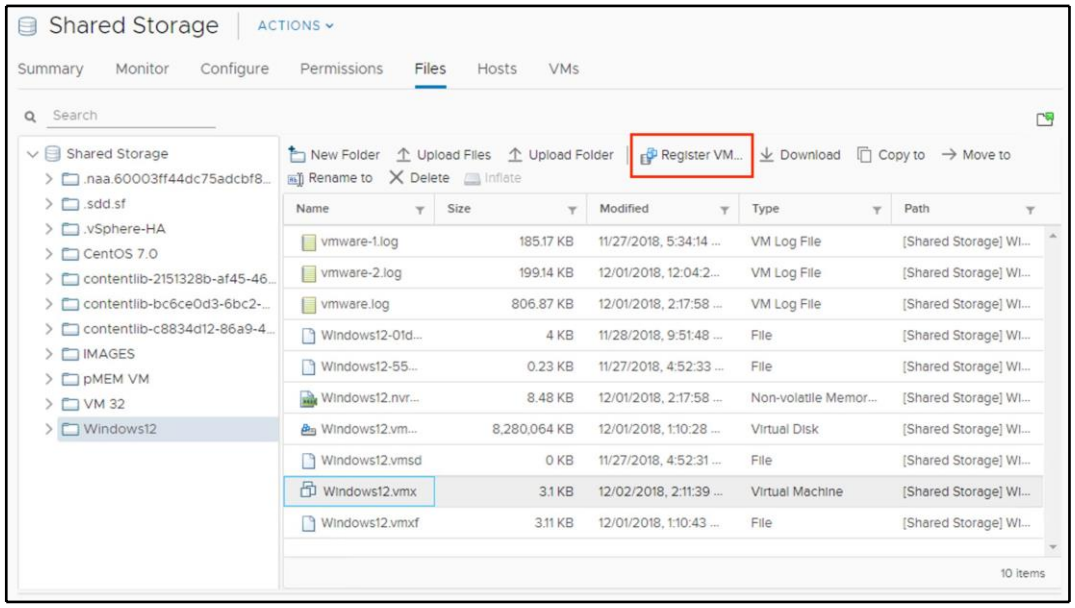
Adding or registering an existing VM

VMs can be created or deployed using the different methods shown previously. In some circumstances, you might need to put in your production environment a pre-created VM from another source. You may be wondering how to deploy this virtual machine.

First, using vSphere Client, you need to upload the VM files (generally the `.vmx` and `.vmdk` files) to an attached datastore that is reachable by the hosts. When the files are in the datastore, you have to register the VM to add it to the vCenter Server or the ESXi host inventory. Once the VM has been added to the inventory, you can start using and managing the VM.

To register a VM to the inventory, follow this procedure:

1. From the vSphere Client, log in to the vCenter Server and select the **Storage** view.
2. Select the storage and the folder in which the VM has been stored.
3. From the available files in the selected folder, select the file with the `.vmx` extension and click **Register VM...** to register the VM to the inventory:



4. By default, the system populates the virtual machine name field, reading the info from the `.vmx` file. Enter a different name if you want to change the default value. Specify a location in which to run the VM and click **Next**.
5. Select the compute resource the VM will access to get the resources. If the compatibility checks succeed, click **Next** to continue.
6. When you are ready to complete, click **Finish** to register and add the VM to the inventory.

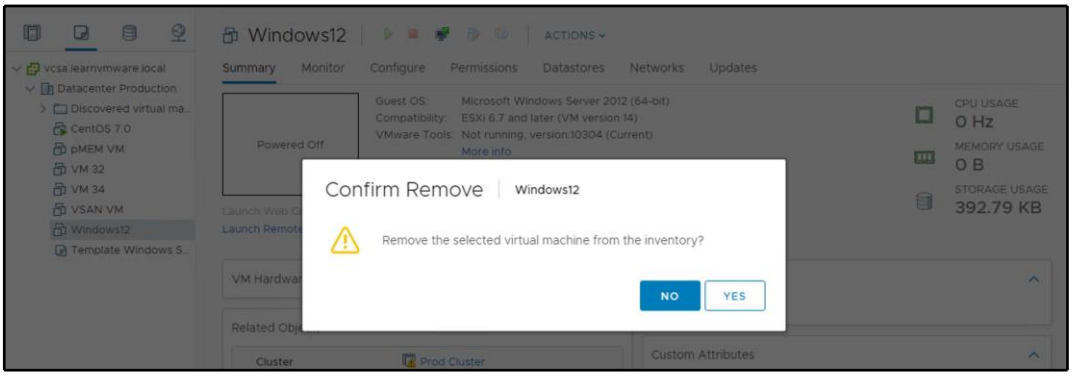
When the VM has been added to the inventory, you can power it on and manage it as you would do with other VMs.

Removing or deleting a VM

Removing and deleting a VM are two different but straightforward procedures that lead to different results. Removing a VM from the inventory doesn't delete the VM (the files remain in the same location in storage), but removes its view from the inventory, and it won't be listed anymore. Removing a VM from the inventory can be useful if you want to remove a no-longer-used VM, but you want to keep the data.

To remove a VM, the VM must be powered off. The procedure is quite simple:

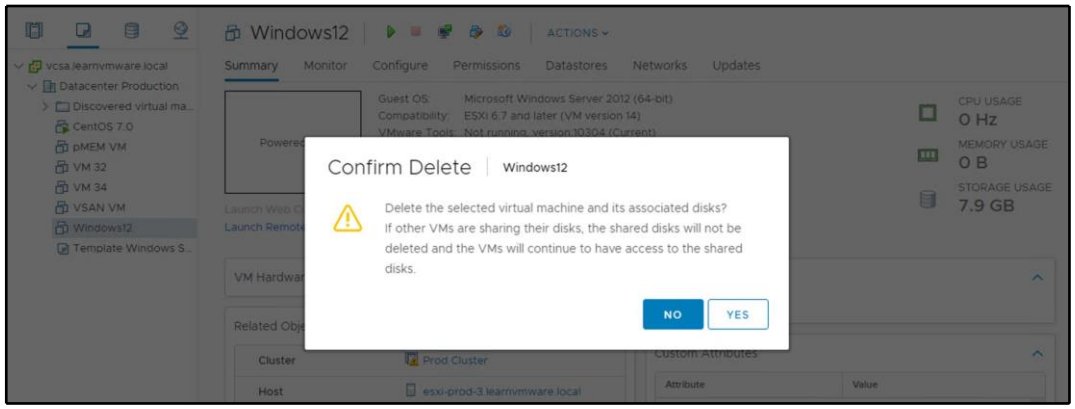
1. Right-click on the VM to remove and select the **Remove from Inventory** option.
2. Click **YES** to confirm the removal:



When you delete a VM instead, all VM files are removed from the datastore with no way to recover them if the deletion is done by mistake. *How do we recover a VM that has been deleted accidentally?* From the backup.

The deletion procedure is similar to what we just saw:

1. Right-click the powered-off VM (this option will be grayed out if the VM is still running) to delete and select the **Delete from Disk** option.
2. Click **YES** to confirm the deletion:



Managing the power state of a VM

In vSphere 6.7, you can change the VM power state in different ways. To change the power state, right-click the VM and select **Power** followed by the type of the state. You have the following states available:

- **Power On** and **Power Off**: This function powers the VM on or off immediately without any interaction with the guest OS. Be careful when powering off the VM, because the process doesn't perform a clean shutdown of open files and there is the risk of corrupting files that are not closed properly.
- **Suspend**: This feature suspends the VM, freezing its current state. When the VM is resumed, it starts from the state that was suspended.
- **Reset**: This command emulates the reset button of a physical computer.
- **Shut down guest OS**: This is the correct command to use to shut down the guest OS since it avoids data corruption. This function is available if Virtual Machine Tools is installed on the VM.
- **Restart guest OS**: This command is available only if Virtual Machine Tools is installed on the VM and it allows a graceful restart of the guest OS.

Managing VM snapshots

A snapshot takes the state of a VM at a specific point in time and allows you to revert to that state whenever you like. You can have several snapshots in a VM and, depending on the changes that have occurred, you may decide to keep the changes by deleting the snapshots, or to discard the changes by reverting to a previous snapshot.

A snapshot is taken on a per-VM basis and can be used for different situations. When a new patch is released from a vendor for the guest OS running on a VM, if something goes wrong during the upgrade process, the VM can become unresponsive and sometimes the blue screen of death may be displayed, in the case of a Windows guest OS. If the guest OS can't be recovered, the backup is the only lifeline you have that allows a quick recovery of the VM. If the failure occurs on a core VM and the process of restoration from a backup takes a long time, the users will not be happy because the services won't be available for a while.

Taking a snapshot before applying a patch is a trick that allows you to immediately revert to a working state of the VM before the patch was applied, with limited service disruption.

However, the use of snapshots has some limitations:

- Raw disks and **Raw device mapping** (**RDM**) physical mode disks are not supported, RDM with virtual compatibility mode is supported
- Independent disks are supported only if the VM is powered off
- VMs configured for bus sharing are not supported
- You can have a maximum of 32 snapshots in a chain, and a single snapshot should not be kept for more than 72 hours to avoid the snapshot storage location running out of space
- Keeping snapshots for a long time may negatively impact the performance of the VM
- For disks larger than 2 TB, snapshot creation can take a long time

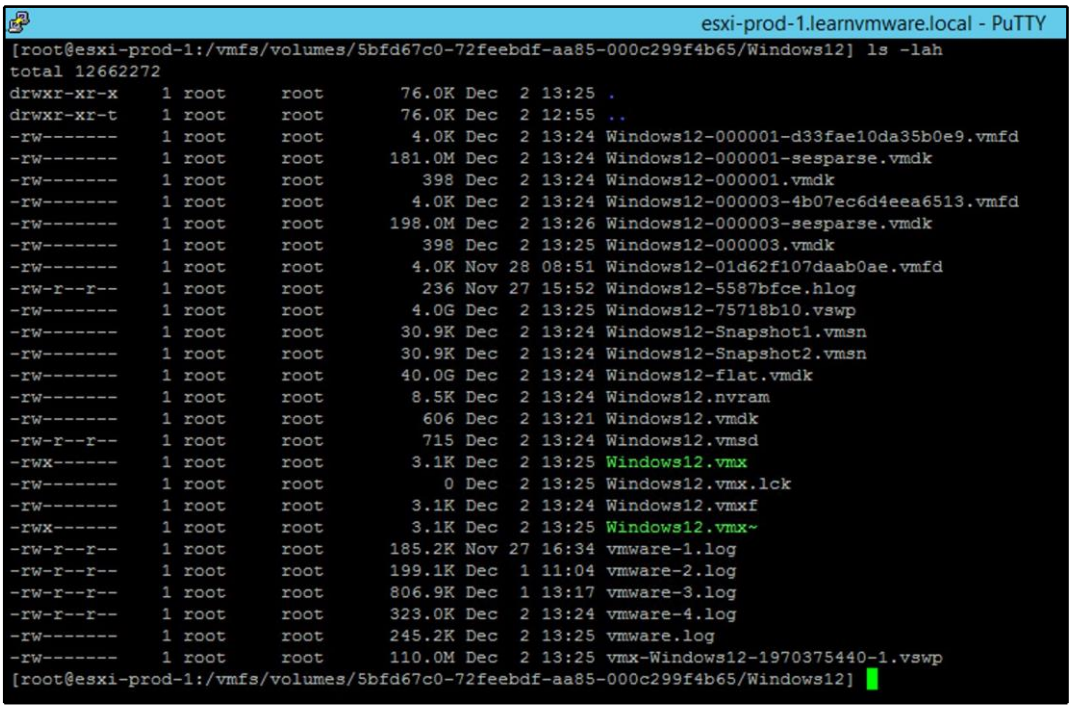
Snapshots should not be used as a backup because if the files of the VM are lost, or the storage itself fails, the snapshot files are lost as well.

Creating a snapshot

To create a snapshot, follow these steps:

1. From the vSphere Client, right-click on the VM you want to process and select **Snapshots | Take Snapshot**.
2. Enter a name and provide a description. If the VM is powered on during snapshot creation, you have the option to snapshot the virtual machine's memory (grayed out if the VM is off). If this option is enabled, the RAM of the VM is also included in the snapshot. The quiesce guest file system option, which is only available if Virtual Machine Tools is installed, brings the on-disk data into a state that is suitable for backups, ensuring that backups are consistent and work as appropriate.
3. Click **OK** to take a snapshot of the selected VM.

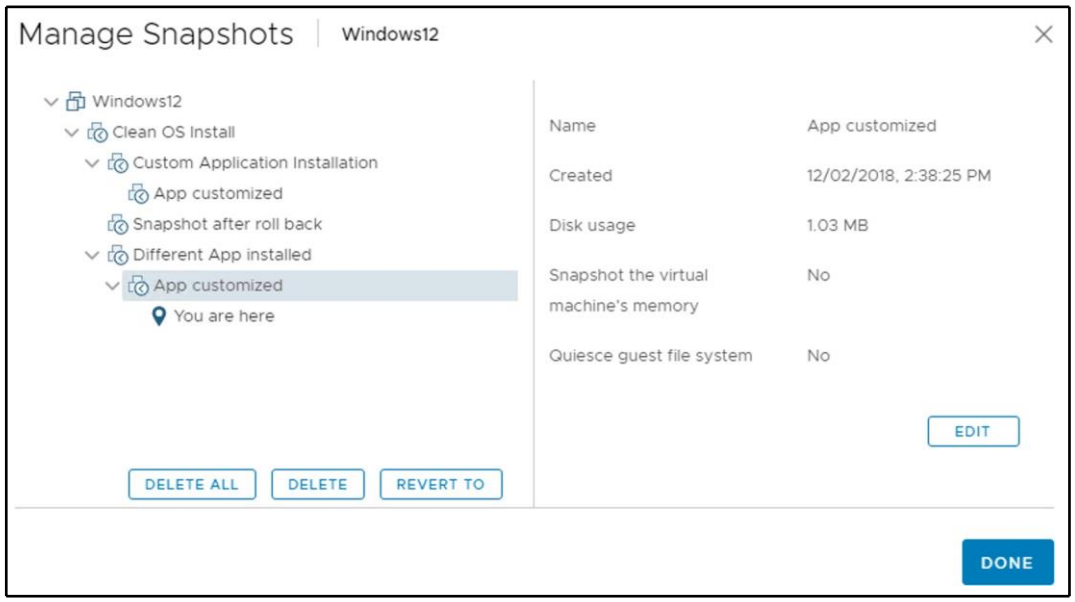
When a snapshot is taken, multiple new files are created in the VM folder. These include `.vmdk`, `-sparse.vmdk`, `.vmsd`, and `.vmsn`, as shown at the following screenshot:



Let's have a look at the files created when a snapshot is taken and their roles:

- `vmname-00000#.vmdk`: This is a text file that contains info about the snapshot and snapshot disks. For every snapshot taken, this file is created for each of the `.vmdk` files.
- `vmname-00000#-sparse.vmdk`: This is the delta disk file that represents the difference between the current state of the VM and the state at the time of snapshot creation.
- `vmname.vmsd`: This file holds snapshot information such as names, descriptions, and relationships between snapshots.
- `vmname.snasphot#.vmsn`: This stores the memory state of the VM when the snapshot is taken, and it is created each time you take a snapshot.

You can have a complex snapshot tree based on your requirements, and a snapshot chain does not need to be linear, as shown at the following screenshot:



Reverting to a snapshot

If you need to, you can quickly revert to a previous snapshot using **REVERT TO** from the Snapshot Manager. If you revert to a snapshot, every change performed between the last snapshot and the current state will be discarded.

Committing changes

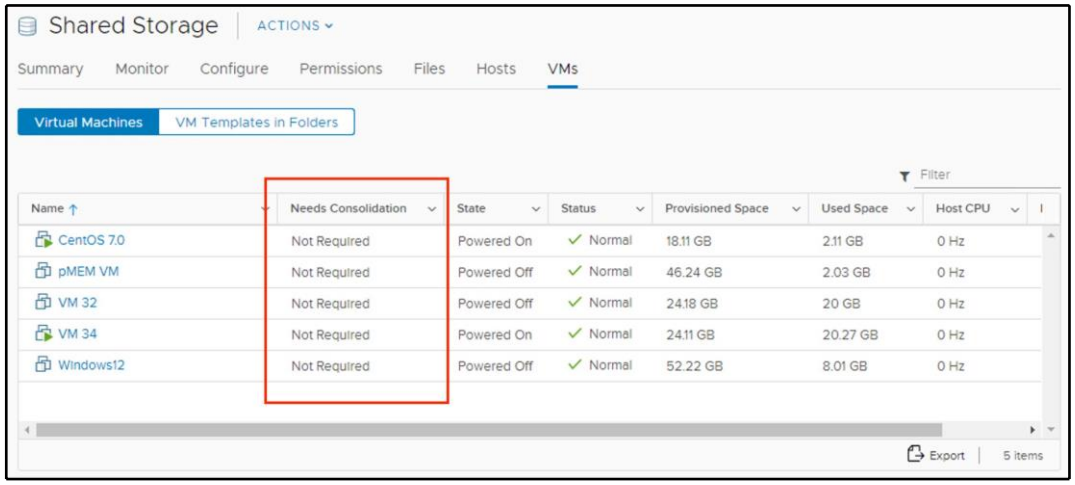
To commit changes and the current state of the VM, delta disks are merged with the base disks. This operation is done using the DELETE option in the Snapshot Manager:

- **DELETE**: Deletes the selected snapshot from a chain, consolidating the changes that occurred between the state of the snapshot and the previous disk state to the parent snapshot.
- **DELETE ALL**: All snapshots are deleted from the VM, consolidating, and writing the changes occurred between snapshots and previous delta disks to the base disks, merging them with the base VM disks. DELETE ALL is an alias for committing all changes, and your virtual machine will merge everything to the base disk. The result is that the VM will have only the base disk with the current running state.

Snapshot consolidation

Snapshot consolidation is a procedure that can be used when the delete or delete all operations fail. For example, consolidation may be required if the backup software that utilizes the snapshot technology is not able to remove redundant delta disks. If the snapshots are not removed, the VM performance may suffer, and the storage could run out of space. By performing a consolidation, these redundant delta disks are removed, keeping the VM in a healthy state.

To determine whether a VM requires consolidation, from the vSphere Client, select the vCenter Server, cluster, or host, and then click the **VMs** tab. If the **Needs Consolidation** column is not visible, click the arrow on the right side of the column head, select **Show/Hide** columns, and check the **Needs Consolidation** option:



Importing and exporting VMs

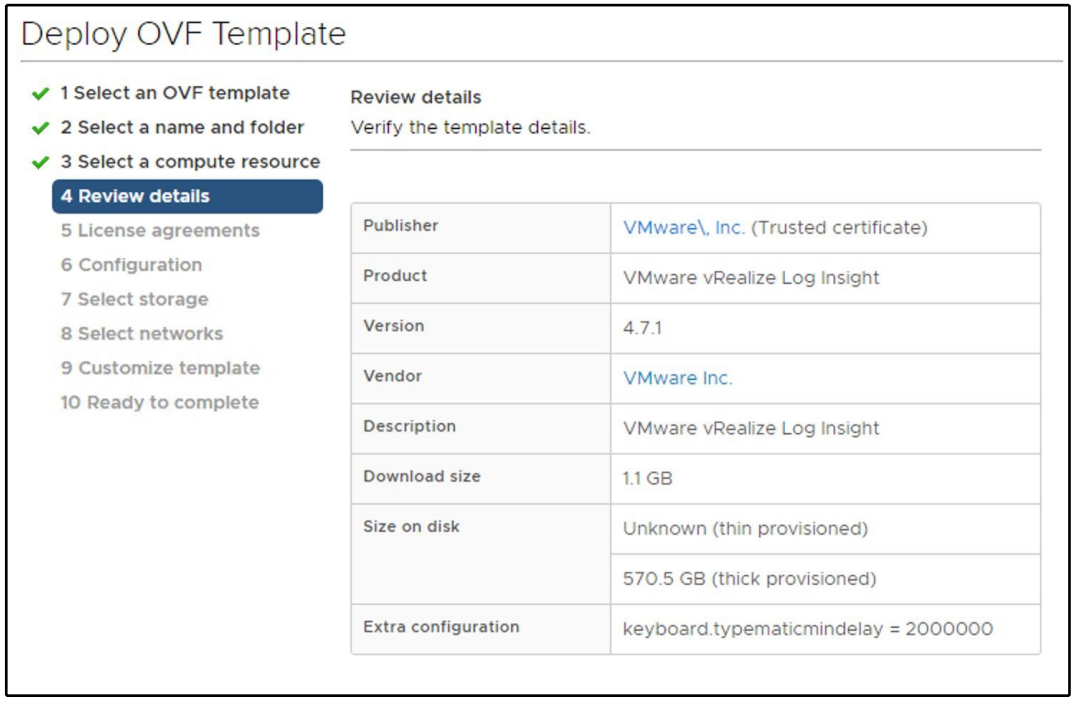
The vSphere infrastructure allows you to import and export virtual machines. VMs that are deployed or exported from the inventory are usually referred as OVFs or OVAs.

Deploying Open Virtual Format (OVF) and Open Virtual Appliance (OVA) templates

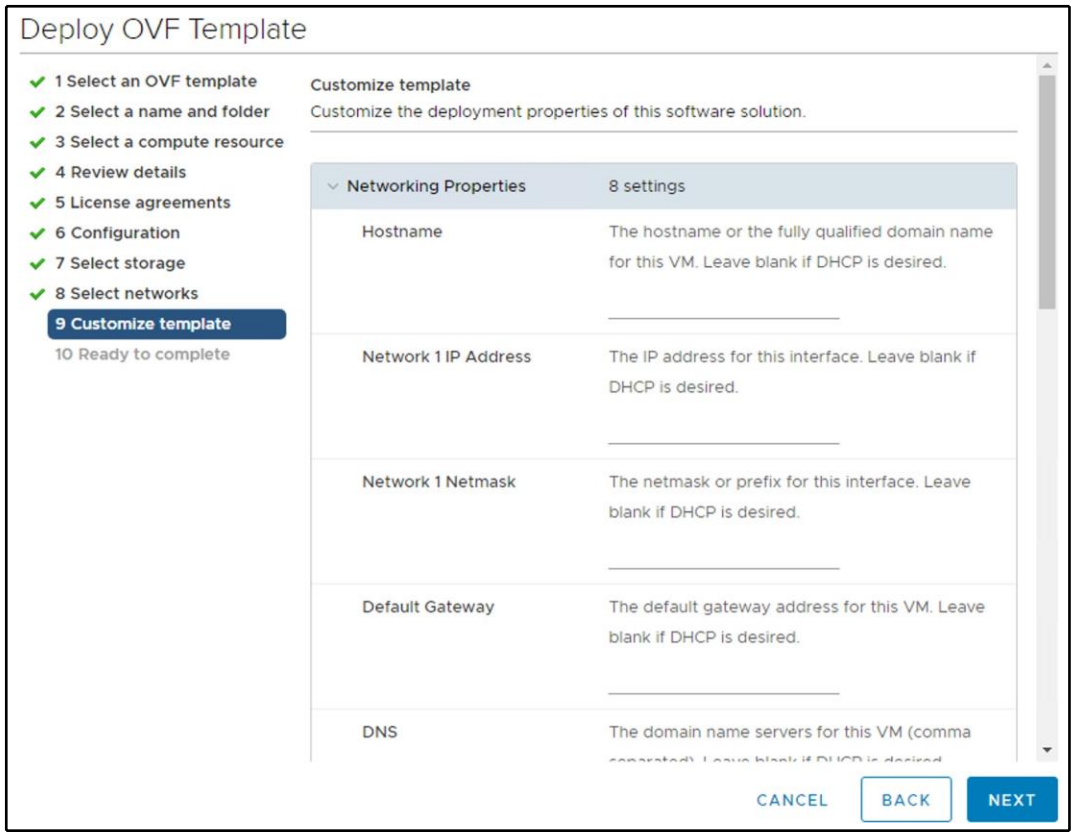
Virtual machines can be exported in OVF and OVA formats and deployed in the same or different environments. OVA and OVF are compressed file packages that enable faster deployment and may contain more than one VM. From vSphere 6.5, the installation of the CIP is no longer required to import and export OVF or OVA templates.

The procedure to deploy a VM from an OVF or OVA file is similar to deployment from a template:

1. From the vSphere Web Client, right-click a valid inventory object (host, datacenter, cluster, or resource pool) and select the **Actions | Deploy OVF Template** option.
2. Click the **Choose Files** button to specify the `.ovf` file to use, or you can deploy the OVF directly from the URL.
3. Enter a name and select a location to deploy the VM to. Click **Next**.
4. Select the resource to run the deployed appliance and click **Next**. A validation check is performed.
5. Review the details to verify the configuration is correct. Click **Next** to define the storage to use.
6. Depending on the OVF, additional steps might be displayed, including the License Agreement and Configuration:



7. Under **Configuration**, you can perform additional configuration of the new VM. Those configuration options are based on the source OVF file and the author of the OVF.
8. Specify the virtual disk format and select the location in which to store the files. Click **Next** to continue.
9. Select the network to use from the **Destination Network** drop-down menu, then click **Next**.
10. In the **Customize template** section, you might be able to assign variables that will be slipstreamed to the new virtual machine for its initial configuration. Again, these variables are defined by the author of the OVF/OVA file:



11. Click Finish to begin the deployment of the VM.

OVA and OVF files are commonly used to deploy third-party virtual appliances without a complete installation of some specific applications. Usually, specific deployment scripts are built into the virtual appliance, so all you need is to provide the initial configuration details in the OVF import wizard.

Exporting a virtual machine and an Open Virtual Format (OVF)

The OVF template can also be used to export a captured state of the VM in a compressed and sparse format. The procedure to export an OVF template requires that the VM is powered off before proceeding:

1. From the vSphere Client, right-click on the VM to export and select the **Template** | **Export OVF Template** option.
2. Specify the virtual machine name and, optionally, an annotation that can be useful to identify the VM configuration better. If you need to include additional information or configurations, such as BIOS UUID or MAC addresses, check Enable Advanced Options. Be careful if you enable these options because the portability will be limited. Click **OK** to proceed with the export.
3. Specify where to save each file associated with the template.

You can use OVF Export to transport your virtual machines between different environments or for cloud migrations. The OVF file has a generic structure so different service providers can deploy virtual machines exported as OVF appliances.

Converting VMs

There are some situations in which you might need to convert a physical machine into a VM or import a VM from a third party to take advantage of the scalability, reliability, security, and features provided by the vSphere platform. If you have old physical machines or physical machines running specific applications, OSes, and configurations that require time for a fresh reinstallation, and service downtime for an extended period is not tolerated, conversion to a VM might solve the problem.

You may also be requested to import VMs created for different virtual platforms in vSphere, and to run those VMs in a VMware environment, but they must be converted into a supported format.

To migrate the OS, applications, and data to the virtualization platform, the **VMware vCenter Converter** tool (available to download from the VMware website at `https://www.vmware.com/products/converter.html`) is the solution you should use to import a physical machine or VM into the vSphere environment.

You can perform two types of conversion:

- **Physical to virtual** (**P2V**)
- **Virtual to virtual** (**V2V**)

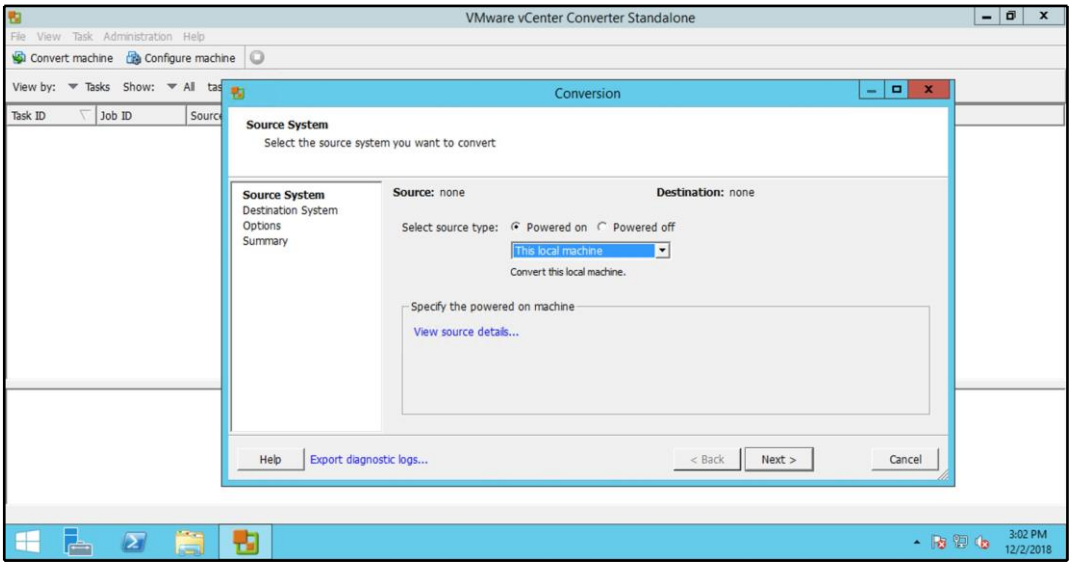
P2V conversion

P2V conversion is a procedure used to convert a physical computer into a VM. VMware vCenter Converter allows conversion from physical machines running Windows and Linux and supports both desktop and server editions. With the supported hot cloning feature, admins can convert a running machine in a non-disruptive way, with no downtime or reboot requirements.

To convert a physical server into a VMware vSphere machine, you can use VMware Converter:

1. Download and install VMware Converter on the source physical server

2. Click the **Convert machine** button and select **This local machine** as a source:



3. In **Destination System**, select vSphere infrastructure and provide an IP address or FQDN of the vCenter Server as well as the credentials.
4. Specify the name of the virtual machine and its location.
5. In the **Destination Location**, select on which cluster or ESXi server you want to deploy the virtual machine.
6. Options allow you to customize the P2V migration. You can, for example, define to which port group the NICs will be connected or how the virtual hardware will be defined for the new VM.
7. In the **Summary** tab, check the properties of the migration and, once you hit **Finish**, the migration will start.

Once the task is completed, you can access your converted server from the vSphere infrastructure.

Note that any changes to the source system during the migration will not be captured and migrated.

V2V conversion

V2V conversion refers to the migration of an OS, application programs, and data from a VM or disk partition to another VM or disk partition. VMware vCenter Converter supports the conversion from third-party VMs, such as Hyper-V and KVM to vSphere.

Other products can be used for V2V migrations:

- **StarWind V2V Converter**: `https://www.starwindsoftware.com/starwind-v2v-converter`
- **Acronis Backup and Recovery Solutions**: `https://www.acronis.com/en-au/virtualization/`
- **5nine V2V Easy Converter**: `https://www.5nine.com/5nine-v2v-easy-converter/`

14
VM Resource Management

Maintaining a resource-optimal vSphere infrastructure is a critical day-to-day operation and should be performed with a strict focus on delivering adequate resources to the **virtual machines** (**VMs**) at any given time.

The resources of your vSphere infrastructure are limited, even though vSphere provides many overcommitment techniques so that you can assign more resources than you physically have, but you should try to avoid contention scenarios at all costs because such contention can significantly affect your applications' and workloads' performance.

One of the fundamental techniques that you can use to provide the best possible performance to your VMs is resources, limits, and shares, which you can use to fine-tune resource allocation to different vSphere objects, such as VMs, vApps, and resource pools.

Using vMotion, you can freely move your workloads within a vSphere cluster, allowing you to utilize the ESXi hosts evenly.

For more complex environments, you can also utilize **Distributed Resource Scheduler** (**DRS**), a cluster feature that is not only responsible for maintaining your cluster balance automatically but also provides advanced functions that allow you to specify how the VMs should be run concerning different affinity and anti-affinity rules.

Resource pools, on the other hand, can provide you with a pool of computing and memory resources that VMs inside the resource pool can consume without taking more than you have defined, and by using vApps you can even extend this functionality to complex application management, where you treat multiple VMs as a single logical application.

This chapter covers the following topics:

- Virtual machine resource management
- Virtual machine migration
- DRS
- Resource pools and vApps
- Network and storage resources

Virtual machine resource management

The number of VMs that can run on ESXi is not infinite, and optimization of resources ensures the best performance. In contrast to the physical world, where each server is often equipped with more resources than it needs, in a virtualized environment, you can allocate suitable resources to a VM based on its role and function.

An FTP server, for example, doesn't need to be equipped with a dual processor and 6 GB of RAM because the resources will be underutilized. By allocating a suitable amount of RAM and a suitable number of CPUs, you can obtain the best performance, saving resources for other VMs. Understanding how to manage and reallocate resources is then a key way to avoid overcommitment of resources (that is, when you have more demand than the available capacity), which can compromise the entire infrastructure's functionality.

Hosts and clusters (a group of hosts where the cluster owns the overall CPUs and RAM), as well as datastore clusters (a group of datastores), provide physical resources to the infrastructure. Default settings configured on a VM during creation are generally suitable, but sometimes may not ensure the correct allocation of resources. You can always edit the VM settings later on to adjust assigned resources in order to avoid issues due to lack of resources.

Reservations, limits, and shares

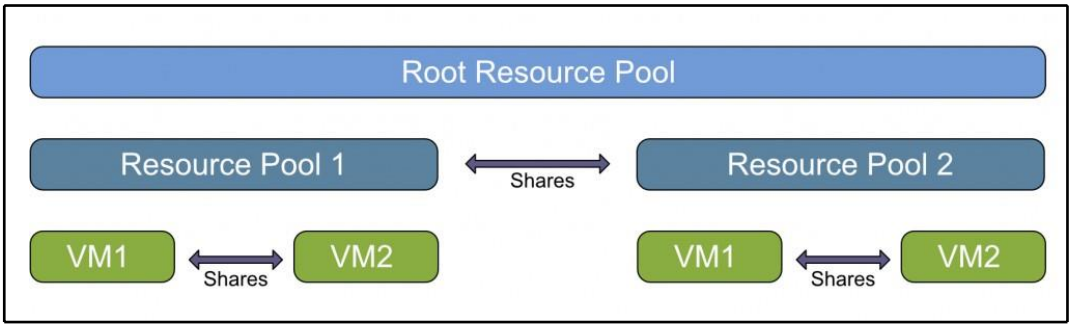
Not all VMs are the same. Some of them are used for business-critical workloads, some of them might be used for internal workloads, and some of the might be only some development and test VMs, and because of that, you want to treat them differently.

You should guarantee that your critical line-of-business application has some resources that are always available, but on the other hand, that your development and test VMs will never consume more than a certain amount of resources.

The most straightforward way to manage the resources of VMs is reservations, limits, and shares. We have already touched on these topics in previous chapters when we talked about network I/O control, but let's have a look at them in more detail.

Shares

Shares specify the priority of a VM to get resources during a period of contention. When resources in an ESXi host are limited, and the VMs compete to access resources, the VMs configured with higher shares will have higher priority to access more of the host's resources. Shares can be specified as high, normal, or low, with a ratio of 4:2:1, and are applied between siblings in the vSphere hierarchy.
If you do not use a resource pool or vApps, all your VMs will be on the same level in the hierarchy, and thus the shares will be split between all of them, as you can see in the following diagram:



Keep in mind that the shares are only applied during contention. When there is no contention, they are not applied.

Reservations

Reservations specify the minimum allocation guaranteed to a VM. When the VM is powered on, the ESXi hypervisor assigns resources based on the specified minimum reservation regardless of whether the physical server is heavily loaded. Resources are allocated only when requested by the VM, and if the host's unallocated resources don't meet the reservation requirements, the VM cannot be powered on. The default reservation is set to 0**.**

To make it simple, if you make a reservation, the VM will always have a reserved amount of resources available, even during contention. When there is contention, VMs without a reservation must free up resources to the VMs with a reservation. For the remaining resources, the VMs will compete between each other based on the shares that they have.

Reservation, on the other hand, does not mean that the VM will lock such resources, and they will not be available for anybody else. If the VM does not use resources that are reserved, other VMs can freely use them.

Limits

Using limits, you can specify the maximum amount of resources a VM can use. If the limit is not set, a VM will consume up to the maximum amount of resources based on its configuration and the virtual hardware used. If a VM is configured with a limit, although it has some resources configured, it will never use more than specified by the limit. The default limit value is set to unlimited.

For example, you have a VM with 16 vCPUs, each running at 2.4 GHz, giving you a total of 38.4 GHz of computing resources, but you might want to configure the VM in a way that it will never consume more than 4.8 GHz of computing resources. From the guest OS perspective, it will have 16 CPUs, each running at 2.4 GHz.

CPU resources

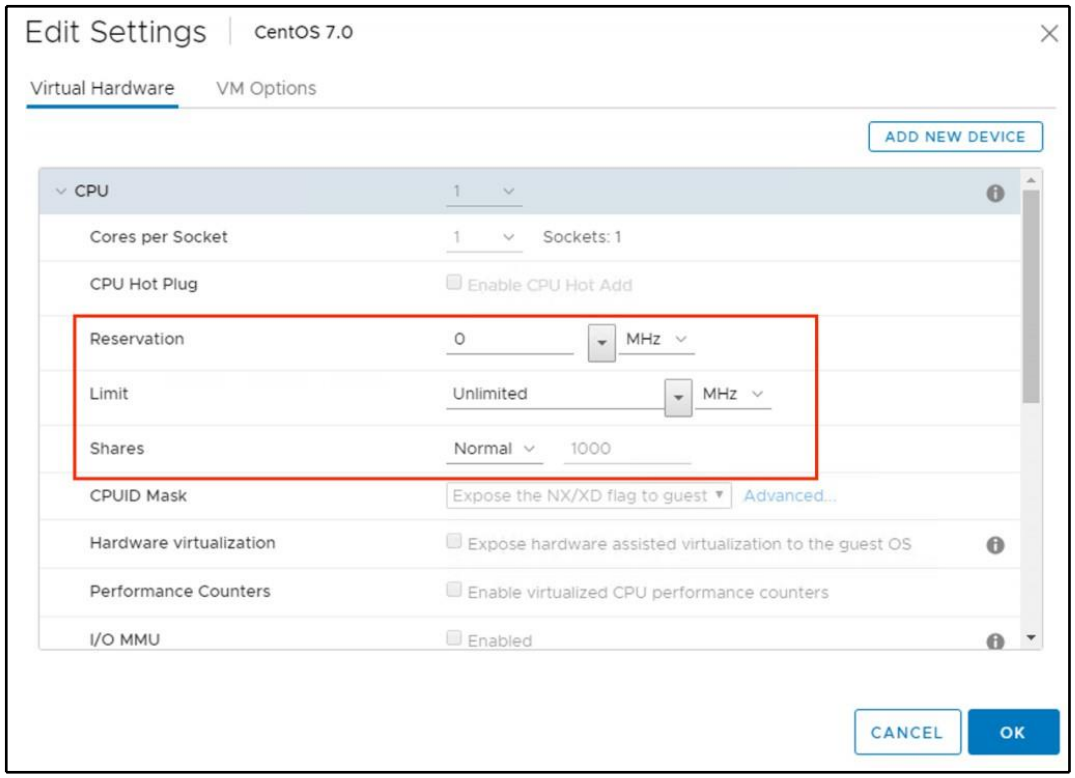
For a vCPU, shares, reservation, and limit parameters can be configured:

- **Shares**: This parameter allows you to prioritize access to resources during resource contention. Shares determine how much CPU power in GHz will be provided to a VM.

- **Reservation**: This is used to specify the minimum CPU power in GHz guaranteed for a VM, and you can't reserve more CPU cycles than ESXi is capable of delivering. The host must have enough physical CPU capacity to satisfy the reservation; otherwise, the VM won't be able to power on.
- **Limit**: This is used to prevent a VM from accessing additional CPU power in GHz, even if they are available. The VM won't use more CPU cycles than specified in the limit.

To configure shares, reservation, and limit CPU parameters, follow these steps:

1. Right-click the VM to configure and select **Edit Settings**
2. Access the **Virtual Hardware** tab and expand the **CPU** item
3. Configure the parameters you need, and then click **OK** to confirm:



To improve resource management, CPU configuration can be enhanced by enabling additional components and parameters:

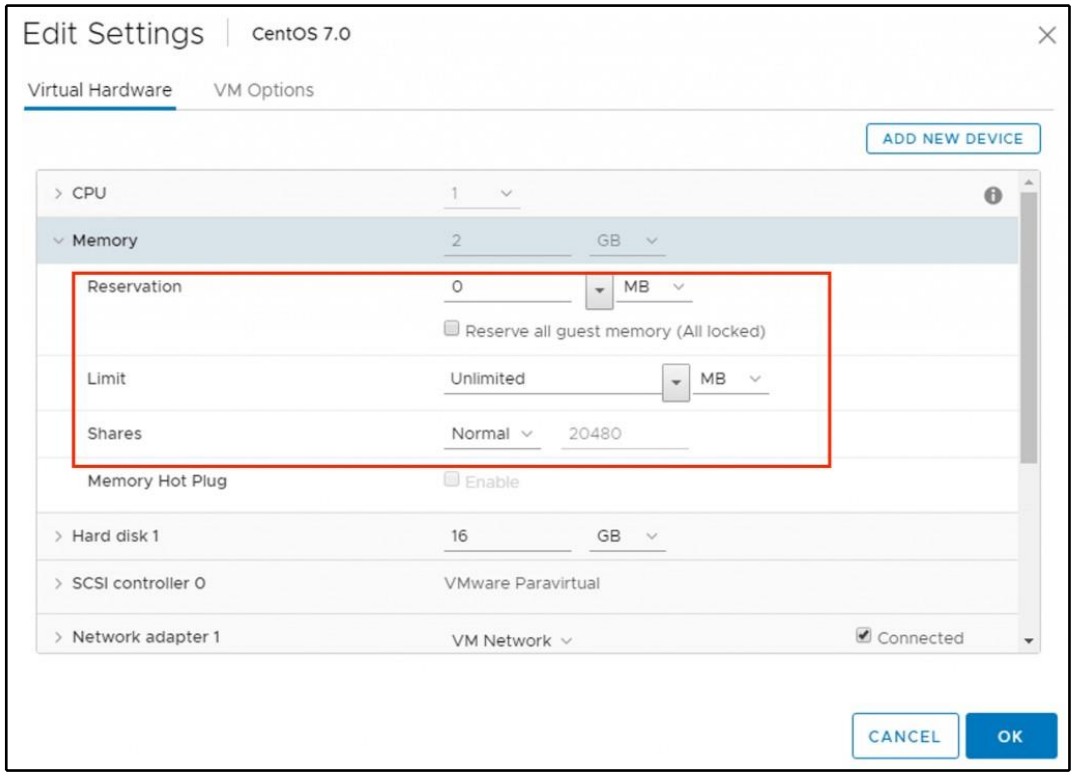
- **Hyperthreading**: This is a technology that allows a single physical processor core to behave like two logical processors. With this technology, a single processor core can execute two independent threads simultaneously, improving performance.
- **CPU affinity**: This is a configuration that assigns a specific VM to a specific physical CPU core and can be used for compliance to license requirements. CPU affinity should be used carefully because it can introduce potential issues, such as interfering with the ESXi host's ability to meet the reservation and shares that have been specified for a VM.

Memory resources

As with the CPU, shares, reservation, and limit are settings that are used to allocate and manage memory resources for a VM.

To configure shares, reservation, and limit parameters, proceed as follows:

1. Right-click the VM to configure and select **Edit Settings**
2. Access the **Virtual Hardware** tab and expand the **Memory** item
3. Set the appropriate values and then click **OK** to confirm:



Although it is possible to configure the limit for memory resources, you should avoid doing that all costs because it will significantly affect the performance of the VM.

If you use a limit, from the perspective of the guest OS, the configured amount of memory will be presented, but only the subset of the memory will be physically available. *What about the rest of the resource?* You are right. It will be swapped at the ESXi level.

Although ESXi memory swapping is one memory reclamation technique, you should avoid it at all costs, because the guest OS is not aware of such swapping. This means it cannot effectively distribute the memory from within the guest OS.

Let's have a look at the following example:

- You have configured a VM with 8 GB of memory
- A limit is configured at 6 GB of memory

What is the result? The guest OS thinks it has 8 GB memory available, thus it is acting accordingly, but in reality, only 6 GB of memory is backed up with the physical memory of the ESXi host, and the swap file is backing the remaining amount of the memory (2 GB).

Since this is transparent to the guest OS, it is not able to manage the memory effectively and decide what parts of the programs should be placed in the swap file and what parts are actively used.

It is always preferred to use swap on the guest OS level because in this case, the guest OS can effectively distribute the memory to the active applications.

VM swapping

By default, the swap file for each VM is created when you power it on, and it will be stored within the folder of the VM in a specific datastore.

The size of the VM's swap file equals the size of the VM's configured memory, unless you use reservations for the memory.

When no reservation is configured, the ESXi host cannot guarantee the amount of memory that will be served from physical memory and the size of memory that might be swapped due to contention.

When you reserve all memory for the VM, no swap file will be created because the ESXi hypervisor will ensure that the VM will always get the physical resources it needs.

If you set a partial reservation for the VM, the size of the swap file equals the configured memory size, minus the reservation.

ESXi host memory states

In times of contention, the ESXi host will use different techniques that allow you to overprovision the memory assignment for the VMs. As we already explained, overcommitment means that you assign more virtual resources than your physical hypervisor has. This is quite a common approach, but you need to be aware of what happens once all the VMs start to utilize such resources. This situation is called **contention**.

We have two different contentions when talking about resources: CPU and memory contention. There is nothing specific regarding CPU contention—the shares will simply kick in and the VMs will get the only subset of configured resources based on the shares. But for memory, it is more complicated.

vSphere 6.7 uses different reclamation techniques that you may already know from previous versions:

- **Transparent page sharing** (**TPS**)
- Memory ballooning
- Memory compression
- Memory swapping

The memory reclamation technique you choose depends on the ESXi host's memory state. This is determined by the amount of memory available at a given time in the ESXi host.

If the amount of free memory is lower than a certain amount, different reclamation techniques will be used to free pages from physical memory.

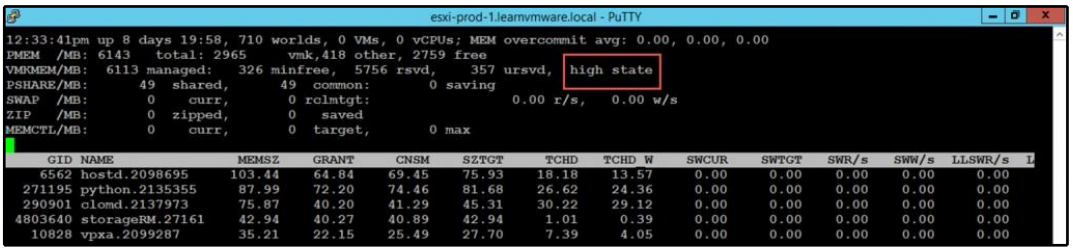
There are five memory states of the ESXi hypervisor:

- **High state**: Enough free memory available
- **Clear state**: <100% of minFree
- **Soft state**: <64% of minFree
- **Hard state**: <32% of minFree
- **Low state**: <16% of minFree

You can quickly discover which memory state your ESXi hypervisor is in by using the `esxtop` command:

1. Connect to your ESXi hypervisor using `SSH`
2. Issue the `esxtop` command

3. Press *m* to switch to the memory view:



minFree is a dynamic value that depends on ESXi's physical memory configuration. For the first 28 GB of physical memory, minFree is set to 899 MB. For every 1 GB above 28 GB, you need to add 1% to memFree**.**

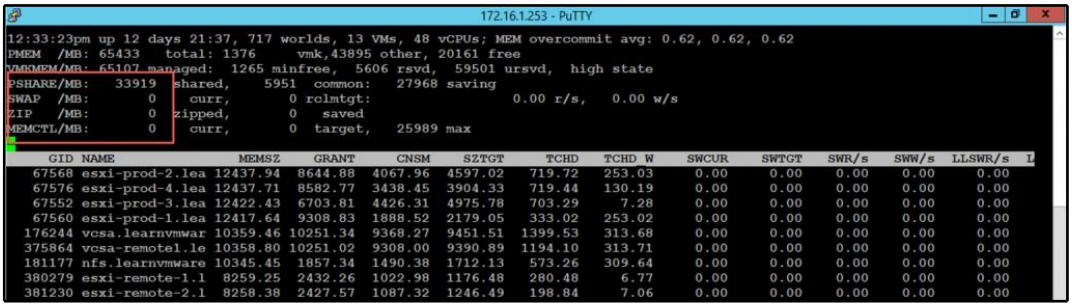
You can determine a minFree memory size based on the following table:

Physical memory	High state/clear state	Soft state	Hard state	Low state
28 GB	899 MB	575 MB	288 MB	143 MB
32 GB	939 MB	601 MB	300 MB	150 MB
48 GB	1,099 MB	703 MB	352 MB	176 MB
64 GB	1,259 MB	805 MB	403 MB	201 MB
128 GB	1,899 MB	1,215 MB	608 MB	304 MB
256 GB	3,179 MB	2,034 MB	1,017 MB	508 MB
384 GB	4,459 MB	2,854 MB	1,427 MB	713 MB
512 GB	5,739 MB	3,672 MB	1,836 MB	918 MB
768 GB	8,299 MB	5,311 MB	2,656 MB	1,328 MB
1,024 GB	10,859 MB	6,950 MB	3,475 MB	1,737 MB

Depending on the memory state, different memory reclamation techniques are invoked:

Memory state	Transparent Page Sharing	Ballooning	Compression	Swapping	Blocking
High	Standard TPS cycles				
Clear	ESXi actively calls TPS to collapse pages				
Soft	X	X			
Hard	X		X	X	
Low	X		X	X	X

Again, using `esxtop`, you can see what memory techniques are currently invoked and the amount of reclaimed memory:



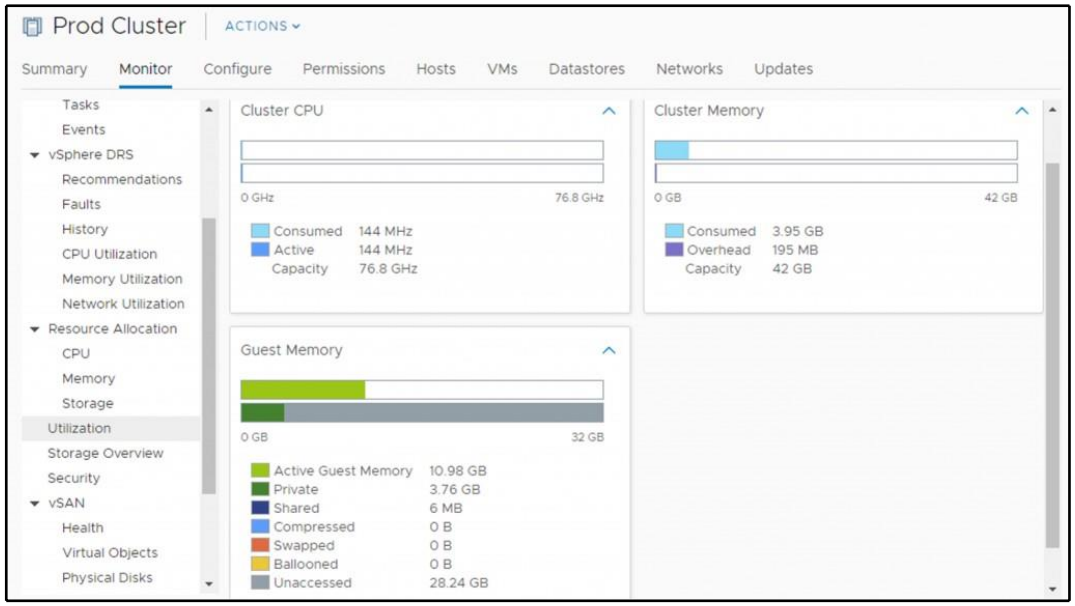
The following metrics are related to the ESXi reclamation techniques:

- `PSHARE`: Memory reclaimed using TPS
- `SWAP` : Memory reclaimed using ESXi host swapping
- `ZIP`: Memory reclaimed using compression
- `MEMCTL`: Memory reclaimed using ballooning

You can quickly identify your overall memory reclamation techniques that are in use from the vSphere client for each cluster:

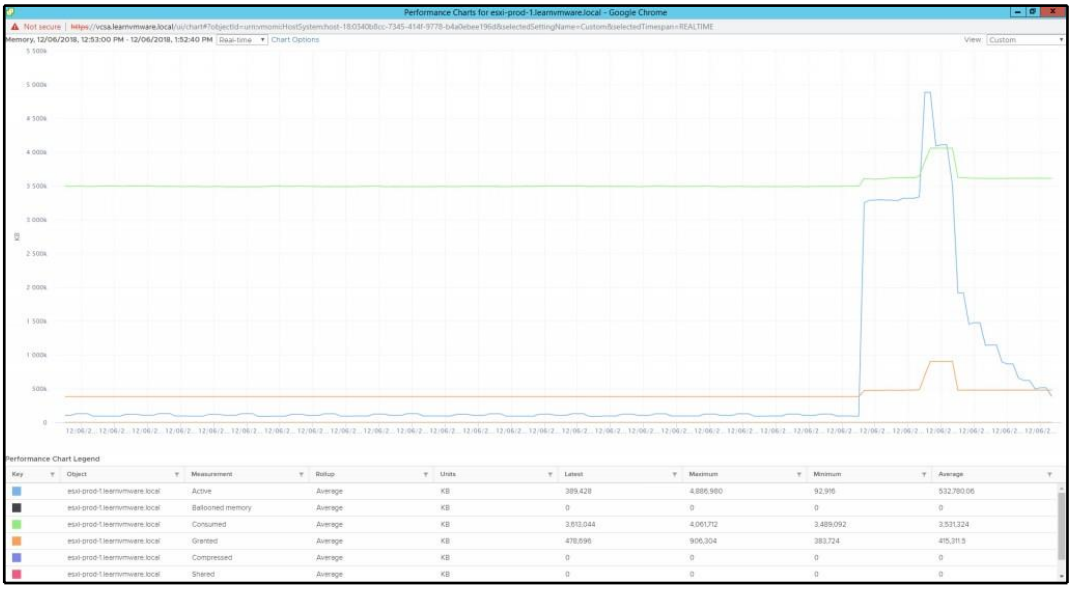
1. Select the cluster you are interested in
2. Switch to the **Monitor** tab
3. Navigate to **Utilization**

Guest Memory is the graph you are interested in:



You can identify different memory reclamation techniques that are currently invoked by the ESXi hypervisor following this procedure:

1. Select your ESXi hypervisor
2. Switch to the **Monitor** tab
3. Click **Advanced** under **Performance**
4. Select the **Memory** view
5. Click **Chart Options**
6. Select the **Ballooned memory**, **Compressed**, **Shared**, and **Swap consumed** metrics:



TPS

TPS has been around for a long time, and its purpose is to save memory at the host level. It is similar to storage deduplication, but this time focusing on the memory.

When multiple instances of VMs are run on the same ESXi hypervisor and access the same memory pages, they are stored only once. With TPS, the hypervisor will eliminate the redundant memory pages by mapping the identical content in only one memory page in the physical memory.

The TPS mechanism runs in the background and calculates a hash of the memory page. Those hashes are stored in a hash table and they are compared to each other by the ESXi server. If the ESXi kernel discovers two corresponding hashes, it will compare the content of the memory page. If the content is exactly the same, then only one memory page will be stored in the physical memory and the other one will be pointed to the same location.

Two types of memory sharing techniques are available:

- **Intra-VM:** Memory pages within the same VM will be deduplicated by TPS, but TPS will not share the memory pages between different VMs.
- **Inter-VM**: Memory pages within the same VM will be deduplicated by TPS and TPS will share the memory pages between different VMs.

There was a major change with vSphere 6.0 and Inter-VM TPS is now disabled by default.

There is no real-world example of exploiting Inter-VM memory sharing to inject malicious code as far we know, but as a security hardening best-practice, the behavior was rather changed.

If you are running a Service Provider environment, you should probably keep the settings at the default to prevent any malicious misuse of the feature. However, if you need, you can change the default behavior.

There are three possible values of `Mem.ShareForceSalting`:

- `2`: Default value. No Intra-VM TPS
- `1`: Intra-VM TPS will be used for VMs with the same `sched.mem.pshare.salt` advanced configuration option.
- `0`: Inter-VM TPS works as expected.

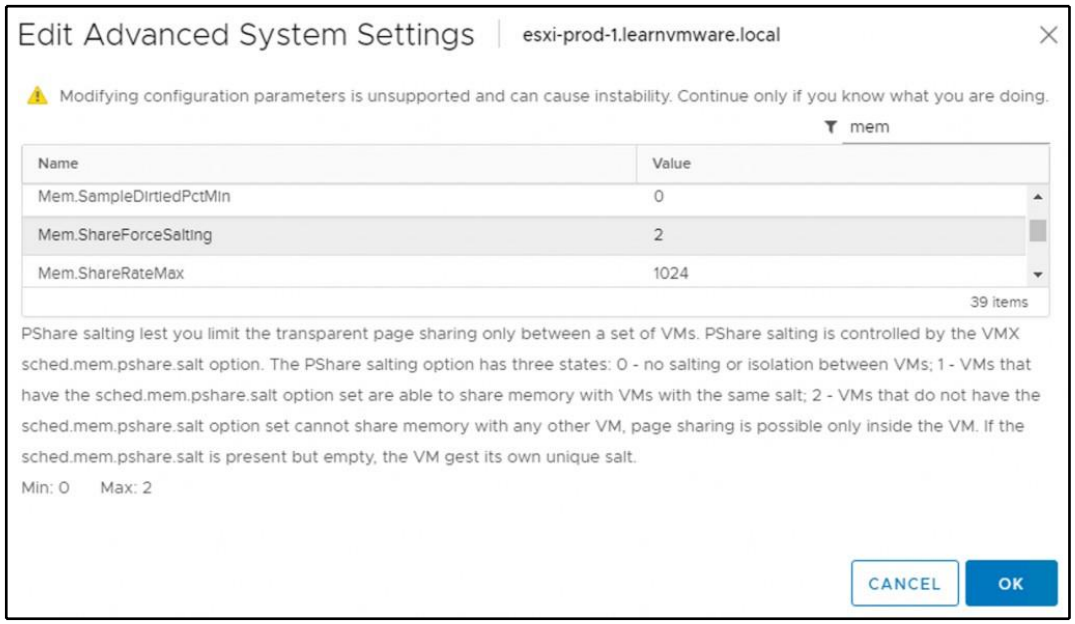
For more informations about Intra-VM TPS, feel free to visit the following KB: `https://kb.vmware.com/s/article/2097593`.

For Enterprise companies, I would suggest to switch to the old behavior and enabling Inter-VM TPS since the benefits of the TPS will—from my perspective—outweigh the possible security concerns. For service providers—from my perspective—I would use the same salt for all VMs belonging to the same customer, so the result will be that VMs from a single tenant can share memory pages between each other, but they can't share memory pages between different tenants.

Please note that `Mem.ShareForceSalting` is a per-host setting and `sched.mem.pshare.salt` is a per-VM setting.

You can change the `Mem.ShareForceSalting` settings from the vSphere client by following these steps:

1. Select your ESXi hypervisor
2. Switch to the **Configure** tab
3. Locate **Advanced System Settings** under **System**
4. Click **Edit**
5. Locate the `Mem.ShareForceSalting` configuration parameter and change it to the desired value, as shown in the following screenshot:



Ballooning

The hypervisor uses a memory reclamation technique called **memory ballooning** to reclaim the memory from a VM.

The ESXi server is not aware of the content of the memory page of the VM. Only the guest operating system knows what is inside and which memory pages are more important than the others.

That is why the balloon driver is an essential memory reclamation technique. Memory ballooning is not happening at the ESXi level but inside of each VM. As a part of VM tools, a specific balloon driver is installed into the guest operating system—`vmmemctl.sys`. As you already know, any driver running inside of the guest operating system, no matter whether the VM is Windows- or Linux-based, is run on the kernel level, thus having more significant priority over the user-space where the applications are being run.

As the balloon driver is invoked from within the guest operating system, the underlying operating system knows what is running inside and can decide what will be swapped to the virtual disk where the operating system is installed.

Let's have a look at the following example.

When there is no contention (the host is not in soft memory state), the balloon driver is deflated, consuming almost no memory resources, but when the contention occurs, the balloon driver starts to inflate. As we mentioned earlier, because it is from within the guest operating system, the OS will determine by itself what memory pages are not used (but still active) or are not accessed frequently. Those memory pages (in the virtual memory of the guest OS) will be swapped to the virtual disk. Once the contention is over, the memory balloon will deflate and the guest operating system will swap the memory pages from the system back to the virtual memory.

By default, the balloon driver (`vmmemctl.sys`) can reclaim up to a maximum of 65% of the physical guest memory. For example, your VM is allocated with 4,096 MB of memory. It can reclaim up to 2,662 MB using this technique.

Compression

ESXi provides a memory compression cache to improve VM performance whenever you use memory overcommitment. Memory compression is enabled by default. ESXi compresses virtual pages and stores them in memory when a host's memory becomes overcommitted.

For more information, take a look at the following PDF: `https://pubs.vmware.com/vsphere-51/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-51-resource-management-guide.pdf`.

By default, up to 10% of VM memory can be compressed—the advanced system parameter `Mem.MemZipMaxPct` determines this value. The value is a percentage of the size of the VM and must be between 5 and 100 percent.

Compression is enabled by default, but if you want, you can disable the memory compression mechanism by using the `Mem.MemZipEnable` advanced system parameter.

Host swapping

This is the last resort technique that you do not want to be in. Since host swapping use the underlying storage infrastructure to swap page files from the memory to the disk and it is not aware of the content of the memory pages, it will significantly impact the performance of the VMs.

System swap is determinated automatically by the kernel of the ESXi server. You can alter the behavior by changing the **preferred swap location**. If no feasible option is available, then the system swap is not activated at all.

On the ESXi level, you can define the default location of the ESXi swap location:

1. Select your ESXi server
2. Switch to **Configure** tab
3. Select **System swap** under the **System**
4. Click **Edit** settings

Virtual machine migration

In vSphere 6.7, VMs can be moved from one host or storage to another using hot or cold migration. Wherever possible, hot migration is the preferred option to use to avoid service disruption because it performs a live migration of the VMs:

- **Hot migration**: A powered-on VM can be moved to a different host or datastore without service disruption using the vMotion or Storage vMotion features.
- **Cold migration**: This is the migration of a powered-off or suspended VM. You can move associated disks from one datastore to another and VMs are not required to be on shared storage. A cold migration can be performed manually or by scheduling a task.

There are two types of vMotion:

- Compute vMotion (otherwise known as standard vMotion) is responsible for migrating active state (the content of VM memory) between two ESXi hosts
- Storage vMotion (SvMotion) is responsible for the migration of the storage resources between two different datastores

Compute vMotion

vMotion has been around for more than fifteen years. The first version was introduced in 2003, which means it is a mature technology.

vMotion utilizes a similar technology (similar based on its behavior, but completely different from a technical perspective) as a snapshot. When you invoke vMotion, a snapshot of the memory is created, allowing new write operations to memory to be stored in a dedicated, known, section. Then, the content of the memory is migrated, and in the end, the data that has changed during the vMotion invocation are synced.

You might be wondering why you should use vMotion, so there is a list of several tasks that involves the use of vMotion:

- **ESXi maintenance mode**: If you have a DRS-enabled cluster, DRS will automatically migrate VMs from the ESXi host that is going into maintenance mode.
- **Troubleshooting**: You are experiencing one of the VMs behaving strangely. You can try to migrate it to a different ESXi hypervisor to determine if the problem is widespread or is only occurring on a single ESXi server.
- **Cluster balancing**: DRS might invoke vMotion to move VMs around the cluster for better resource balance.
- **Host standby:** DPM might invoke vMotion to move VMs from the ESXi host that is going into standby mode.
- **Affinity rules:** Some VMs should not be run together or should be run on the same ESXi hypervisor. If not, vMotion will be used to correct the situation.

Although it might sound like an easy task, many things are going on under the hood:

1. As a first step, vCenter server will validate whether the source VM can be run on the destination server.
2. Then, a new VM process is started on the second ESXi server and resources for the VM are reserved.
3. A memory checkpoint is initialized on the source VM so that all changes in the memory are written to the dedicated memory section.
4. The content of the memory is transferred over the network to the destination ESXi hypervisor.
5. Changes in memory during the transfer are again checkpointed and synced with the destination ESXi hypervisor. The checkpoint/checkpoint-restore operation might repeat several times.
6. The source VM is stopped and the remaining memory fragments are synced to the destination.
7. Once the vMotion process is finished, a reverse ARP packet is sent to the physical switches.

The **Notify Switches** option must be enabled on the virtual switch). Hard disk access is switched to the destination ESXi server.

8. The VM process running on the the source ESX hypervisor is terminated and deleted.

As stated, there will be multiple iterations of the memory checkpoints. The reason behind this is that if the VM is configured with a lot of memory and the memory is actively used, the delta between the creation of the checkpoint and the resulting changes will be quite large, meaning that the VM will be suspended for a quite long period. Therefore, multiple checkpoints are created when using vMotion.

Let's have a look at the following example:

Iteration number	**Memory to transfer**	**Time for the transfer**	**Changes during the memory transfer**
1	16,384 MB	30 seconds	3,072 MB
2	3,072 MB	8 seconds	512 MB
3	512 MB	2 seconds	64 MB
4	64 MB	0.25 seconds	4 MB

Iteration number	Memory to transfer	Time for the transfer	Changes during the memory transfer
5	4 MB	VM freeze for takeover	No new delta checkpoint created

vCenter server is responsible for validation, and it invokes the vMotion process on the ESXi hypervisors, but it is not involved in the actual data transfer. Therefore, an active vMotion process must always be allowed to run to completion, even if the vCenter server crashes.

As with any technology, specific prerequisites need to be met:

- The ESXi hypervisor must be licensed for vMotion
- The ESXi hypervisor must be configured with a VMkernel adapter with the vMotion service enabled on the adapter
- The ESXi hypervisor must have the same physical CPU or EVC must be configured
- The ESXi hypervisor must have shared storage accessible by the source and destination ESXi hypervisor

Moreover, there are of course certain limitations as well, which can be found at `http://www.vmwarearena.com/vmware-interview-questions-vmotion/`.

The migration of a VM is performed by following this procedure:

1. From vSphere Client, log in to vCenter Server, right-click the VM to move, and click the **Migrate** option.
2. In the migrate window, select the migration type to perform, choosing from the following options:
    - **Change computer resource only**: The VM is moved to a different compute resources, such as host, cluster, resource pool, or vApp. A powered-on VM is moved using vMotion.
    - **Change storage only**: The VM disks are moved to a different datastore on the same host. The storage vMotion feature is used to move a powered-on VM to a new datastore.
    - **Change both compute resource and storage**: Virtual disks are moved to a new datastore and computer resources are moved to another host. Cold or hot migration can be used to change the host and datastore. If the network of the VM is moved between distributed switches, network configuration and policies are transferred to the target switch.

3. Based on the migration type selected, you must then specify the compute resource, storage location, and vMotion priority (high or normal) to finalize the migration.

4. You also have the option to change the vNIC assignment during the vMotion invocation (such as changing the Port Group to which the VM belongs); this is handy if you do not use a distributed vSwitch, and the naming convention is not the same on the source and destination ESXi hypervisor.

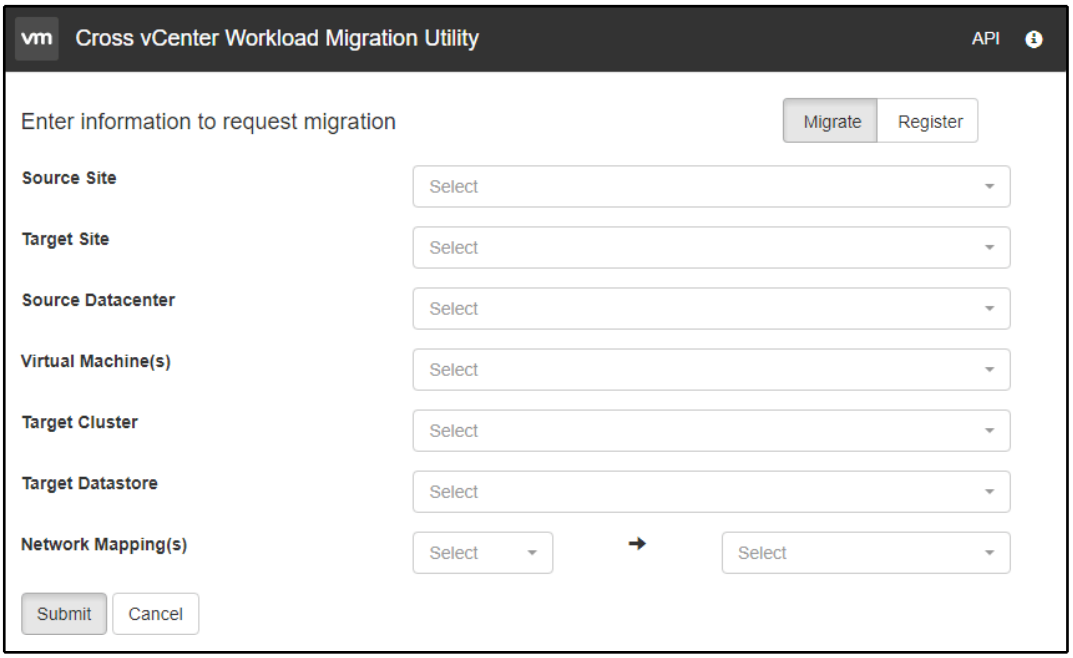
Starting from vSphere 6.0, vMotion has been enhanced, introducing new functionalities such as Cross vSwitch vMotion, **Cross vCenter vMotion** (**xVC-vMotion**), and **Long Distance vMotion** (**LD-vMotion**):

- **Migrate to another virtual switch**: A VM can be migrated to a different type of virtual switch (standard or distributed) without reconfiguring the physical and virtual network. You can move the VM from a standard to a standard or distributed switch and from a distributed to another distributed switch.
- **Migrate to another datacenter:** During the migration, you can specify the target data center to move the VM between data centers. In the target data center, you can specify a dedicated port group on a distributed switch for network settings.
- **Migrate to another vCenter Server system:** VMs can be moved between vCenter Servers if they are connected in Enhanced Linked Mode, and also between vCenter servers that are located a long distance from each other.

Please note that it is even possible to migrate to a different vCenter server that is not connected using Linked Mode. Although this functionality is presented in the APIs, it is not possible to use the UI to invoke such a migration.

If you are interested in Shared-Nothing Cross vCenter Server migrations, you can use the Cross vCenter Workload Migration Utility, which is available at `https://labs.vmware.com/flings/cross-vcenter-workload-migration-utility`.

You can see **Cross vCenter Workload Migration Utility** GUI in the following screenshot:



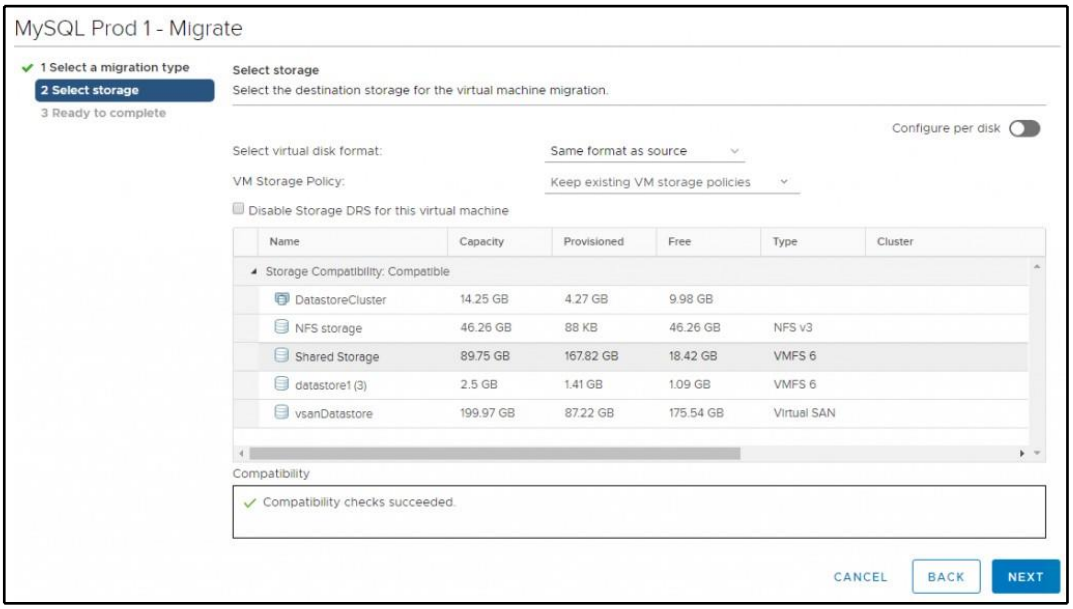
Storage vMotion

Storage vMotion works the same way as compute vMotion. The only difference is that the virtual disks and VM configuration files are moved between different datastores, not the memory between ESXi hypervisors.

Under the hood, again, a type of snapshot technology is used. When you invoke the storage vMotion, a new snapshot file is created on the datastore, and all write operations are performed on the snapshot. A base disk is switched to the read-only state so that the **Storage vMotion** (**SvM**) process can access it and the copy between datastores starts. Once the main VMDK file is transferred, the snapshot on the first snapshot is created, and the first snapshot is copied and merged to the base disk on the destination datastore. This process will be performed several times until only a small amount of data will be on the source datastore. At this time, again, the same as with compute vMotion, the VM is frozen, and the remaining bytes are copied and synced, and the VM is resumed from the new datastore.

The process of how to invoke storage vMotion is the same as with compute vMotion:

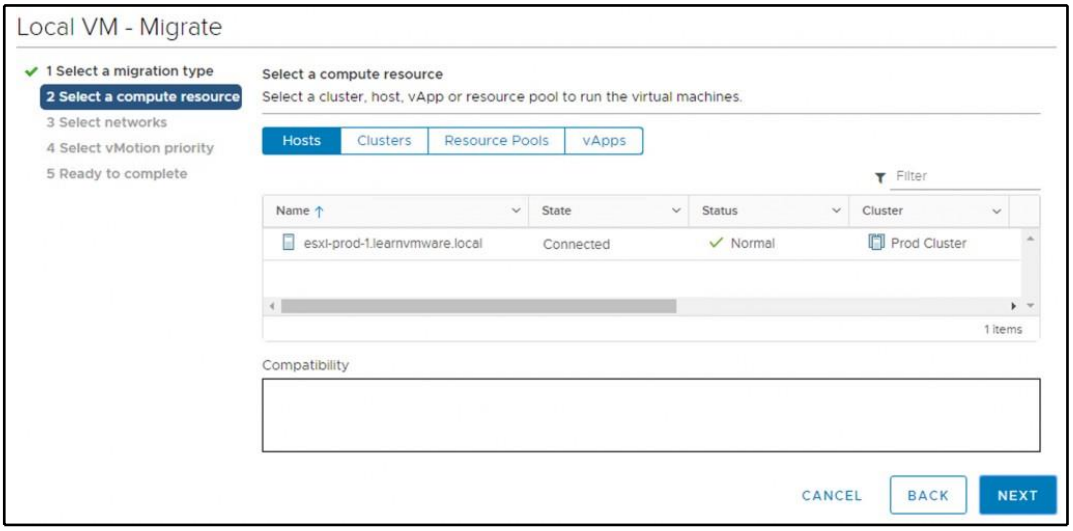
1. Select a VM and open the **Migrate** wizard.
2. Select the **Change storage only** option.
3. Select the destination datastore to which you would like to move the VM. You can also change the disk format from thin to thick or attach a different storage policy:



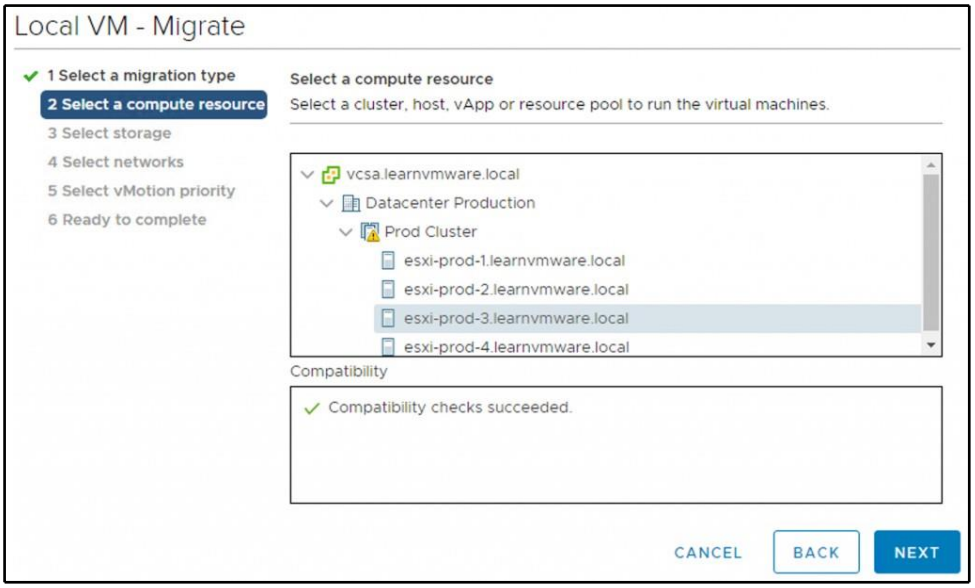
vMotion without shared storage

To perform vMotion, environments with shared storage are not required. For example, you can migrate a running VM between ESXi hosts with only local storage. When migrating a VM cross-cluster, the target cluster VM might not have access to the source cluster's storage.

Let's have a look at the VM that runs on a local datastore on the **esxi-prod-1** hypervisor. If we choose to migrate the compute resource only, the wizard will not allow us to migrate the VM to any other ESXi hypervisor, since the VM is backed up with local storage:



However, if we select **Change both compute resource and storage**, we will be able to migrate the VM to a different ESXi hypervisor and shared storage:



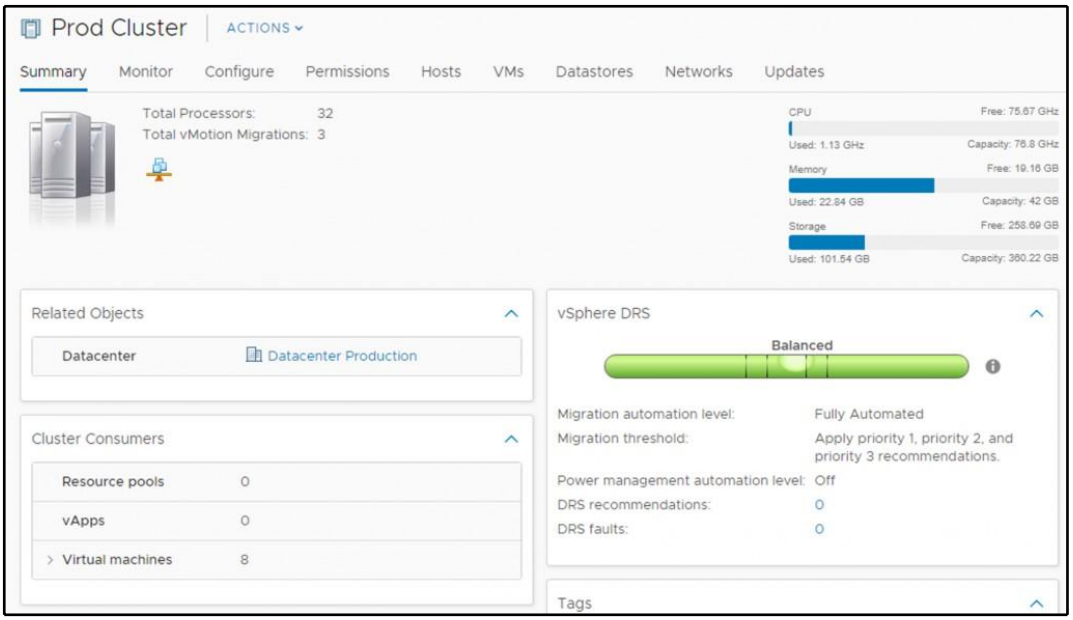
DRS

As discussed in *Chapter 11*, *Configuring and Managing vSphere 6.7*, a vSphere cluster is a collection of ESXi hosts that share resources and a management interface. Some of the vSphere's features are available only on the cluster level and DRS is one of them. Once the DRS is enabled on the cluster, the capability to automatically balance loads across the ESXi hosts will be available. vSphere DRS provides two main functions:

- Executing the placement of the just-powered-on VM on a specific host in the cluster
- Periodically (every 5 minutes by default), DRS checks the load on the cluster, providing recommendations for migration or automatically migrate the VM (using vMotion) to get a balanced cluster

If you have a DRS-enabled cluster and one of the hosts is heavily loaded compared to other host members, you might notice DRS doesn't vMotion any running VM off the host, leaving the workload unchanged. Until the ESXi host can satisfy resource demand from the VM, DRS doesn't perform any action. DRS ensures that the cluster is balanced, regardless of the workloads distributed on individual host members. To get balanced clusters and host members, there are third-party applications that provide real-time automation to allocate resources efficiently.

You can find a basic overview of your cluster's balance in the **Summary** tab of your vSphere cluster, as shown in the following screenshot:



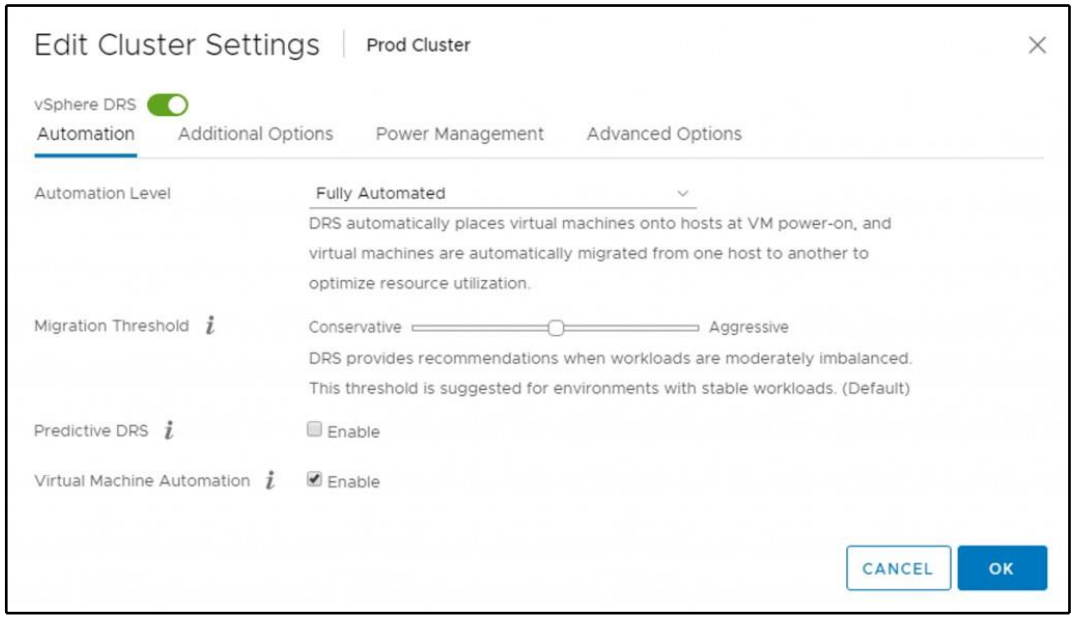
When a VM in a DRS-enabled cluster is powered on, the vCenter Server checks whether the cluster has enough resources to support the VM, that is, it performs admission control. If the available resources in the cluster are not sufficient to power on the VM, a warning message appears. If the resources are sufficient to support the VM, a recommendation on which host the VM should run is generated by the DRS and, based on the automation level configured in the cluster, one of the following actions is taken:

- The placement recommendation is executed automatically

- The placement recommendation is displayed, leaving the user with the option to accept or override

When DRS is disabled, no recommendations are provided, and VMs are not moved among the cluster's hosts.
To enable DRS in a cluster, proceed as follows:

1. From the vSphere client, log into vCenter Server and right-click the cluster in which you want to enable DRS and select **Settings**.
2. Under **Services**, select **vSphere DRS** and click the **Edit** button.
3. Enable the vSphere DRS option, and from the Automation drop-down menu, select the level of automation you want to apply to the cluster:
    - **Manual**: Placement and migration recommendations are displayed, but must be applied manually
    - **Partially Automated**: The initial placement is performed automatically, but migration recommendations are only displayed without running
    - **Fully Automated**: Placement and migration recommendations run automatically:

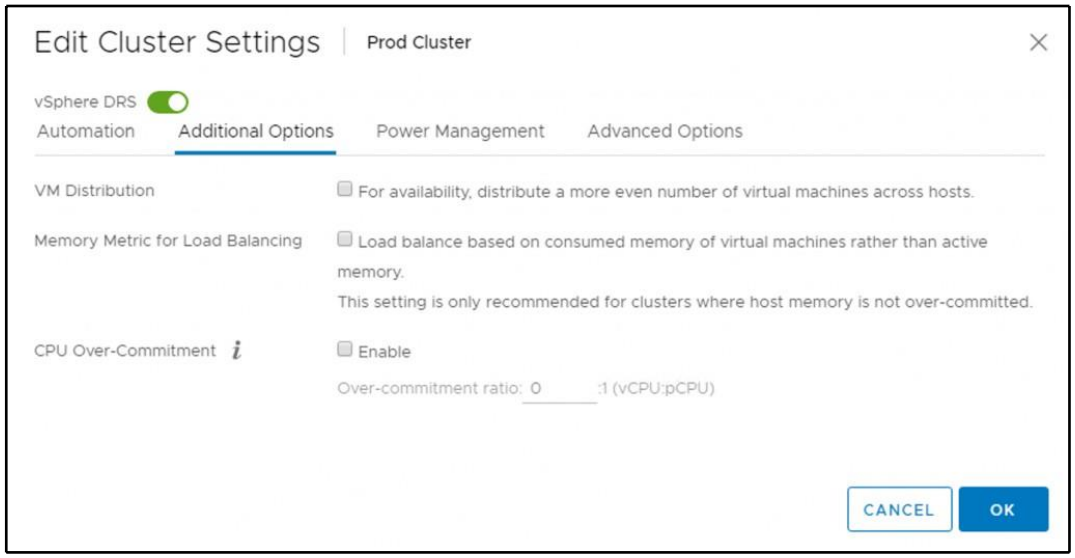


4. Select your **Migration Threshold:**
    - **Priority 1**: Those recommendations are not based on cluster imbalance but rather on system requirements such as anti-affinity and affinity rules, maintenance mode, or DPM.
    - **Priority 2 to 5**: Based on the cost/benefit of the VM migration, DRS rules are generated for priority 2 to 5. This configuration option basically adjusts the sensitivity of the DRS.

    For more information about DRS migration thresholds, feel free to visit the following website: `https://blogs.vmware.com/vsphere/2016/10/drs-migration-thresholds.html`.

5. If you have vRealize Operations in place, you can also enable **Predictive DRS**. With Predictive DRS, migration can occur even before your cluster is unbalanced based on historical data.
6. **Virtual Machine Automation** is enabled by default, and it gives you the option to override the automation level on a per-VM basis.

If you need to, you can also check the advanced parameters of DRS:



- **VM Distribution**: This setting overrules this logic and incurs the cost of migration to achieve a more even distribution of VMs. Please note that this setting will still keep VM happiness in mind, so even the distribution of VMs is done on a best-effort basis.
- **Memory Metric for Load Balancing:** This setting can be helpful for environments that attempt to minimize the impact of host failures or attempt to balance the load on network IP connections across the ESXi hosts in the cluster. Please note that this setting can increase the number of VM migrations without specifically benefitting the application's performance.
- **CPU Over-Commitment:** By default, DRS uses a default CPU over-commit (vCPU to pCPU) ratio that is approximately 80 to 1. A latency-sensitive workload can benefit from a lower CPU over-commit ratio by reducing the number of vCPUs waiting to be scheduled. This setting limits the number of vCPUs that can be powered on in the vSphere cluster.

If DRS is disabled, resource pools configured in the cluster are removed.

Virtual network-aware DRS

Virtual network-aware DRS is a new feature that was introduced in vSphere 6.5, where DRS now also considers the network utilization when it generates the migration recommendations. If a host has **Transmit** (**Tx**) and **Receive** (**Rx**) rates of utilization of the connected physical uplinks that's greater than 80%, the VM won't be placed on that host. Network utilization is an additional check to evaluate whether a specific host is suitable for the VM.

Managing DRS rules

VM placement can be controlled using affinity rules. Affinity rules are useful for administrators to control how specific VMs should be placed in the host members of the cluster for performance and security reasons.

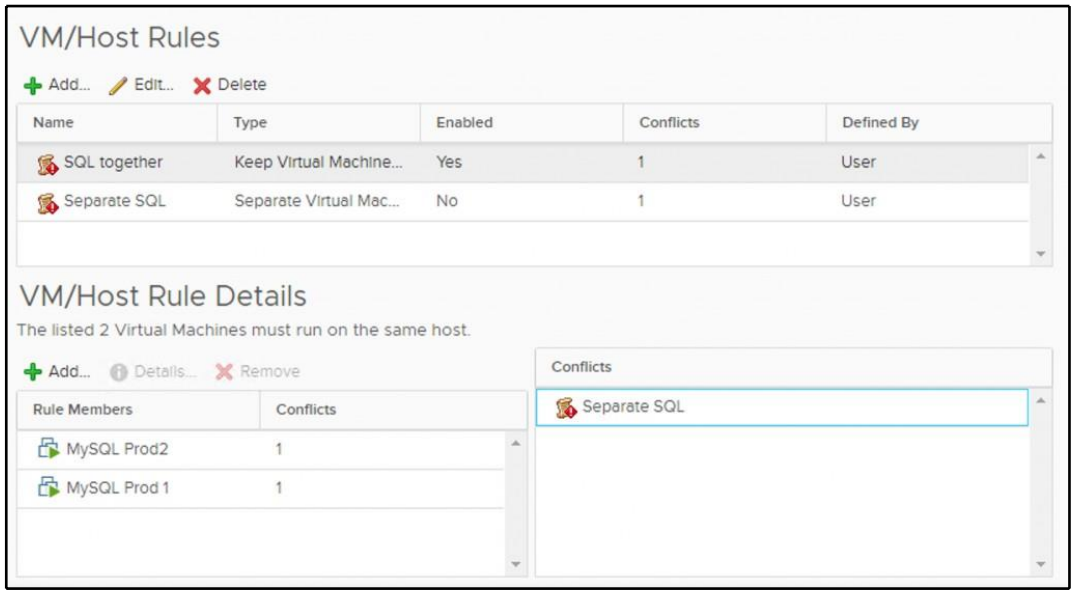
Let's have a look at the DRS-supported affinity rules.

VM-VM affinity rule

The VM-VM affinity rule is used to specify that selected VMs should run on the same host. You can configure this rule to improve performance. The anti-affinity rule behaves in precisely the opposite way, and it's used to ensure that some VMs are kept on different hosts.

Anti-affinity can be applied to AD domain controllers, for example, to keep them on different hosts to avoid AD issues in case one host fails. Only the DC running on the failed host is not available, while the others won't be affected, continuing to provide the authentication service with no interruption.

You can't enable two affinity rules if they clash. For example, if one rule is configured to keep VMs together and another rule keeps the same VMs separated, you can't enable both. In the event of a conflict between two affinity rules, the first rule takes precedence and the newer rule is disabled, as you can see in the following screenshot:



To create a VM-VM affinity rule, perform these steps:

1. From vSphere Client, right-click the cluster that you want to configure and select **Settings**.
2. Under **Configuration**, select **VM/Host Rules** and click the **Add** button.
3. Enter a name and select from the **Type** drop-down menu one of the available options:
    - **Keep Virtual Machines Together**: The specified VMs are kept together on the same host
    - **Separate Virtual Machines**: The specified VMs are separated on different hosts
4. Click the **Add** button to specify the VMs that must run with the specified rule. Click **OK** to save the configuration.

VM-Host affinity rule

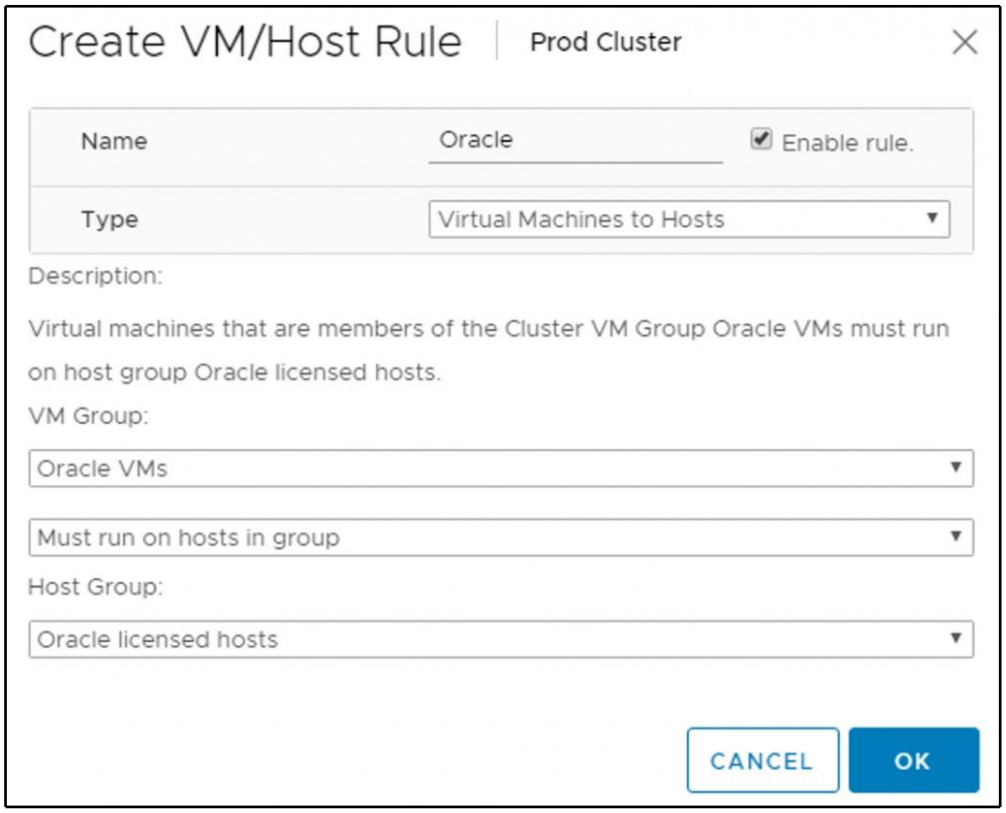
The VM-Host affinity rule allows you to control which hosts in the cluster can run which VMs and requires that at least one VM DRS group and at least one host DRS group are created before managing host affinity rules.

The typical use case is about licensing, where only a subset of ESXi hosts are licensed for a particular software that the VM consumes; Oracle, for example. Without DRS, you would need either a dedicated cluster for Oracle VMs (because you need to assign an Oracle license to each physical server where the VM might run), or, by using DRS affinity rules, you can specify the subset of hosts that will be used by such VMs.

To create a VM-Host affinity rule, proceed as follows:

1. From vSphere Client, right-click the cluster to configure and select the **Settings** option.
2. Under **Configuration**, select **VM/Host group** and click on the **Add** button to create a VM group and a host group.
3. Specify a name and select from the **Type** drop-down menu the VM group. Click the **Add** button to add members to this group, and then click **OK** to save the configuration.
4. Repeat steps 2 and 3 to create a host group.
5. Now, under **Configuration**, select **VM/Host Rules** and click the **Add** button.

6. Enter a name and select from the **Type** drop-down menu the **Virtual Machines to Hosts**. Specify **VM Group** and the rule (for example, **Must run on hosts in a group**), select the **Host Group** to associate, and click **OK** to save the rule:



The options available for the rule can be one of the following:

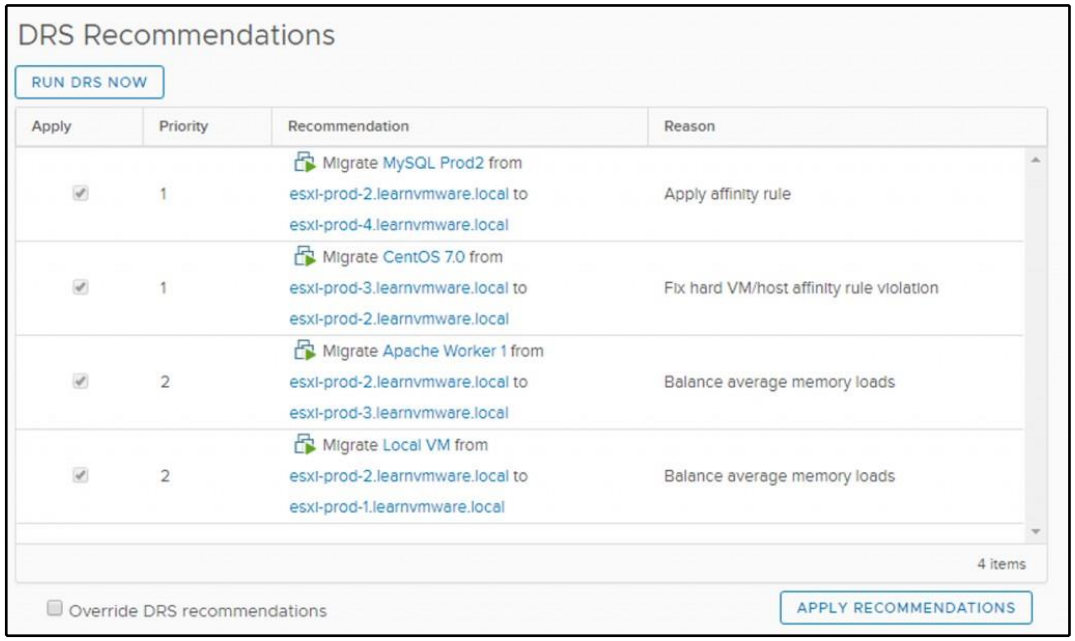
- **Must run on hosts in group**: VMs in the selected VM group must run on host members of the specified host group. DRS will never break the rule, nor will vSphere HA. If there is only one ESXi host in the group, HA will not restart the server on the other nodes.
- **Should run on hosts in group**: VMs in the VM group should run on hosts of the specified host group, but it is not required. DRS will try its best to satisfy the rule, but in some cases, the rule might be broken.
- **Must not run on hosts in group**: VMs in the VM group must never run on host members of the specified host group. DRS will never break the rule.

- **Should not run on hosts in group**: VMs in the VM Group should not, but might, run on hosts of the specified host group. DRS will try its best to satisfy the rule, but in some cases, the rule might be broken.

DRS recommendations

If your DRS cluster is set to partially automated or manual mode, no migrations will be performed automatically by the DRS algorithm. You have to approve the recommendations manually.

You can check what recommendations are available in the Monitor tab of the DRS enabled cluster under vSphere DRS and Recommendations:



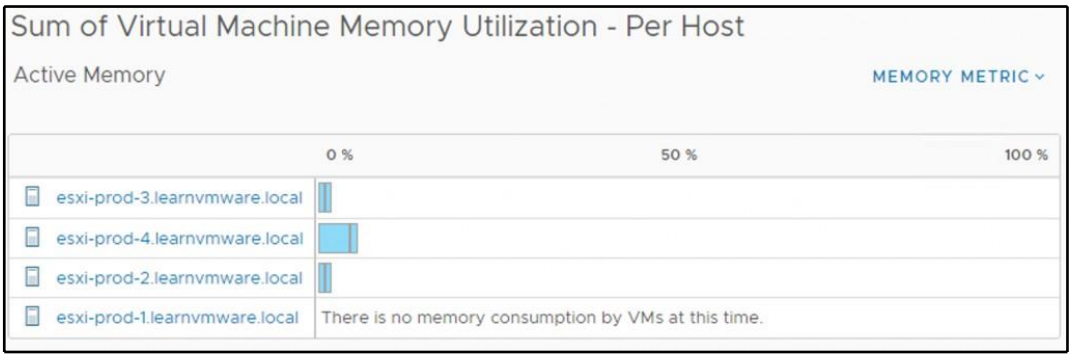
You can either apply all recommendations using the **APPLY RECOMMENDATIONS** button, or you can override them by selecting **Override DRS recommendations** and then select what recommendations you would like to apply.

Once you apply the DRS recommendation, vMotion will be issued to move the VMs between ESXi hypervisors.

In DRS history section, you can find what migrations were performed by DRS and when were they performed.

DRS utilization

DRS decisions are based on the overall utilization of your DRS-enabled cluster, and you can find the current utilization of three key components that are part of DRS decisions under the **Monitor** tab of the cluster, access the vSphere DRS subsection and select either CPU, memory, or network utilization:



For memory utilization, you can switch between consumed and active memory consumption.

Managing power resources

Based on cluster resource utilization, a DRS-enabled cluster can reduce its power consumption by powering on or off ESXi hosts through the vSphere **Distributed Power Management** (**DPM**) feature.

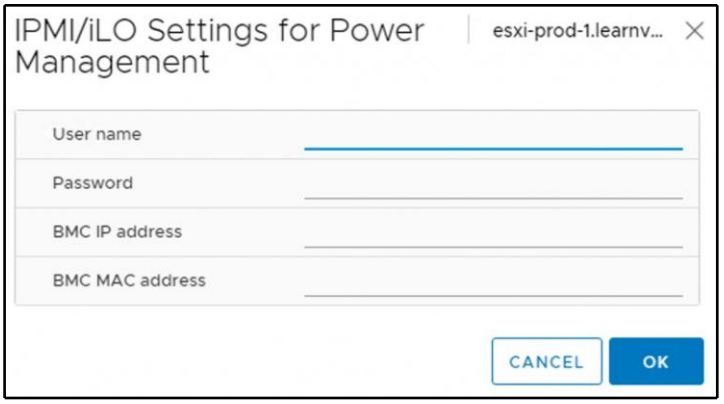
Memory and CPU resources demanded by VMs in the cluster are compared with the total resource capacity that's available from the hosts in the cluster. If the cluster is providing excessive resources, one or more hosts are placed in standby mode by DPM and powered off after migrating the VM to other hosts. When the capacity that's provided is deemed not sufficient, DRS powers the host on, bringing them out of standby mode and vMotions the VMs to them.

vSphere DPM can use three protocols to bring a host out of standby mode:

- **Intelligent Platform Management Interface** (**IPMI**)
- Hewlett Packard **Integrated Lights-Out** (**iLO**)
- **Wake-On-LAN** (**WOL**)

vSphere DPM can put a host in standby mode only if at least one protocol is supported. If a host supports multiple protocols, the following order is used: IPMI, iLO, WOL.
Before you can start using DPM, you must configure the IPMI/iLO/WOL configuration for each ESXi host. To do that, perform the following operation for each ESXi hypervisor:

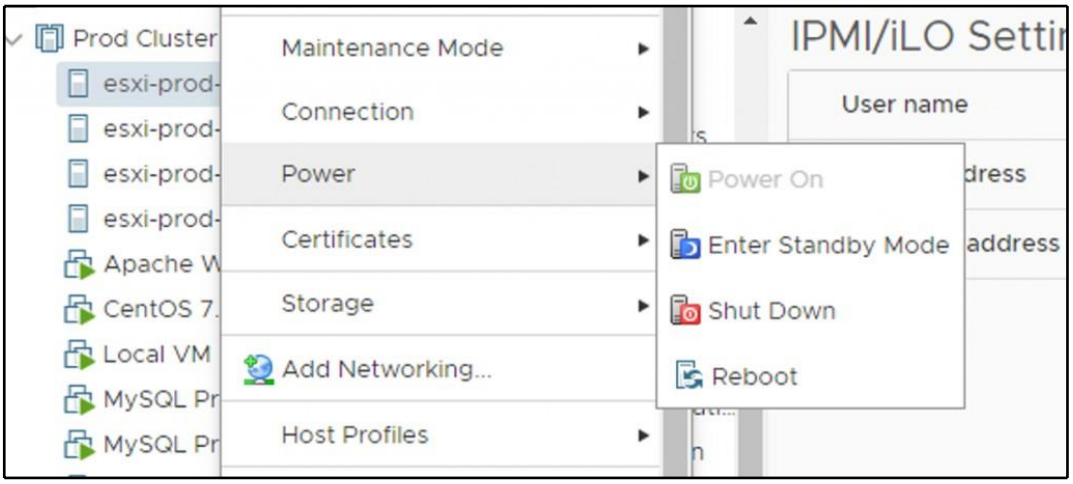
1. Switch to the **Configure** tab of the ESXi hypervisor
2. Navigate to **System** and select **Power management**
3. Click **Edit** to configure the IPMI/iLO settings for power management:



User name and **Password** are the credentials for the out-of-band management module. **BMC IP address** is the IP address of the module. You must also configure the MAC address of the BMC because it will be used for the Wake-On-Lan feature if IPMI / iLO communication fails.

After you configure the IPMI/iLO settings, do not forget to test the communication with the BMC module; otherwise, DPM might shut down the server, but powering on won't work. To do that, right-click on the ESXi hypervisor, and select **Power** and **Enter Standby Mode** to shut down the server. Once the server is offline, you can select the **Power On** operation to bring the server up.

You can invoke power management features of the ESXi server from the vSphere client as shown in the following screenshot:



Resource pools and vApps

Resource pools are logical containers that can be used to allocate compute resources to a group of VMs (or child resource pools). The configuration options are exactly the same as with single VM—you can assign different reservations, shares, or limits for both CPU and memory resources on the resource pool, but compared to individual assignment to the VMs, resource pools provide a better and smoother management process and added scalability for the control of resources for groups of VMs.

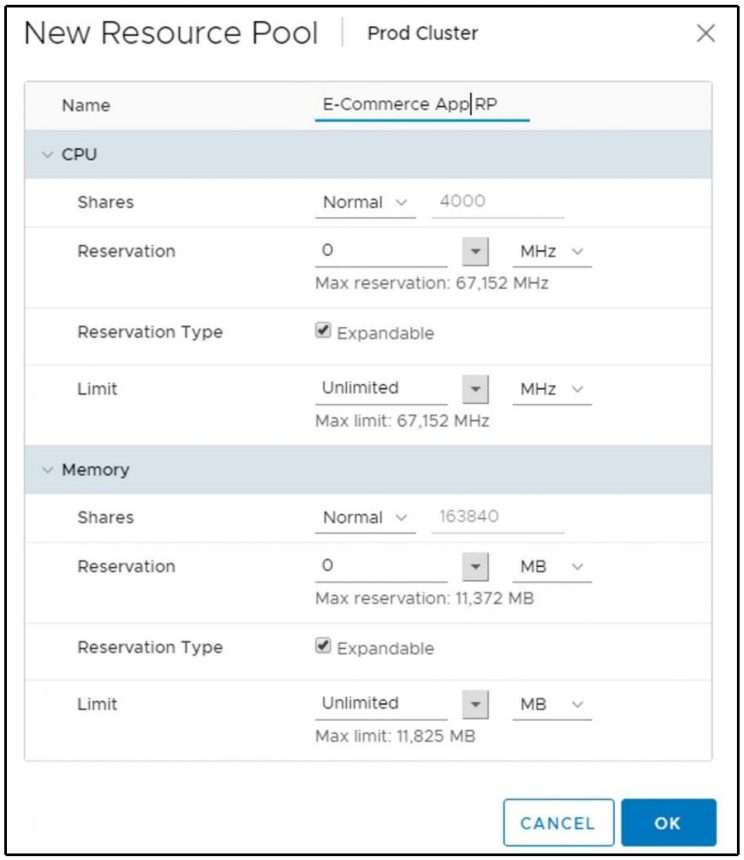
Please note that resource pools can be used only in DRS-enabled clusters.

Resource pool configuration

To create a resource pool in a cluster (this procedure is similar for the single ESXi host), proceed as follows:

1. Right-click the cluster and select the **New Resource Pool** option.
2. Specify a name for the resource pool, giving a meaningful name that's useful to identify the resource scope better.

3. Specify how CPU and RAM resources should be allocated, and then click **OK**. When the resource pool has been created, you can start adding VMs to it. Share values set as **High**, **Normal**, or **Low** specify share values in a 4:2:1 ratio, as shown in the following screenshot:

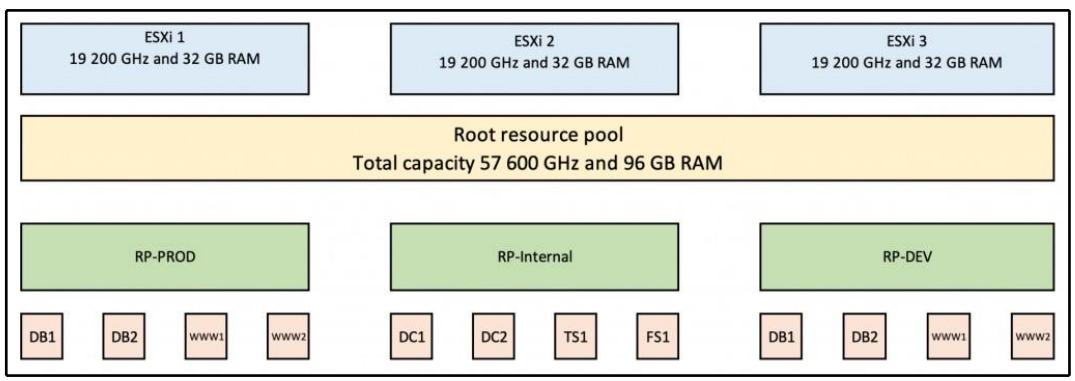


Reservations, **Limits**, and **Shares** work the same as with VMs. **Reservation Type** can be set to **Expandable** (we will discuss this specific type later).

Once you create a resource pool, you can quickly move VMs to the resource pool simply using the Drag and Drop function in the inventory.

You can also create a more complex structure of resource pools. Inside of the resource pool, you can create child resource pool(s) as well:

You may be wondering how resource pools work. To explain how they work, let's take a look at the following example. Three resource pools have been created that correspond to three different departments—**RP-PROD**, **RP-Internal**, and **RP-DEV**:



A configuration of the resource pools is shown in the following table:

Cluster	CPU shares	CPU limit	CPU reservation	Memory shares	Memory limit	Memory reservation
RP-PROD	Normal	Unlimited	20 GHz	Normal	Unlimited	32 GB
RP-Internal	Normal	Unlimited	10 GHz	Normal	Unlimited	16 GB
RP-DEV	Normal	10 GHz	0	Normal	16 GB	0

And following VMs are created:

VM	CPU	RAM	RP
DB1	8 vCPU / 19.2 GHz	16 GB	RP-PROD
DB2	8 vCPU / 19.2 GHz	16 GB	RP-PROD
WWW1	4 vCPU / 9.6 GHz	8 GB	RP-PROD
WWW2	4 vCPU / 9.6 GHz	8 GB	RP-PROD
DC1	1 vCPU / 2.4 GHz	4 GB	RP-Internal
DC2	1vCPU / 2.4 GHz	4 GB	RP-Internal
TS1	4v CPU / 9.6 GHz	12 GB	RP-Internal
FS1	2 vCPU / 9.6 GHz	8 GB	RP-Internal
WWW1	2 vCPU / 4.8 GHz	8 GB	RP-DEV
WWW2	2 vCPU / 4.8 GHz	8 GB	RP-DEV
DB1	4 vCPU / 9.6 GHz	16 GB	RP-DEV
DB2	4 vCPU / 9.6 GHz	16 GB	RP-DEV

Our clusters consist of three ESXi hosts, each containing eight physical CPU cores at 2.4 GHz and 32 GB RAM.

The overall cluster capacity is 57.6 GHz of CPU power and 96 GB RAM. If every VM consumes 100% of the configured resources, the total amount of required resources is 110.4 GHz and 124 GB of memory, which does not fit the cluster. Now, the resource pools come into play. Let's assume that all VMs are 100% utilized; *what will the resource allocation for the VMs be?*

First, the reservation must be satisfied. Based on that, we have 27.6 GHz of CPU power to be distributed and 48 GB of memory.

Resource pools are configured on the same level. Thus, the remaining resources are divided between resource pools using shares:

Resource pool	Reservation for CPU	Remaining resources based on shares for CPU	Total available resources for CPU	Reservation for memory	Remaining resources based on shares for memory	Total available resources for memory
RP-PROD	20 GHz	9.2 GHz	29.2 GHz	32 GB	16 GB	48 GB
RP-Internal	10 GHz	9.2 GHz	19.2 GHz	16 GB	16 GB	32 GB
RP-DEV	0	9.2 GHz	9.2 GHz	0	16 GB	16 GB

So now the shares, reservations, and limits will be applied if they are configured on the individual VMs within the resource pool. If no RLS on the VMs are configured, each VM will get an equal amount of resources, so in the case of RP-PROD VMs the allocation will be as follows (no RLS is configured on any VM):

VM	CPU	RAM	RP
DB1	7.3 GHz	16 GB	RP-PROD
DB2	7.3 GHz	16 GB	RP-PROD
WWW1	7.3 GHz	8 GB	RP-PROD
WWW1	7.3 GHz	8 GB	RP-PROD

Memory allocation is not affected because the total size of configured memory does not exceed the Total Available Resource for Memory, but the CPU will be throttled for the VMs since the required power is 57.6 GHz, but the Total Available Resource for CPU is 29.2 GHz.

Using resource pools, resources assigned to a group of VMs can be adjusted from a single point with no need to edit every single VM.

Keep in mind that you can configure RLS settings on multiple levels, so the resource hierarchy might be quite complicated to calculate.

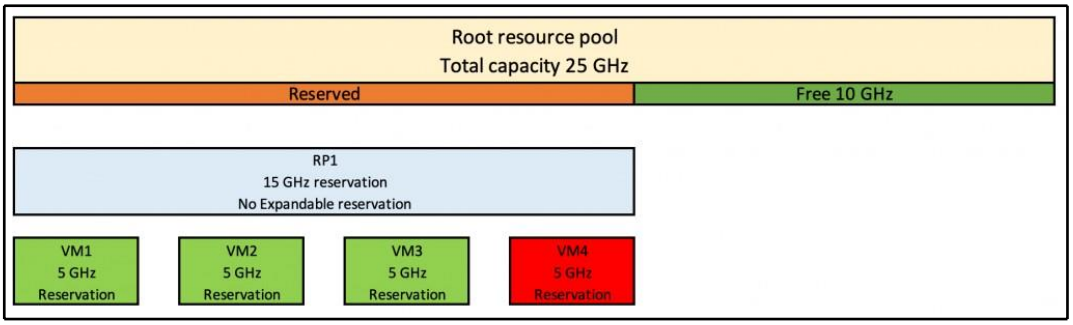
Expandable resource pool

An expandable option is related to the reservation that's configured on the resource pool. If you use reservation on the VM, DRS makes sure that the resources are available and if not, the VM will not be started.

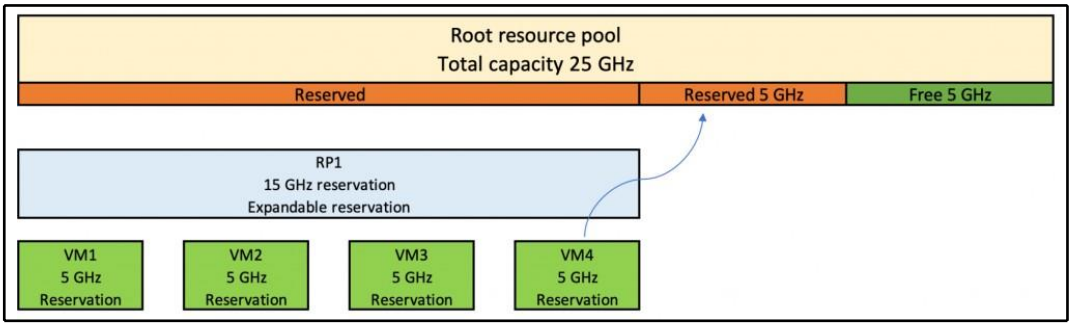
Imagine a situation where you have created a resource pool with 15 GHz of CPU resources and then four VMs, each with 5 GHz of CPU resources reserved.

If you try to power-on all of the VMs and expandable reservation is not checked, only the first three VMs will be powered on, not the last one.

Why? The first VM will claim 5 GHz of CPU resources from the RP (which has a reservation of 15 GHz), the second one and the third will do the same, but when the fourth VM tries to claim resources, there won't be enough resources since the remaining capacity of the resource pool is 0 GHz, and the VM wants to claim 5 GHz, as shown in the following diagram:



If you enable expandable reservation, the remaining capacity will be reserved on the parent resource pool (if multiple parent-child resource pools are configured) or from the root resource pool (total cluster capacity):



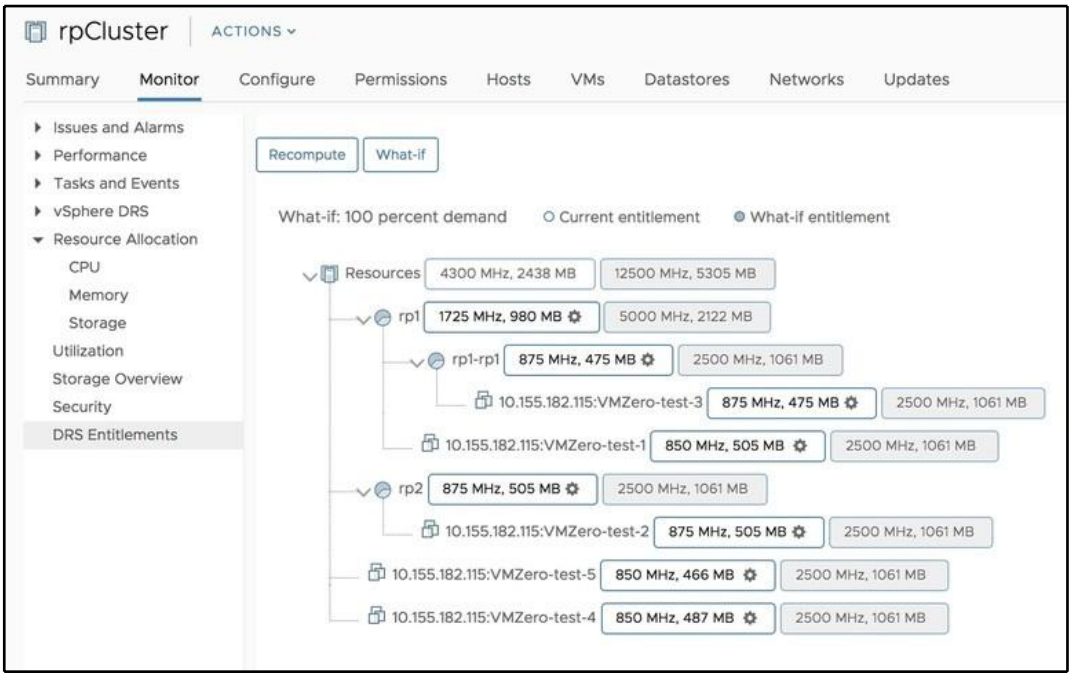
So, again, multiple resource pools (parent-child) can be configured, so if the expandable resource pool cannot satisfy the resource reservation, it will try to reserve the resources on the parent object. If the parent resource pool has the expandable option configured, it will try to reserve the resources on its parent object. If the expandable option is not enabled, the resource reservation fails, and the VM is not powered on.

Resource allocation monitoring and calculations

As you have already seen, working with the reservations, shares, and limits on multiple levels might be a tricky job, and unfortunately, there is no mechanism directly built into vSphere to help you with the calculations.

You can either use Excel or any other tool to calculate the resources and model what-if scenarios based on the resource consumption. You can use DRS Entitlement Viewer, a free tool available at VMware Labs (`https://labs.vmware.com/flings/drs-entitlement-viewer`).

DRS Entitlement Viewer is a small application that can integrate with the vSphere web client (HTML5 only), and it can do all the calculations for you:



I would strongly suggest installing this Fling to your vSphere environment, especially if you are working with multiple resource pools and you have configured RLS settings on multiple items.

The installation is simple—all you need to do is download the Fling itself and follow these instructions:

1. Unzip the plugin package to the `/usr/lib/vmware-vsphere-ui/plugin-packages/` folder.
2. Add the advanced options to the cluster:
    - `CompressDrmdumpFiles` – 0
    - `DrmdumpResActions` – 1
3. Restart the vsphere-ui service:
    - `service-control --stop vsphere-ui`
    - `service-control --start vsphere-ui`

Once installed, you can see the new DRS entitlements section under the **Monitor** tab for each cluster.

Managing resource pools

Once created, you can edit, delete, add, or remove VMs from the resource pools:

- **Edit a resource pool**: From the vSphere client, right-click on the object and select **Edit Resource Settings**. Change the CPU and RAM settings and then click **OK** to confirm.
- **Delete a resource pool**: From the vSphere client, right-click the resource pool and select **Delete**. Click **OK** to confirm the deletion. Deleting a resource pool doesn't delete the VM it contains.
- **Adding a VM**: A VM can be added to a resource pool during the creation process using the migrate functionality or using the drag and drop feature.
- **Removing a VM**: From a resource pool, right-click on the VM to remove, and select the option **Migrate to move it to another resource pool**. Use the drag and drop feature to move a VM off the resource pool instead.

vApps

VMware vApps is perhaps one of the most underutilized features of vCenter Server. A vApp is an application container, such as a resource pool.

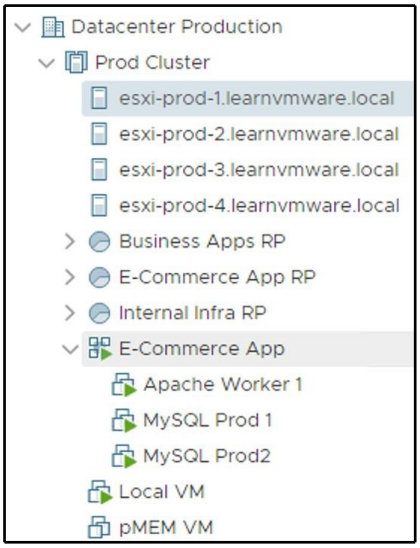
You can assign multiple VMs to a vApp and then treat this set of VMs as a single application. You can configure the startup and shutdown order of the VMs or monitor the utilization and health of the vApp itself, instead of individual VMs.

Let's imagine a situation—you have a two-tier application, consisting of two database servers and one web frontend server. Those VMs are used only for this e-commerce application, so you can create a vApp and move VMs inside the vApp. On the vApp level, you can configure the resource parameter in terms of reservations, limits, and shares for the whole vApp, not the individual VMs, and you can treat the vApp like a VM. Let's say you need to deploy another instance of the application. All you need to do is to clone the whole vApp instead of cloning all of the VMs. Lastly, you can configure the boot and shutdown order of the VMs within a vApp. First, you need to start the databases, and only after that do you start the web frontend.

Creating a vApp is a straightforward process:

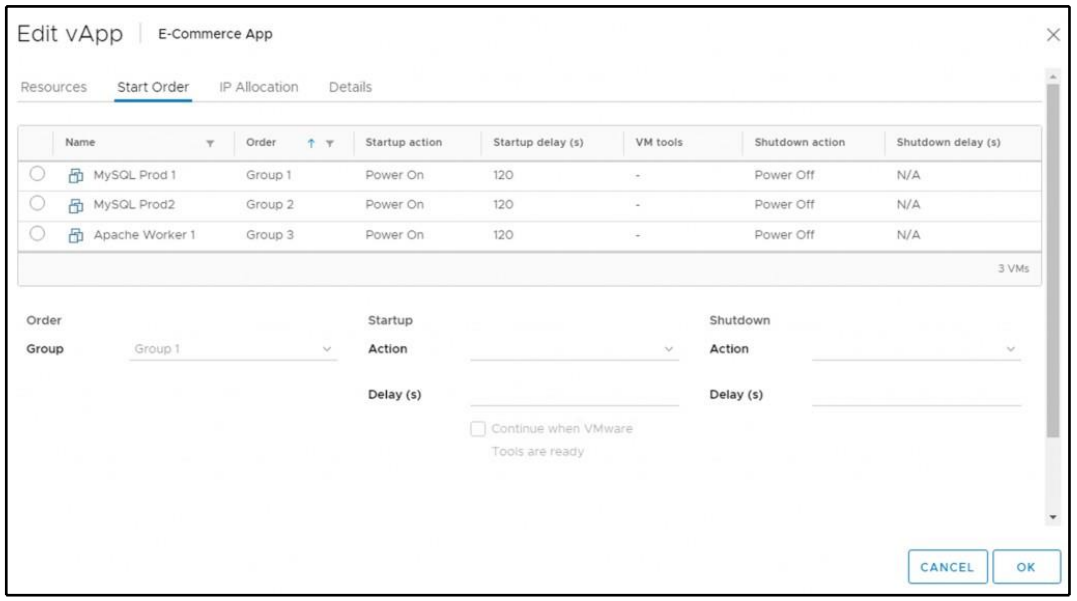
1. In the **Hosts and Clusters** view, click in the inventory and select **New vApp**
2. A vApp can be created on the ESXi hypervisor, in a cluster, or even inside the parent resource pool
3. Provide a name and location for the vApp
4. You have an option to configure the resource allocation of the vApp, the same as with a resource pool
5. Review the settings and create a vApp

Once the vApp is created, you can drag and drop VMs to the vApp:



In the Hosts and Cluster view, you can see individual VMs in the vApp. In the **Virtual Machines and Templates** view, all you can see is the vApp itself.

You can configure the vApp to change the resource allocation settings or to configure the **Startup** and **Shutdown** order:

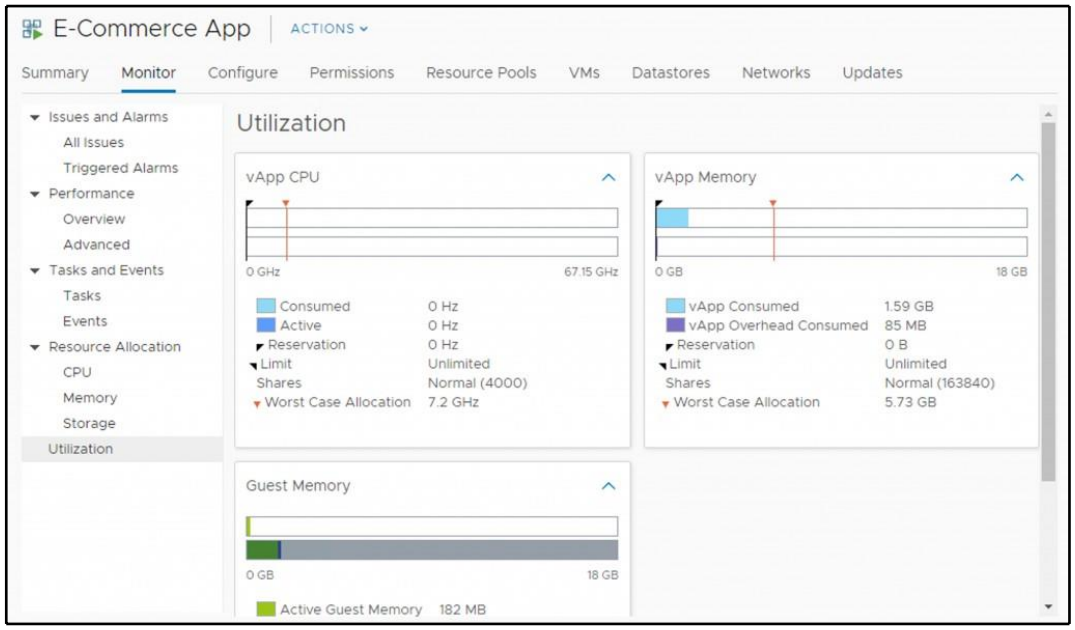


A VM within the vApp can be part of a single group, but multiple VMs can also be part of a single group. For instance, you have three web frontend servers. You do not care in which order the individual frontend servers start, but you need to start them only after the HTTP load balancer successfully starts. In such a case, you create two groups. Load balancers will be placed in **Group 1**, and all web frontend servers will be placed in **Group 2**.

Now, if you decide to shut down the whole vApp, you can click on the vApp and select shutdown or power off, and all VMs within the vApp will be turned off based on the start order.

Using the resource allocation on the vApp level, you can easily configure reserved resources for the vApp (the same as with resource pool) or the limits. Then, on the individual VMs inside the vApp, you can configure shares that will be applied during the resource contention of the vApp.

You can check the current vApp resource utilization in the **Monitor** tab of the **Utilization** section:



Network and storage resources

vSphere 6.7 includes additional resource management features that are useful for optimizing the virtual infrastructure performance and the efficiency of hardware components, such as storage devices and network resources.

Network resources can be allocated and controlled through the vSphere Network I/O Control feature to solve situations of resource contention.

Storage I/O Control (**SIOC**), **Storage DRS** (**SDRS**), **Storage-Based Policy Management** (**SBPF**), and other storage-related features, are used to control and optimize storage performance and resource availability.

From the single VM perspective, the configuration is the same as with computing resources. You can configure reservations, limits, and shares for both network and storage resources.

15
Availability and Disaster Recovery

This chapter will focus on specific availability (and resiliency) solutions in vSphere, whereas, with the VMware technology, it is possible to create an entire HA infrastructure on every level.

When we think about how we can work with all the benefits of VMware solutions, we see a lot of positions and levels where it is possible to protect the infrastructure. Still, the physical and infrastructural parts of your environment are really important and crucial, so the right design and configuration are needed for your hardware, such as servers, switches, storage, and **host bus adapter** (**HBA**), in order to design and configure all of them without a potential single point of failure.

However, only the infrastructure level is essential and must be resilient. Also, workloads need a good HA level according to business requirements and needs. There are not only several native solutions that can be used, such as NIC teaming, multipathing for storage, vSphere **High Availability** (**HA**), and vSphere **Fault Tolerance** (**FT**), but also other solutions, typically from the physical world, such as guest clustering.

In this chapter, we will cover the following topics:

- vSphere HA
- vSphere FT
- Virtual machine clustering
- Virtual machine backup
- VMware vSphere Replication
- Disaster recovery and disaster avoidance
- VMware solutions

VMware vSphere HA

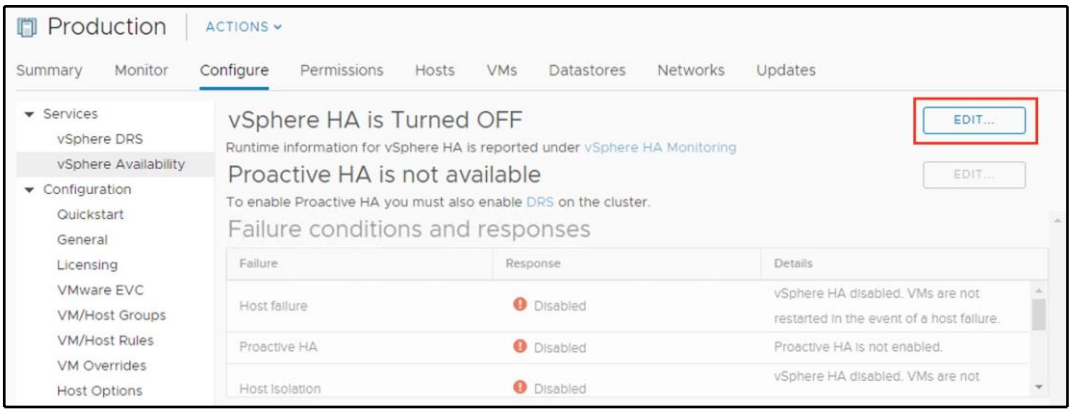
vSphere HA is the most commonly used technology in the vSphere Suite in terms of HA. vSphere HA is responsible for restarting VMs in cases of ESXi downtime. Highlighting the word **restart** is crucial. There is a hard restart of the VM on the other ESXi hypervisor within the same vSphere cluster.

Keep that in mind that your OS will boot from scratch after the HA event because, in such an event, it is not possible to synchronize the memory of the VM between hosts.

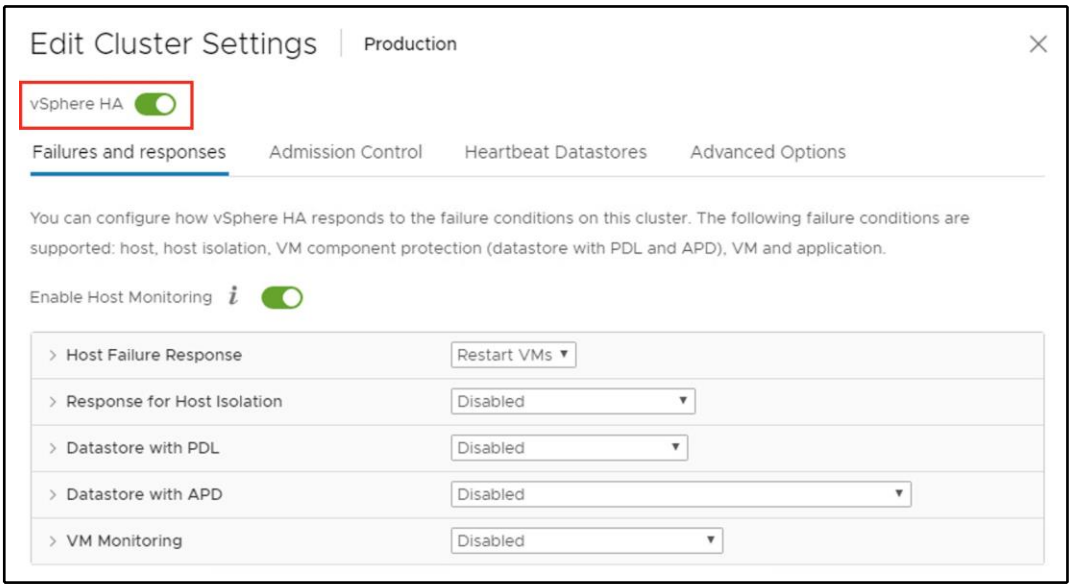
We will have a look at the different aspects of the vSphere HA in the next few sections, but now let's have a look at how to enable HA.

vSphere HA configuration

Configuring vSphere HA is a part of the cluster configuration in vSphere Web Client; click on the **EDIT...** button in the **vSphere Availability** area, as shown in the following screenshot:



Configuration in the basic state is straightforward; click on the **Turn ON vSphere HA** checkbox and then **OK**:



Under the **Recent Tasks** area, you can see **Configuring vSphere HA** on each host, so wait for success on all hosts of the cluster:

When you click on **Turn ON** on every ESXi host in the cluster, a special agent called **Fault Domain Manager** (**FDM**) is installed. A log specific to the FDM agent is available on each host in the `/var/log/fdm.log` file.

Now we have to enable HA and protect VMs. When a host fails, vSphere HA restarts the VMs on other hosts in the cluster. In each cluster, there is a master host that is responsible for managing all other slave hosts, and the master node reports the current state of the cluster to the vCenter server. The vCenter server is used to configure the vSphere HA, but the master node is responsible for HA events, so even if your vCenter server is not available, the vSphere HA will work as expected.

To configure vSphere HA, we have the following requirements:

- **Right license**: Minimal Essential Plus
- **Two hosts minimal**: 64 hosts, which is the maximum per cluster in vSphere 6.0 and 6.5
- **At least two share datastores**: These are required for datastore heartbeats, but you can also have a single shared datastore (in this case, you need an advanced option to remove the warning on the datastore numbers)
- **vCenter Server**: Needed to configure vSphere HA, but is not involved in actual HA failovers

vSphere HA heartbeats

Heartbeats are used to ensure that the host is up and running. If no heartbeats are received within a configured timeframe, the host is considered down, and the HA event will occur, restarting the VMs running on the unresponsive host.

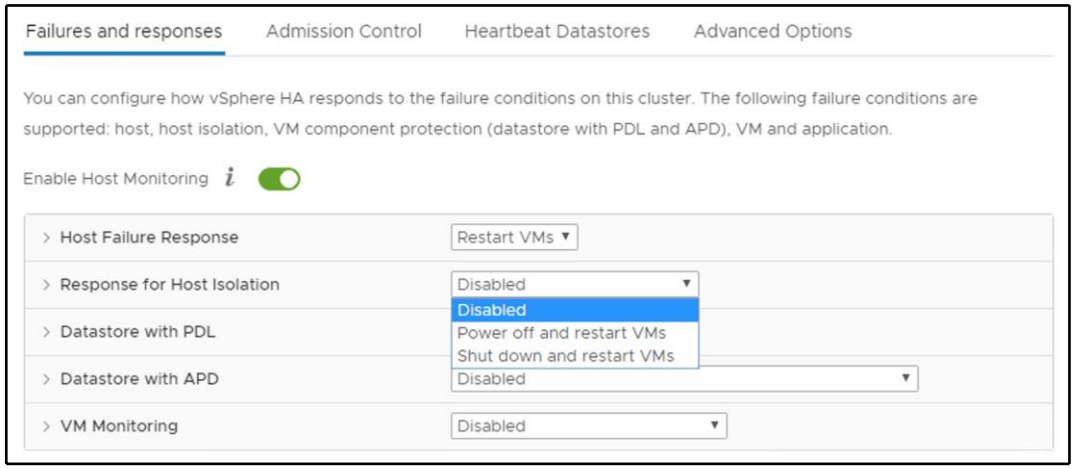
There are two kind of heartbeat:

- **Network heartbeat**: The master node will periodically check if the slave node is available over the network using ICMP pings.
- **Storage heartbeat**: Each ESXi hypervisor stores an empty file on the shared datastore, which is exclusively locked by the ESXi hypervisor. If the file is not locked, it means that the hosts have lost access to the storage.

vSphere HA network heartbeats

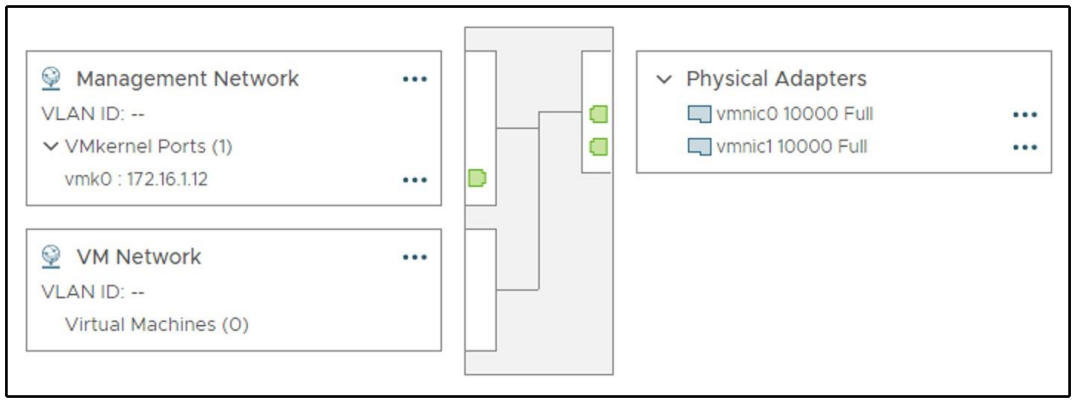
The master ESXi server sends ICMP pings to the slave hosts over the management network. The management network is a port group that is configured for management traffic.

Also, by default, the default gateway is used as an **isolation address**. When the slave doesn't receive any network heartbeats from the master, it will try to reach its isolation address. If the isolation address is not reachable, the slave host will be considered isolated—meaning that access to the network is limited; based on your preferences, you can choose what to do with the VMs by configuring **Response for Host Isolation**:



You can change the isolation address from the default gateway to any other IP address based on your network design by using the advanced `das.isolationaddress[0-9]` cluster parameter. There can be up to 10 different isolation address.

It is also recommended to have a physical network that's highly available, so always use dual physical NICs to carry your management traffic as well as two physical switches to provide maximum network availability:

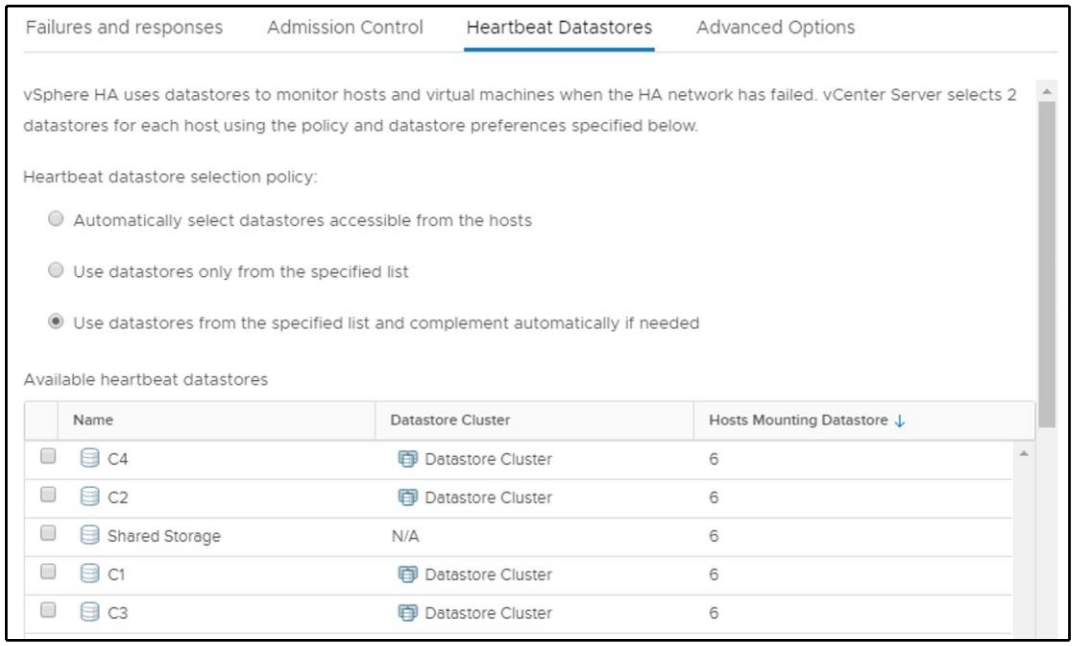


vSphere HA storage heartbeats

In addition to network heartbeats, storage heartbeats were introduced to increase the resilience of the infrastructure.

By default, two different shared datastores will be used for storage heartbeats, again to increase the resilience of the infrastructure. It is possible, however, to configure an advanced setting, `das.heartbeatDsPerHost`, to change the value to a different number of datastores, or `das.ignoreInsufficientHbDatastore`, set to true to ignore the warning.

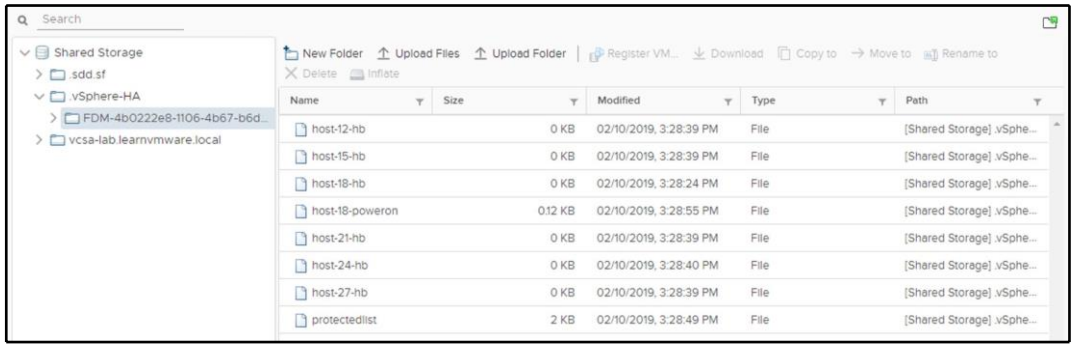
The following screenshot is the configuration page for **Heartbeat Datastores**:



Storage heartbeats are used only when network heartbeats have failed. Each host will create a specific file on the shared datastore that can be accessed by all other ESXi hosts. This file, `host-XXXX-hb`, is an empty file. The corresponding ESXi hypervisor exclusively opens that. If the ESXi hypervisor loses access to the storage, the master ESXi node will discover that the lock from the file disappears, meaning that the host has lost access to the storage and an HA event will occur.

vSphere HA protection mechanism

How does HA work? Each ESXi server that is a part of the cluster will maintain its `poweron` file, which contains the list of VMs running on the host. This file is located on your `vmfs` volumes in the hidden. `vSphere-HA/<FDM cluster ID>` directory, as shown in the following screenshot:



The `poweron` file is not just used only to track which VMs are running but also for identification if the host is isolated . The first line of the `poweron` file contains the isolation identification, and it contain either a `0` (zero) or a `1`. A `0` means that the host is not isolated, and a `1` means that it is isolated. The master is responsible for informing vCenter about the isolation status of the host.

When the lock from the heartbeat file is lost, the master ESXi node will check the corresponding `poweron` list and initiate a restart of the VMs on the other ESXi nodes.

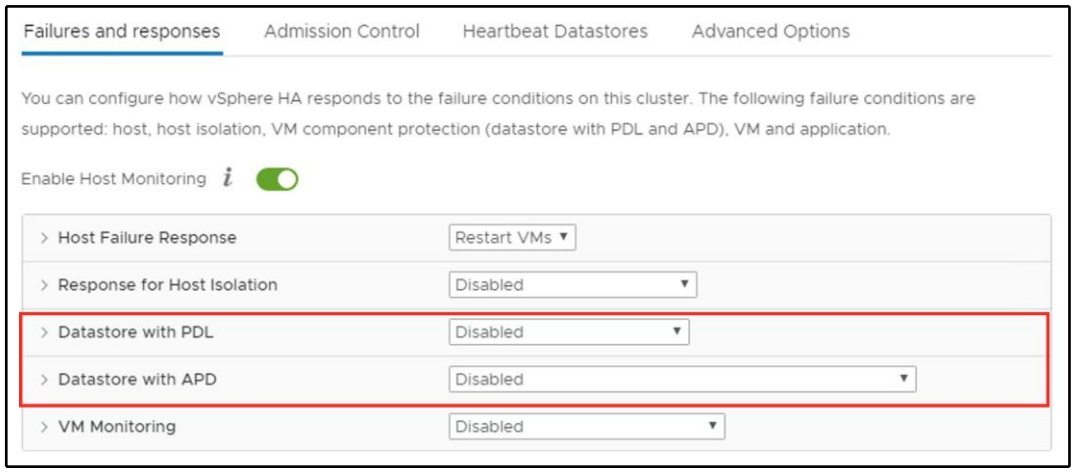
Virtual Machine Component Protection (VMCP)

VMCP is a new technology from vSphere 6.0. VMCP protects storage failures, used typically for block-based storage. Before vSphere 6.0, vSphere HA only protected hosts by monitoring network failures; if the FDM agent is not able to talk with other agents, it can be considered in a fault state by the other agents, and it will be considered in an isolated state itself.

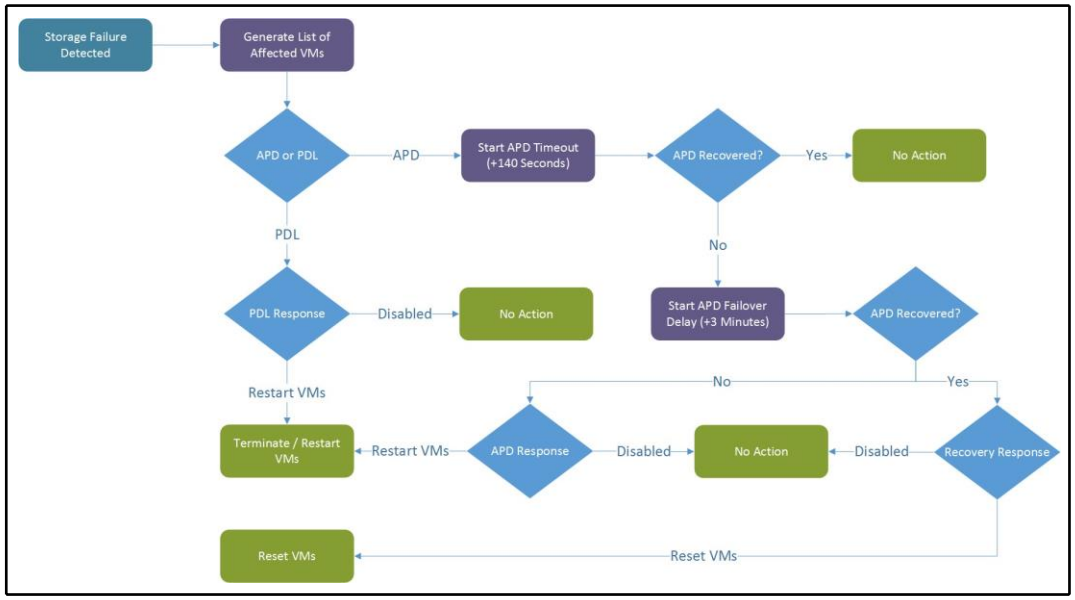
The following two types of storage failure can now be handled:

- **PDL**: When the storage array has lost access to the device (the device is offline or unavailable) then a specific SCSI sense code is sent to the ESXi server. When PDL situation occurs, the ESXi server will stop sending I/O commands to the device.
- **APD**: If the ESXi host is not able to reach the storage array at all (no SCSI sense codes are received) then the storage device is marked as unavailable, but since the ESXi server validates the status of the storage array, it will still try to resend I/O commands to the device until APD Timeout is reached.

When you want to use VMCP, you must configure the **Failures and responses** tab of the configuration:



The VMCP recovery workflow is shown in the following diagram:



Proactive HA

Proactive HA is a new feature integrated with server vendor monitoring systems. The idea is to provide HA to VMs even before an actual failure occurs. For example, the server has two redundant power supplies, and one of them fails. There is no direct impact on the ESXi server itself, but this condition might lead to downtime when the second power supplies fail. With proactive HA, you can evacuate the VMs from the affected ESXi hypervisor event before it becomes unresponsive.

When any of the hardware fails, it is marked by the hardware monitoring agent as unhealthy. Based on the configuration, vCenter Server will classify the hardware failure as **degraded** or **severely degraded** and the host is placed in a specific state called **quarantine mode**.

In quarantine mode, DRS won't use the affected ESXi server for the placement of any new VMs, and it might even proactively evacuate the VMs of the ESXi host. Evacuation will only take place in situations when the migration will not affect the performance of the VMs. There is an option to place the host into **maintenance mode** as the result of degraded state.

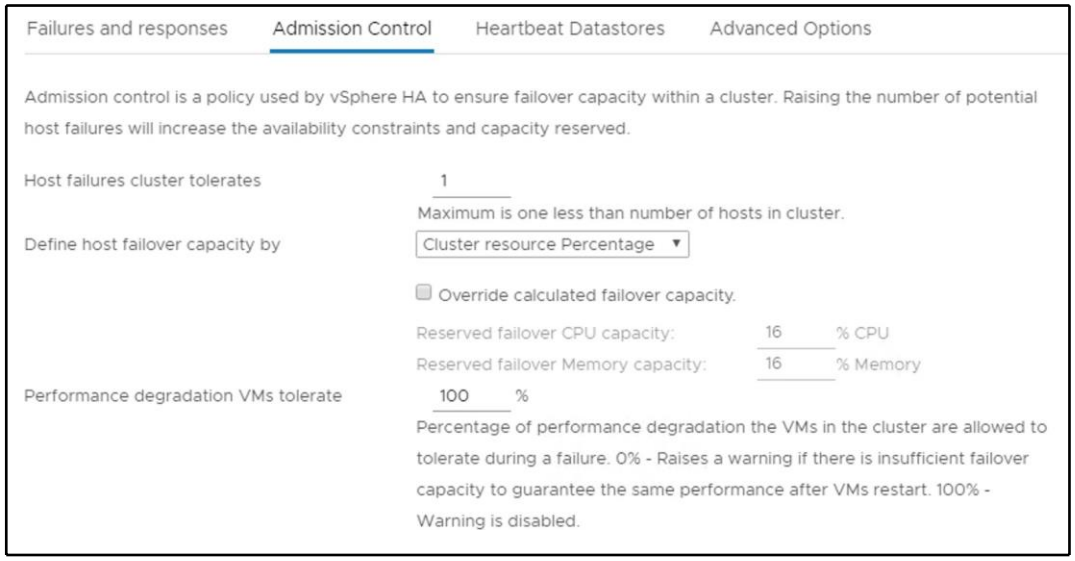
Remediation actions available with proactive HA include:

- **Quarantine mode for all failures**: Quarantined hosts are not used to run new VMs, but currently running VMs will still be run on the top of the quarantined host.
- **Mixed mode (quarantine mode for moderate and maintenance mode for severe failure)**: If the host suffers from moderate degradation, the VMs will be kept running on the host, but new VMs will not be run on the host. However, all VMs will be migrated off the host in severe failures.
- **Maintenance mode for all failures**: All VMs will be migrated off the host no matter whether moderate or severe degradation occurred.

Admission control

Admission control guarantees vSphere HA failover by ensuring enough spare failover capacity within the cluster. If you have four ESXi hosts and all of them are utilizing almost 100% of the total compute capacity, and one of the ESXi hypervisors fails, the VMs will be restarted, but the remaining hosts won't be able to provide the required number of the resources other VMs thus performance will be degraded. Admission control takes care of that by reserving resources for the failover.

Configuration is at the cluster level under the **vSphere Availability** section, as follows:



You can define the host's failover capacity using one of the following options:

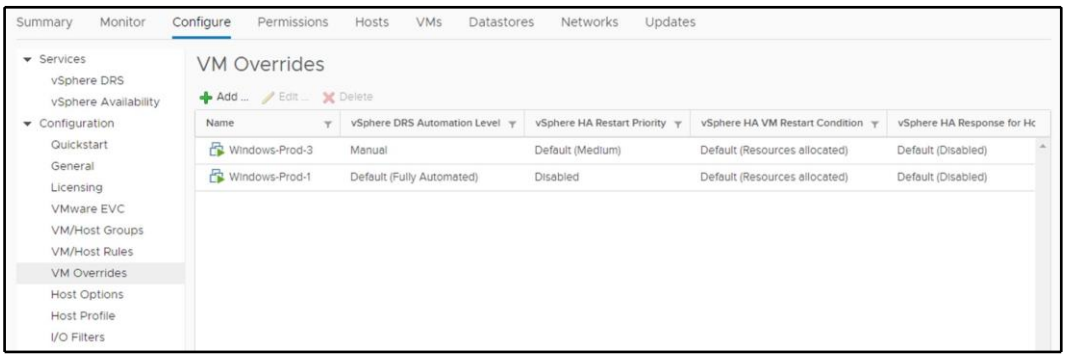
- **Disabled**: No admission control is configured, and no resources are reserved for failover.
- **Slot Policy (powered-on VMs)**: A slot is a logical representation of memory and CPU resources. A slot is the memory and CPU reservation required for any powered-on VMs in the cluster. Slot Policy can do good work in the environment where there is a very similar VM with the same CPU memory configuration. When you have a lot of small VMs and two monster VMs, it is not a very good situation because the reservation selects the most significant value. You can change this value for the CPU slot size and memory slot size through **Advanced Options.**
- **Cluster resource Percentage**: You can design vSphere HA to perform affirmation control by holding a particular level of group CPU and memory assets for recuperation from host failure. vSphere HA calculates CPU and memory. CPU calculation uses CPU reservation for powered-on VMs. If you don't use reservation HA, use the default value of 32 MHz. The memory calculates the memory reservation and memory overhead of each powered-on VM; the default value is **0** MB. You can override the calculated failover capacity.
- **Dedicated failover hosts**: This is the last option for defined failover hosts. You can specify and dedicate failover hosts. vSphere HA uses such hosts when it needs failover actions or has insufficient resources.

You can find more information about different **Admission Control** policies at `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.avail.doc/GUID-85D9737E-769C-40B6-AB73-F58DA1A451F0.html`.

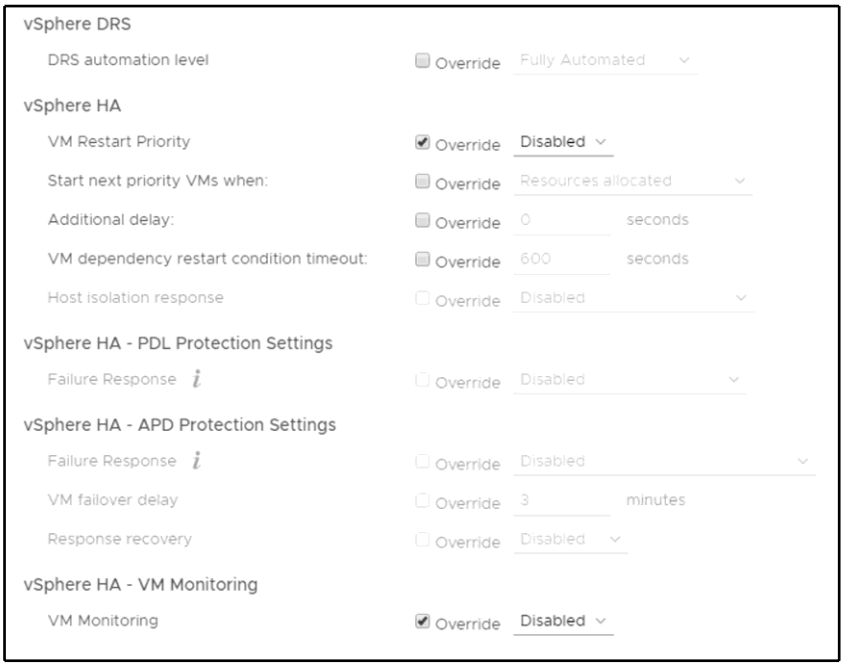
Then, there is a new option called **Performance degradation VMs tolerate**. This new setting in vSphere 6.5, if set, will issue a notice when a host disappointment would cause a decrease in VM execution depending on the genuine asset, not simply arranged reservations.

VM restart and monitoring

You can override specific vSphere HA (but also **vSphere DRS**) configurations for specific VMs directly from the cluster level. In the **Configure** tab, under the **Configuration** menu, select **VM Overrides**:

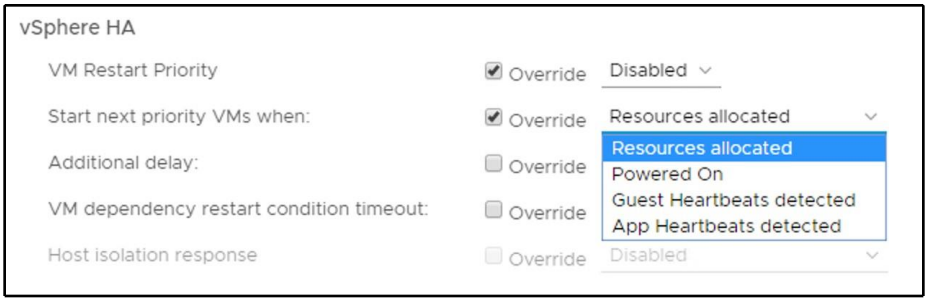


The first option is the automation level, but it's related to vSphere DRS. For vSphere HA, you can specify different restart priorities or disable vSphere HA completely, as follows:



Note that starting with vSphere 6.5 there are two new levels (lowest and highest) to provide more control. If a VM does not need to be restarted (for example, for test purposes), you can also disable vSphere HA.

The next option is called **Start next priority VMs when**, which define a condition on when the next VM should be restarted. For example, you can choose the Guest Heartbeats detected option:



To monitor vSphere HA, it is possible to use the **Monitor** tab, and the **vSphere HA** section on it containing various information about the overall vSphere HA configuration, configuration issues, datastores under APD conditions, or advanced runtime information.

VMware vSphere FT

VMware vSphere FT is a way to improve the availability level for critical VMs, with a *zero-downtime technology*.

vSphere FT works by continuously replicating the state of the VM between two different ESXi hosts. As a result there are two identical copies of a VM—the primary VM and the secondary VM (sometimes called shadow VM). Each VM has its own set of configuration files, `VMX` and `VMDK` files, which vSphere FT automatically keeps synchronized.

When the physical ESXi server where the primary VM is running fails, the secondary VM (shadow VM) automatically takes over and resumes normal operations.

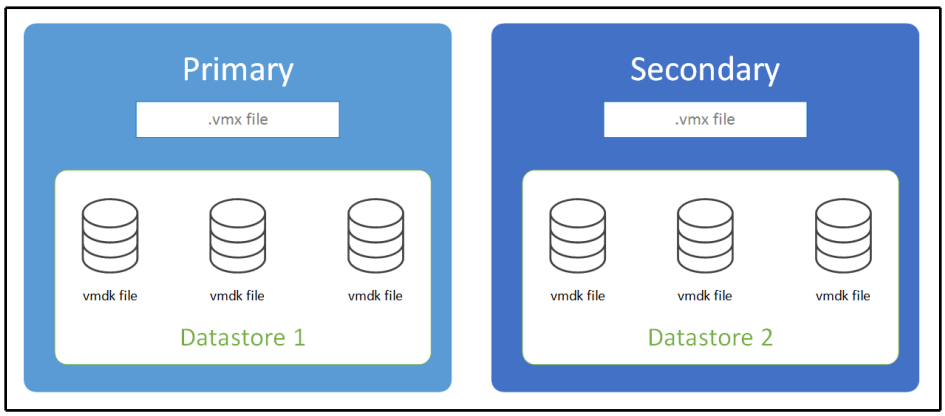
VMware vSphere FT also has some limits—for each VM, it supports a maximum of 4 vCPUs and 64 GB RAM. For each host, it supports a maximum of 4 fault-tolerant VMs. VMware vMotion migration is supported for both VMs, as are the different virtual disk formats and the native backup capability (using VM snapshots).

The requirements for vSphere FT are as follows:

- The physical CPU used in hosts for FT must be the same family or **Enhanced vMotion Compatibility** (**EVC**) must be configured on the cluster.
- The network for FT logging must use a 10 Gbps speed. A dedicated FT network is highly recommended.
- For licensing, only Enterprise Plus allows up to 4 vCPUs; with the Standard Editions, the maximum is 2.

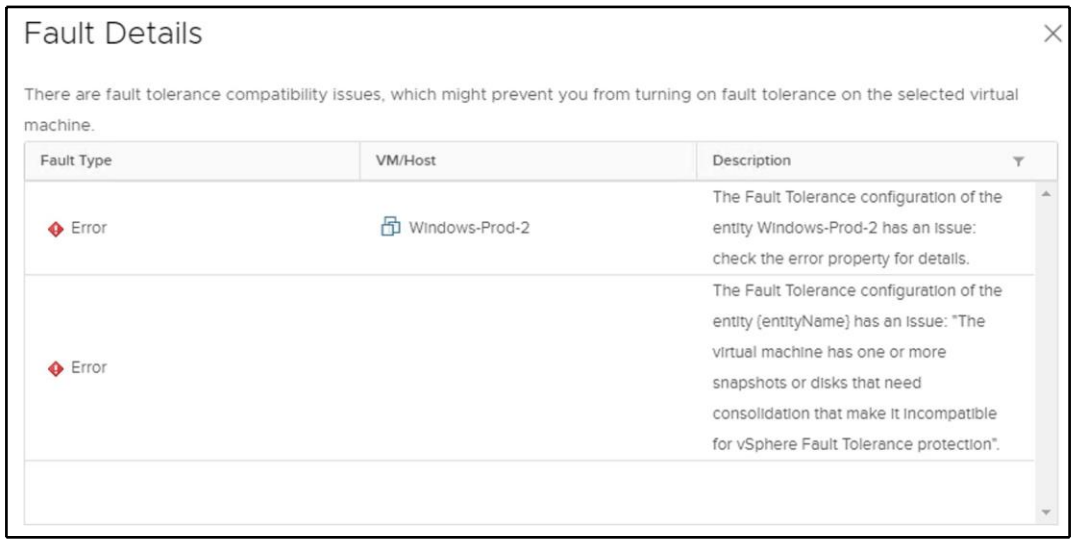
FT provides the following HA benefits:

- Continuous availability with zero downtime and zero data loss:
  - Transparent to guest OS
  - Independent on guest OS and application; any VM or application can be protected using FT technology
  - Zero-downtime failover from primary to secondary VM
- Fault tolerance improvements in vSphere 6.x:
  - Now you can protect VMs with up to 4 vCPUs and 64 GB RAM
  - vMotion is supported for both the primary and secondary VM
  - Supports backing up FT VMs
  - All disk types are supported (thin, thick eager-zeroed and thick lazy zeroed)
  - Each VM has its own set of VM files, such as `.vmx` and `.vmdk`, and both can be on different datastores:



Features not supported for FT-enabled VMs are as follows:

- **Storage vMotion**: Storage vMotion cannot be used in conjunction with FT-enabled VMs.
- **Linked clones**: With FT, you can use the Linked Clones functionality.
- **VMCP**: If you use VMCP, VM overrides are automatically created for each FT-enabled VM and the VMCP is disabled for the VMs.
- **Virtual volume datastores:** VVoLs are not supported with FT. FT-enabled VMs cannot use the VVoL datastore.
- **Storage-based policy management:** SPBM policies cannot be used together with FT-enabled VMs.
- **Snapshots**: VM must not have any snapshots once FT is enabled. Besides, it is not possible to take snapshots of VMs on which FT is enabled. Otherwise, the following error will be displayed:



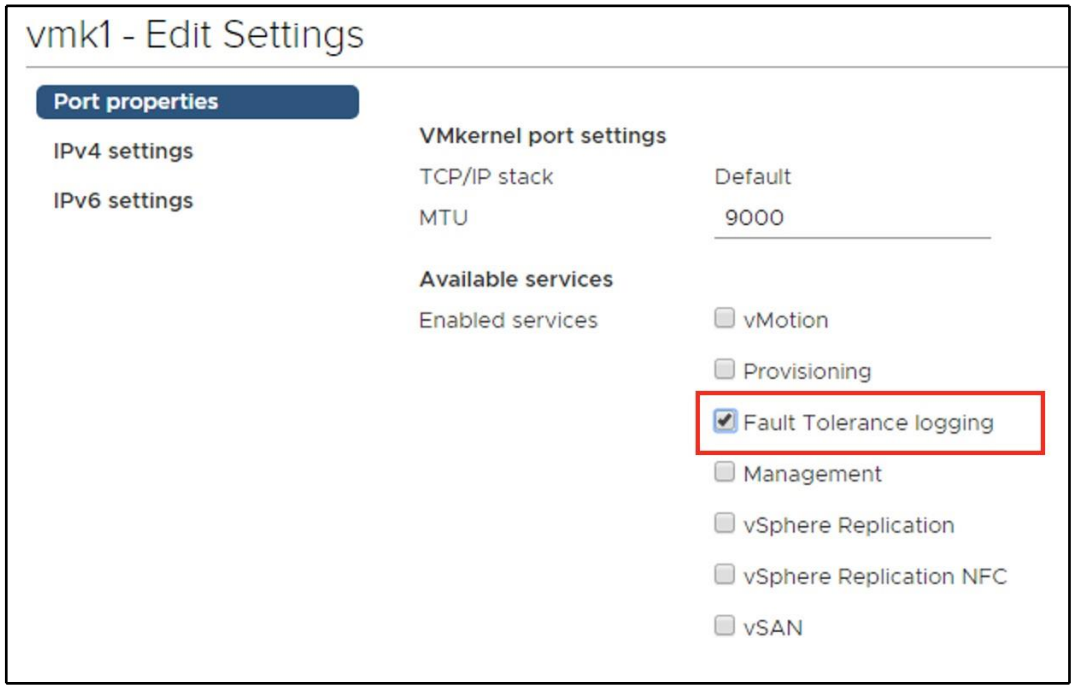
For more information about Fault Tolerance, please visit the official paper, **VMware vSphere 6 Fault Tolerance Architecture and Performance**, at `https://www.vmware.com/files/pdf/techpaper/VMware-vSphere6-FT-arch-perf.pdf`.

FT configuration

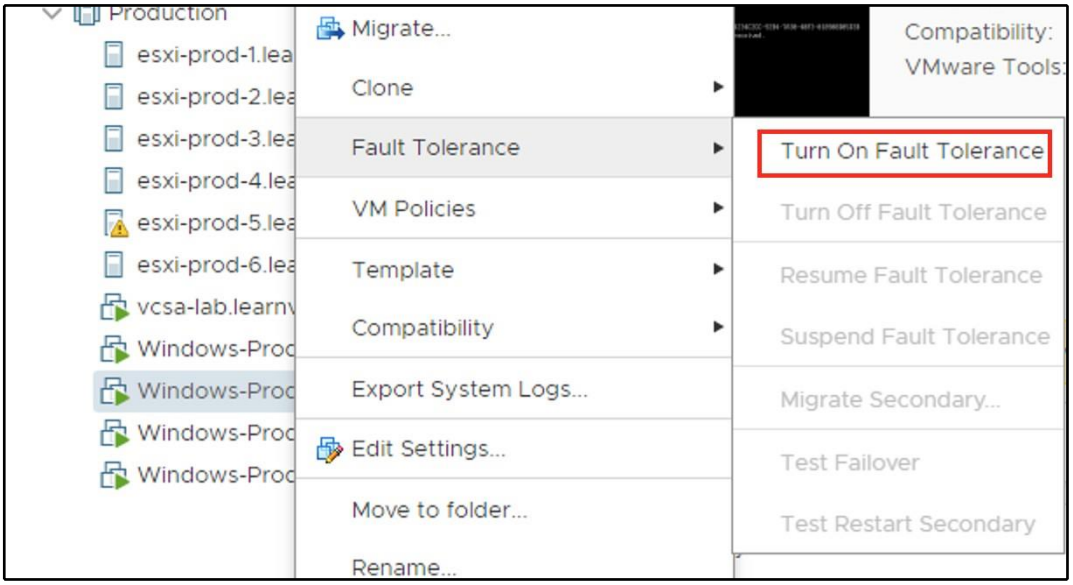
The configuration of the FT is a simple process. It is performed on the individual VM level.

As FT traffic might generate a significant amount of bandwidth, it is highly recommended you use a dedicated physical network interface card for such reasons. If you are not able to dedicate a NIC for FT traffic, you should consider using other **Quality of Service** (**QoS**) features such as **Network IO Control** to guarantee specific bandwidth to different traffic types on the shared physical media.

In both cases, prior to configuring the FT on the VM, you need to configure a VMkernel port that will be used for FT traffic and enable the **Fault Tolerance logging** service on the adapter. Again, it is highly recommended you use a dedicated VMkernel adapter for FT traffic, although you can, of course, enable the **Fault Tolerance logging** service on the existing adapter:



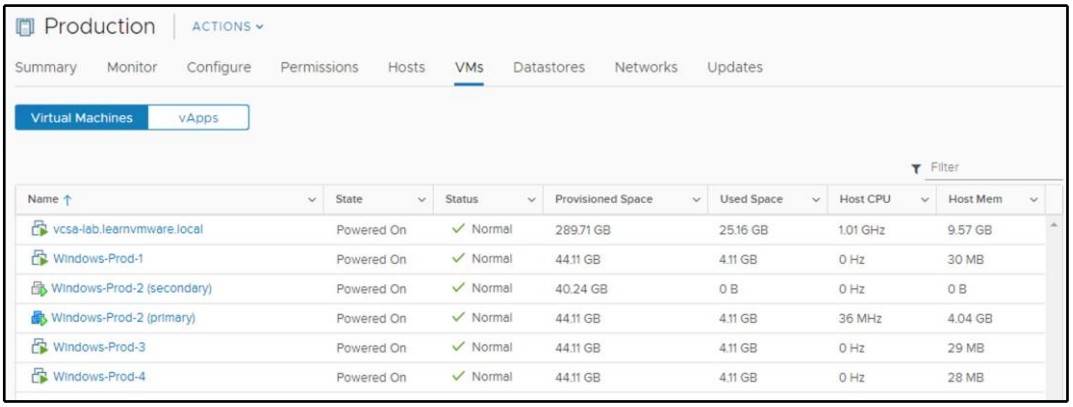
Once the network configuration is done, VMware vSphere FT can be easily activated or deactivated. To turn on this feature, right-click on the VM, select **Fault Tolerance**, and **Turn On Fault Tolerance**:



You need to select which datastore the secondary VM will be stored on as well as which ESXi hypervisor it should run on.

You cannot select the same ESXi or datastore as the one the source VM is located in.

Once the FT is configured, you might notice that the icon of the VM has changed so you can quickly identify those VMs protected by FT:



Working with FT-enabled VM

When the FT is enabled, you can perform several FT-related operations on the VM:

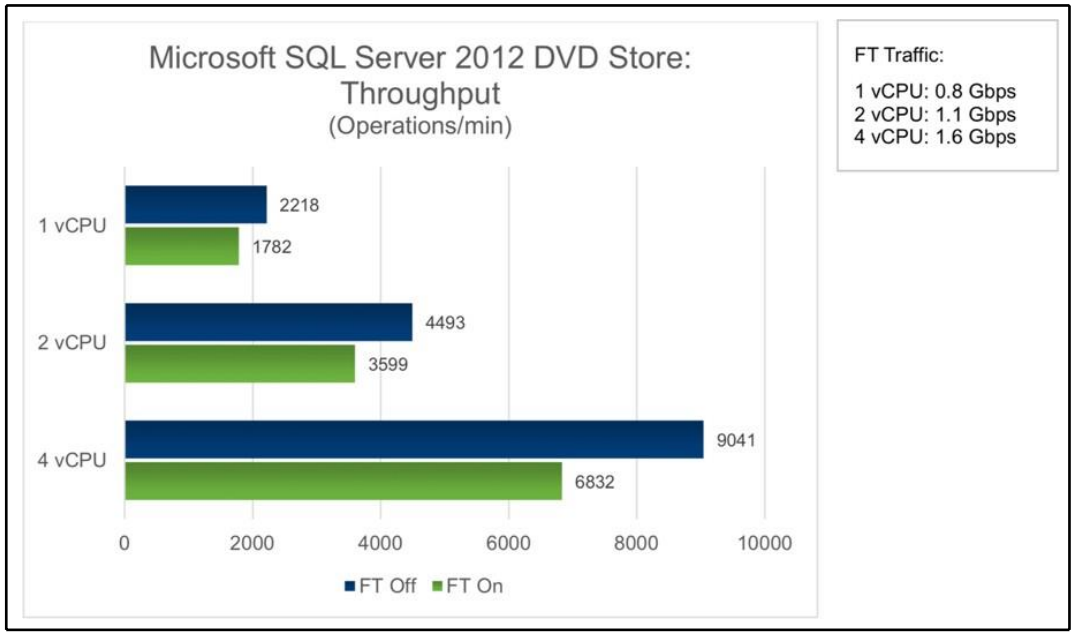
- **Turn Off Fault Tolerance**: The secondary VM is deleted, and FT is unconfigured on the primary VM.
- **Suspend Fault Tolerance**: FT logging is suspended, so the VMs are not in synchronization anymore, but the secondary VM is not removed from the infrastructure.
- **Resume Fault Tolerance**: Resumes the FT logging after the suspend operation; any changes made to the primary VM will be synced to the secondary VM and standard operations will resume, resulting in continuous synchronization of the primary and secondary VM.
- **Migrate Secondary**: You can migrate the secondary VM to the different ESXi hosts, for example, in case you need to perform maintenance on the ESXi hypervisor that hosts the secondary VM.
- **Test Failover**: In this situation, the primary VM is switched between two different hosts and the secondary VM begins the synchronization process from the beginning. This is a non-intrusive test for the primary VM.
- **Test Restart Secondary**: The secondary VM is stopped and destroyed and the new secondary VM is started and synced with the primary VM.

FT performance implications

VMs protected by FT will have lower performance due to the fact that every single CPU instruction, memory change, or storage I/O needs to be replicated to the secondary VM. Depending on the workload, FT might also generate extensive network traffic.

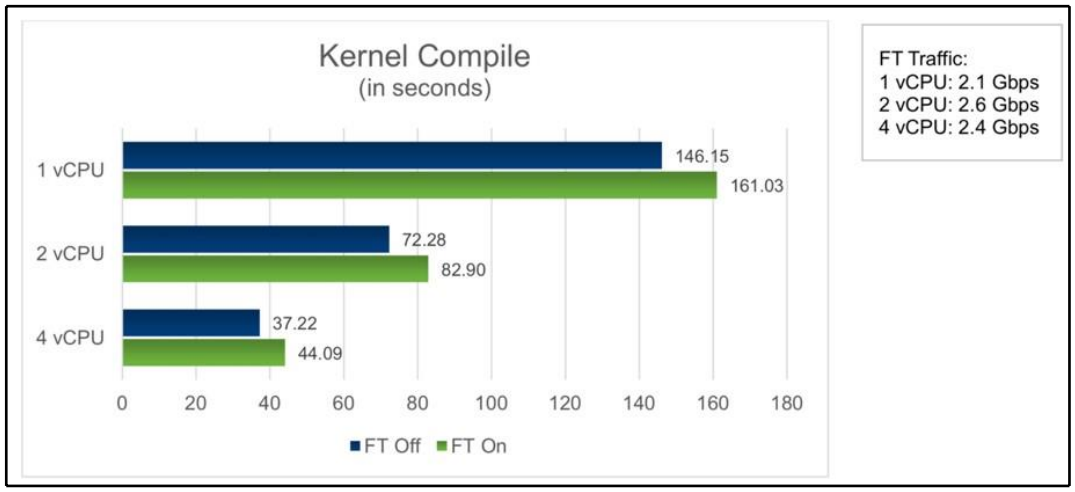
As with almost any technology, there is always some trade-off. In case of FTs, you get zero-downtime for your VMs and applications, but with a performance hit on the VM.

DVD Store is a benchmarking utility that can measure the maximum throughput of the SQL server:



As you can see, in the case of the DVD Store, the difference between FT-protected and unprotected VMs is between 20-30%, depending on the number of vCPUs.

Another benchmark you can use to measure your CPU performance is to compile a Linux kernel. This is a CPU-intensive operation, and as you can see, the difference between FT-protected and unprotected VMs is 10-20%:



Also keep in mind that your physical network carried out the synchronization between VMs, and depending on the application you run inside the VM, it can generate a significant amount of data.

For more information about FT architecture and performance, feel free to check the official technical paper at `https://www.vmware.com/files/pdf/techpaper/VMware-vSphere6-FT-arch-perf.pdf`.

Virtual machine clustering

Virtual machine clustering is an infrastructure configuration, where two systems and applications act as a *single, logical unit*. There is no direct connection to VMware vSphere. Clustering must be supported by the underlying operating system or the application itself.

In general, both systems must have simultaneous write access to the storage device so both can act as a primary or standby instance.

Mission-critical systems such as production databases are usually clustered, so in any situation, you have at least one instance available. For this reason, you can't use VMware HA technology because, as we have already explained, the VM restarts during the HA failover, resulting in application downtime. You can, in theory, use FT, but there will be limitations as well. In the case of FT, there are two significant disadvantages—performance degradation and support for a maximum of 4 vCPUs.

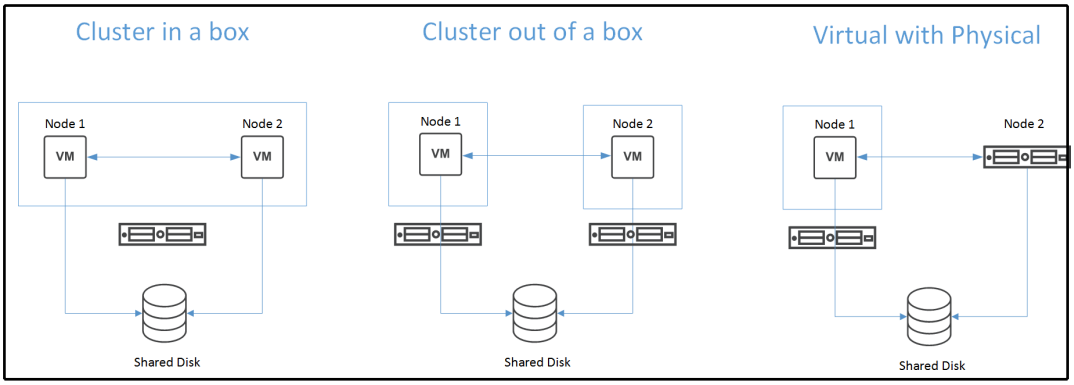
Clustering does not have such negatives, but configuration dramatically depends on the application.

VMware vSphere supports the following clustering options:

- **Cluster-in-a-Box**: This is when VMs are clustered on the same ESXi host. The shared disks or quorum (either local or remote) are shared between the VMs. CiB can be used in test or development scenarios. However, this solution does not protect in the event of hardware failure.
- **Cluster-out-of-the-Box**: The cluster is deployed to two VMs and the two VMs are running across two different ESXi hosts. This protects against both software and hardware failures. Physical RDMs are the recommended disk choice. Shared storage/quorum should be located on a fiber channel SAN or via an in-guest iSCSI initiator.

- **VM and physical server clustering**: This involves one cluster node running natively on a physical server, while the other runs as a VM. This mode can be used to migrate from a physical two-node deployment to a virtualized environment. Physical RDMs are the recommended disk option here. Shared storage/quorum should be located on a fiber channel SAN or through an in-guest iSCSI initiator.

Different cluster solutions are shown in the following diagram:



Clustering features available in VMware vSphere

Many features can be used for guest OS or application clustering within VMware vSphere. There are always pros and cons for each type depending on your use case:

- **SCSI bus sharing for virtual disks on VMFS volume**: You can enable simultaneous access to the single disk for multiple VMs
- **SCSI bus sharing for RDM devices**: In this case, there is no virtual disk located on the VMFS datastore, but the device is mapped using **Raw Device Mapping** (**RDM**) as a disk to the VMs
- **Multi-writer flag on the virtual disk**: No bus sharing is involved, but the disk is unlocked for simultaneous operations from the VMs
- **In-guest iSCSI**: There is no shared disk on the vSphere level, the disk is mapped from the guest OS.

Although VMware vSphere supports different storage configurations that can be used for clustering, always check the vendor or VMware knowledge base to check if your application supports such configuration. If something can be configured and works as expected, it does not mean that such a configuration is fully supported. For example, with **Microsoft Windows Server Failover Clustering** (**WSFC**), you can configure bus sharing for the disks located on the VMFS volume (no RDM), but it is supported only for CiB deployment, not CAB, although it works.

The following are the supported shared storage configurations:

Storage type	CiB	CAB	VM and physical
Virtual disks	Yes	No	No
Pass-through RDM (physical compatibility mode)	No	Yes	Yes
Non-pass-through RDM (virtual compatibility mode)	yes	No	No

Cluster-related configuration parameters are as follows:

- **SCSI Controller settings**:
  - **Disk types**: You have the choice of VMDK, virtual RDM (virtual compatibility mode), or physical RDM (physical compatibility mode).
  - **SCSI bus-sharing setting**: Virtual sharing policy, or physical sharing policy, or none.
- **SCSI bus sharing values**:
  - **None**: Used for disks that aren't shared in the cluster (between VMs) or when a multi-writer flag is used.
  - **Virtual**: Use this value for CiB deployments.
  - **Physical**: Recommended for CAB or physical and virtual deployments.
- **Raw Device Mapping (RDM) options**:
  - **Virtual compatibility mode**: In this situation, RDM acts identically to the virtual disk file, and you can use standard virtual disk benefits such as cloning or snapshots.
  - **Physical compatibility mode**: In this situation, RDM has direct access to the SCSI device. This mode is especially usefully for applications that need low-level control over the device.

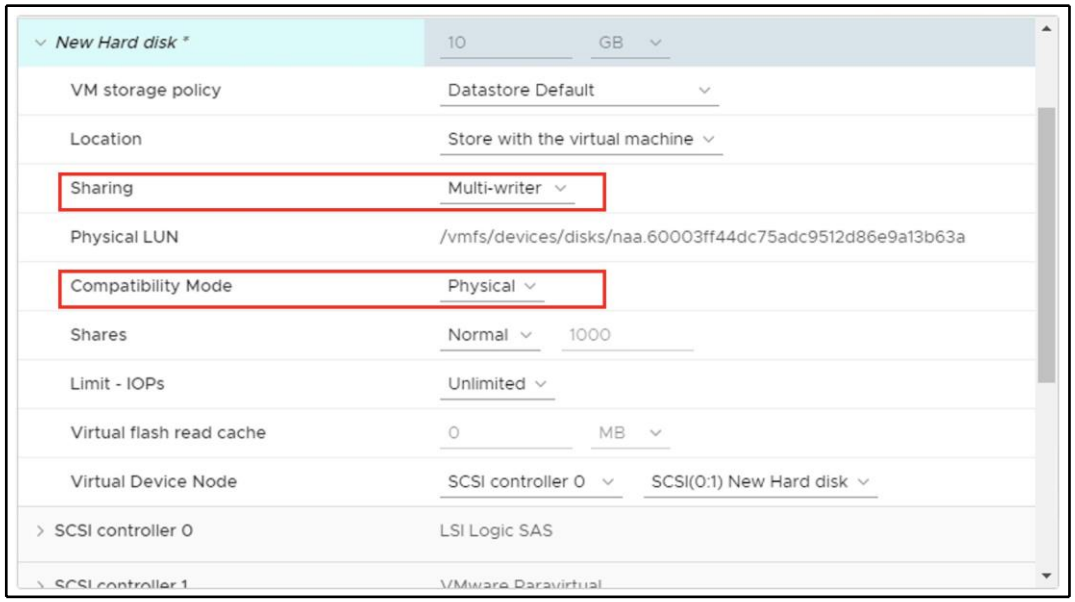
Please note that your operating system and application must be cluster-aware. You can only map the same disk to multiple VMs assuming both of them will have read/write access to the device.

When working with cluster configuration, always follow the official configuration guide of the vendor of the application or the operating system. Any misconfiguration can lead to serious cluster issues or event data corruption.

RDM device and multi-writer flag

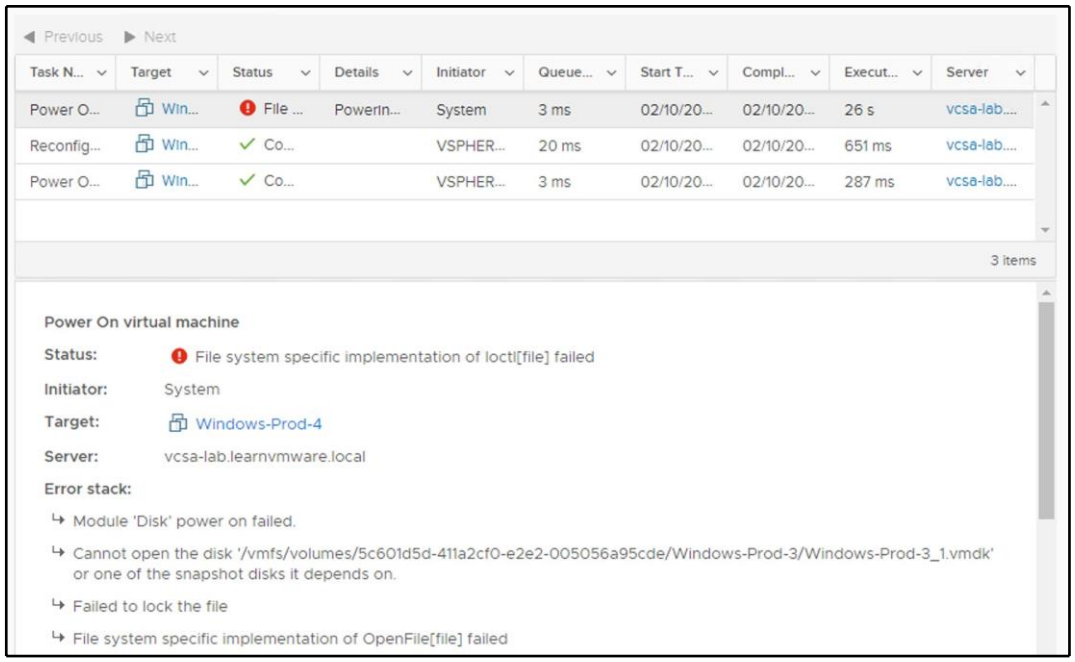
This is probably one of the most common clustered deployments, but always follows the official documentation from VMware or your application vendor for specific configurations. In this example, I assume that a VM is already configured and the operating system is installed. An initial VM has a single virtual disk on the VMFS datastore connected to SCSI controller 0:

1. Power off the VM and edit the settings of the virtual hardware.
2. Assign a new SCSI controller 1:
    1. **Type**: **VMware Paravirtual**
    2. **SCSI Bus Sharing**: **None**
3. Create a new RDM disk and attach it to SCSI controller 1:
    - **Compatibility Mode**: **Physical**
    - **Sharing**: **Multi-writer**:



4. Start the first VM.
5. From the guest OS (assuming it is Windows Server), rescan the storage devices.
6. Bring the disk online and create a new simple volume.
7. Power off the second VM that should be part of the cluster.
8. Assign a new SCSI controller 1:
    - **Type**: **VMware Paravirtual**
    - **SCSI Sharing**: **None**
9. Assign the disk created in step number 3 using **Add existing disk.**
10. Set the sharing policy to **Multi-writer.**
11. Start the second VM.
12. Rescan the storage devices, and you should see the shared disk.
13. Proceed with Oracle RAC/Microsoft Cluster Services deployment.

If you forget to enable the multi-writer flag, the following error will be displayed:



Before the Oracle or Microsoft cluster is configured, do not copy anything to the new disk from within the guest operating system.

You should also configure **DRS anti-affinity VM rules** to keep the VMs on the different ESXi hypervisor.

Working with clusters is not an easy task, and as already stated, any minor misconfiguration can lead to severe problems. From my perspective, I would recommend switching to scaled-out solutions instead of clustered ones because such deployments provide the same availability as clusters, but they do not involve any advanced configuration on the vSphere level. In-guest bunching arrangements that don't utilize a common plate design, for example, **SQL Mirroring, SQL Server AlwaysOn Availability Group,** and **Exchange Database Availability Group**, don't require unequivocal help explanations from VMware. These setups don't require extra VMware consideration with respect to a particular stockpiling convention or various hubs and can be conveyed on VMs in much the same way as on physical devices.

For more information, feel free to visit the following articles at `https://kb.vmware.com/s/article/1034165` for multi-writer information and `https://kb.vmware.com/s/article/2147661` for failover clustering guidelines.

Virtual machine backup

The choice of the suitable backup solution for an infrastructure depends on what you need to protect—configurations, data, VM, applications, or a complete system state.

Depending on that, backup solutions can be categorized as follows:

- **Backup with an agent**: This solution is intended for the protection of a physical environment or specific configurations such as a VM cluster. Vendors such as Arcserve and Veritas Backup Exec provide this solution type.
- **Native backup for VMware**: This was developed explicitly for virtualized environments, and is the recommended solution for a virtualized environment because it takes the benefits of vSphere features (for example, snapshots technology). Veeam, Nakivo, Altaro, Vembu, and HPE are some vendors that provide backup solutions for virtual infrastructure.
- **Hyper-scale backup:** Vendors such as Rubrik and Cohesity provide ready-to-use backup solutions based on appliances installed in the infrastructure.

Let's discuss the protection of infrastructure components and the tools used to guarantee maximum availability.

Transport modes

Software backup solutions use different protocols called transport modes to retrieve VM data from storage. The transport mode to use for backups depends on the design of the network and the storage architecture.

Four main transport modes are supported for handling data:

- **Network Block Device (NBD)**: The ESXi host reads data from the storage and sends it to the application, across the network, using the NBD protocol. This mode can be used in any infrastructure configuration and is the simplest method to implement.
- **Network Block Device Secure Sockets Layer (NBDSSL)**: This is the same as NBD but uses SSL to encrypt the data passed over the TCP/IP connection.
- **SCSI HotAdd**: This is a LAN-free data transfer mode where the `.vmdk` files of a VM are attached to the backup application. Data won't go through the network but is read and written directly from/to the datastore. In many environments, this is the preferred mode.
- **Direct SAN**: In this, data is read directly from the SAN or iSCSI LUN; this provides the fastest data transfer speed. Direct SAN transport mode is recommended if the VM's disks are stored on shared SAN LUNs connected to the ESXi host over FC, FCoE, and iSCSI.

Not all transport modes can be used in all cases: for example, with virtual volumes direct SAN mode is not supported.

Also, some specific backup products can implement other specific transport modes (for example, direct NFS with Veeam Backup and Replication).

Backup solutions for VMware vSphere

The market offers several valid solutions you can choose from, and backup product selection must consider different elements, such as infrastructure complexity, supported platforms, backup types, licensing, and budget.

Depending on what you need to back up, the software solution you choose must provide specific features to meet the requirements. For example, to back up Microsoft SQL Servers, application-aware backups with log truncate features must be supported to ensure database consistency. If you are still performing backups on tape, make sure the product supports tapes.

Since the available backup solutions provide different options, capabilities, and pricing, some popular backup products specific to virtual environments will now be briefly illustrated to show what the market is offering. The listed vendors and product order don't follow any classification or preference.

Veeam Backup and Replication

Veeam offers robust and powerful backup and replication features to protect entire virtual infrastructures. It's a backup solution for enterprises but also for SMBs.

It's installation supports Windows OSes only. The management of the application can be done through a console deployed on the administrator's computers or with a web-based console. Despite its simplicity, Veeam protects the infrastructure in a very robust and reliable way.

The main features of Veeam are as follows:

- **Backup**: Full VM backup, incremental backup, copy backup, cloud backup (AWS, Azure, Veeam Cloud Connect), tape backup, replication, cloud replication.
- **Restore**: Restore full VM, Instant File Recovery, Instant VM Recovery, Instant Object Recovery (AD, Exchange, Microsoft SQL, SharePoint, Oracle).
- **Licensing**: This is per physical CPU socket.
- **Available in three editions**: It is available in Standard, Enterprise, and Enterprise Plus. Veeam also provides an Essential version with an affordable price that is designed for small organizations with fewer than 250 employees and is limited to 6 CPU sockets. A Free Edition is available but is limited to full backups only, and vPower, VMs replication, and scripting features are not available.

NAKIVO Backup and Replication

NAKIVO Backup and Replication is a backup solution for SMBs and enterprises and can be deployed on both Windows and Linux OS or as a virtual appliance, allowing you to save some Windows licenses. Management is done through a simple, comfortable, and intuitive HTML5-based console that guides the user through the configuration steps required by the backup or restores procedures.

The installation and usage are, and this product offers all the features required by modern data centers. You need a few minutes to get the software up and running.

The main features of NAKIVO Backup and Replication are as follows:

- **Backup**: Full VM backup, incremental backup, copy backup, replication, cloud backup (AWS), cloud replication.
- **Restore**: Restore full VMs, Instant File Recovery, Instant Object Recovery (AD, Exchange), Instant VM Recovery.
- **Licensing**: This is per physical CPU socket.
- **Available in three editions**: This is available in Basic, Pro, and Enterprise editions. An Essential version is also available with an affordable price for the Pro and Enterprise Editions, designed for SMBs, and limited to a maximum of 6 socket licenses per organization. A Free Edition is available and supports up to two VMs.

Altaro VM Backup

Altaro VM Backup is a backup solution for SMBs deployed to the Windows platform. It offers all the features required by the Disaster Recovery(**DR**) to protect VMware virtual infrastructures. A dedicated Windows machine is not required, and there is no need for third-party software dependencies such as Microsoft SQL. The product is easy to use, with an intuitive design, and provides full control over backup jobs across all hosts.

You can manage the application from a console that is deployed on the administrator's computers and can be used as a central monitoring station for several Altaro VM instances.

The main features of Altaro VM Backup are as follows:

- **Backup**: Full VM backup, incremental backup, copy backup, cloud backup (Azure), replication.
- **Restore**: Restore full VMs, Instant File Recovery, Instant Object Recovery (Exchange), Instant VM Recovery.
- **Licensing**: This is per physical host with unlimited sockets/CPUs.
- **Available in three editions**: It is available in Standard, Pro, and Enterprise editions. A Free Edition is also available and supports protection for two VMs.

Vembu VMBackup

Vembu is a backup and DR software solution that can be deployed on the Windows and Linux platforms or as a virtual appliance. It is suitable for data centers and small and medium businesses with enterprise-level features. Backup copies of your backups can be sent to offsite storage or to Vembu Cloud, which provides data redundancy and DR.

The main features of Vembu VMBackup are as follows:

- **Backup**: Full VM backup, incremental backup, copy backup, cloud backup (Vembu Cloud), replication.
- **Restore**: Restore full VMs, Instant File Recovery, Instant VM Recovery, Instant Object Recovery (Microsoft Exchange, SharePoint, SQL, and AD).
- **Licensing**: This is per physical CPU socket.
- **Two editions available**: It is available in two editions—BDR Suite and Free. The Free Edition supports a maximum of three VMs.

Deduplication appliances

Data that needs protecting is continuously growing, and the available space on storage devices is never enough. To reduce the space occupied by backup files on storage devices, deduplication technology allows a reduction in storage space consumption.

Deduplicated storage should be used mainly as secondary targets due to their design. These storage systems are often developed to optimize write operations, but random read I/O performance may suffer.

Hyper-scale solutions

Hyper-scale is an architecture capable of scaling appropriately as increased demand is added to the system. This architecture is composed of individual servers, referred to as nodes, that provide resources in terms of compute, storage, and networking, and are put together in a cluster and managed as a single entity.

The advantage of hyper-scale is its architecture, which can be expanded as demand grows by merely adding new nodes to the cluster.

Cohesity

Cohesity DataPlatform is a hyper-converged platform solution that consolidates and manages secondary data with scale. Cohesity offers secondary storage devices with global deduplication, compression, and encryption.
The Cohesity appliance is installed with a proprietary OS and distributed filesystem. The design allows for scaling to any capacity.

When using a Cohesity solution, you have two main benefits:

- You can eliminate secondary storage silos and consolidate backups.
- You can control all your secondary data operations with converged data protection.

Rubrik

Rubrik is a cloud data management solution for protecting workloads that deliver a data management platform for enterprises in private, public, and hybrid cloud environments.

The Rubrik solution is deployed as an appliance to insert in the rack and power on. The scale-out hardware combined with robust backup software manages, with a single platform, all data in the cloud or on-premise for automated backup, DR, archival, and search, in a simple, scale-out platform built for the hybrid cloud.

VMware vSphere Replication

vSphere Replication is extension to vCenter Server, and it provides hypervisor VM replication and recovery. vSphere Replication is an alternative to storage-based replication and can be used together with the **Site Recovery Manager** (**SRM**) for more advanced scenarios.

vSphere Replication can provide very cost-efficient, simple, and powerful replication at the VM level, using scheduled asynchronous file-based replication. It is more cost-efficient because it reduces both storage costs and replication costs, and also because it's included in all editions, starting from the Essential Plus edition.

It is important to emphasize that vSphere Replication is based on asynchronous replication with a minimal recovery point objective of 5 minutes. Based on this technical limitation, you can't protect your site in real time, and you might even lose data from the past 5 minutes if any problems occur on the primary site.

One interesting aspect is that vSphere Replication does not use VM snapshots at. On the other hand, it is strictly dependent on the vCenter Server for its configuration and management.

vSphere Replication installation

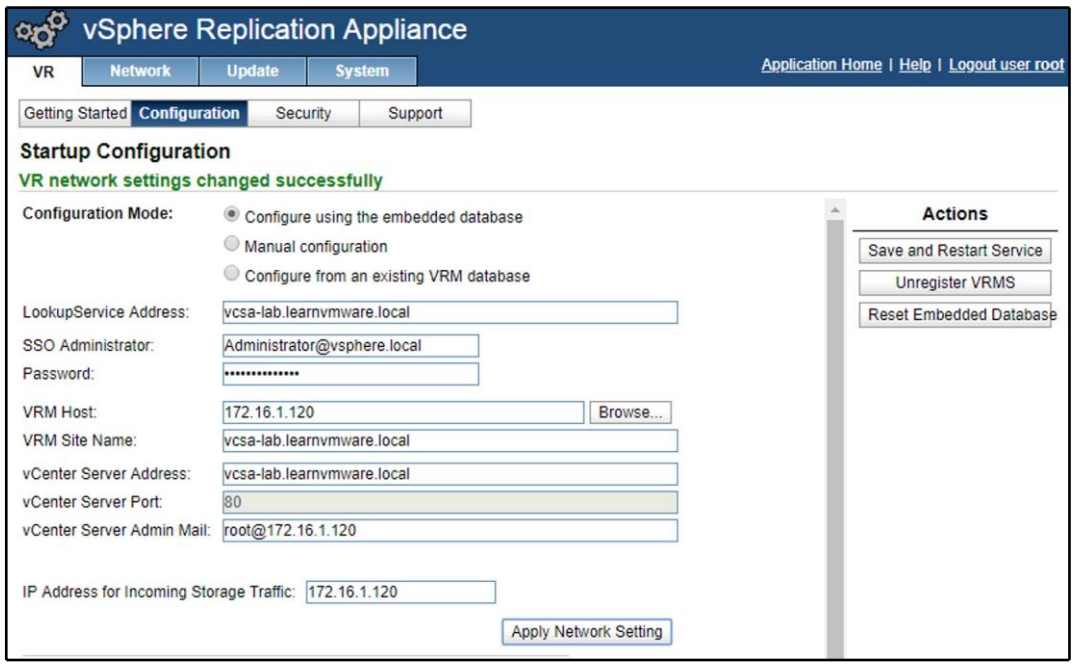
Technology that is used by vSphere Replication is already included in VMware vSphere, but to leverage it, you need to download the vSphere Replication appliance, which will hook to the vSphere APIs and provide the UI.

Once the appliance is deployed and configured, it provides a native vSphere plugin that will be available through your vCenter Server, leading to full integration with the vSphere Suite.

The initial installation and configuration are a straightforward process:

1. Download the ISO for the vSphere Replication appliance from the `my.vmware.com` portal.
2. Deploy the OVF template from the ISO file, as you already know how to do this (that is, select OVF and VMDK files, select the destination cluster and storage, map the VM to the correct port group, and customize the appliance).

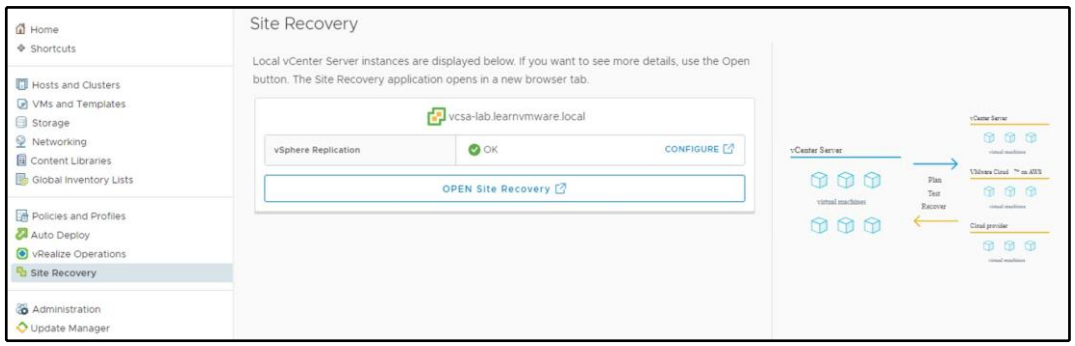
3. Once the vSphere Replication appliance is deployed, you need to link it to the vCenter Server. To do that, connect to the management interface of **vRealize Automation** (**vRA**) at `https://<replication appliance IP/FQDN>:5480`:



Once the binding is configured, you will be able to access vRA directly from the vSphere client.

Working with vSphere Replication

In past adaptations of VMware vSphere Replication, there was a different route called vSphere Replications. VMware has solidified this under **Site Recovery**:

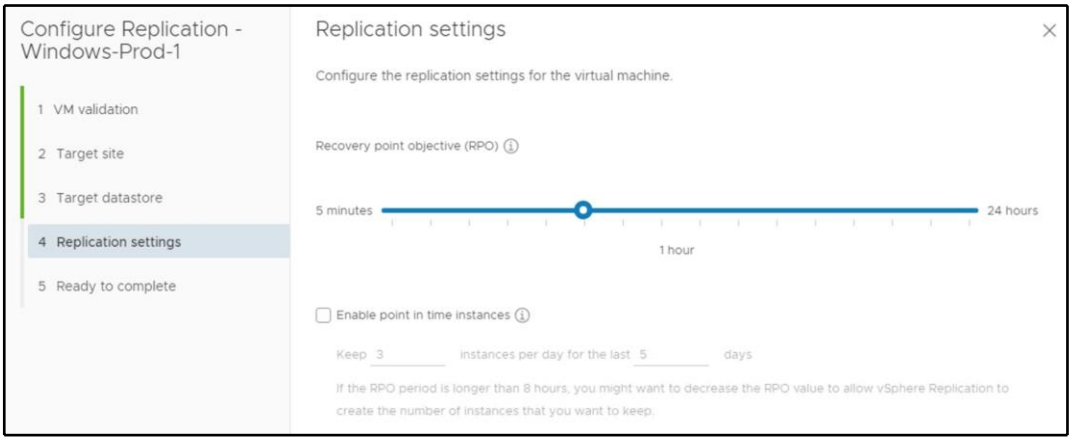


Configuring vSphere Replication

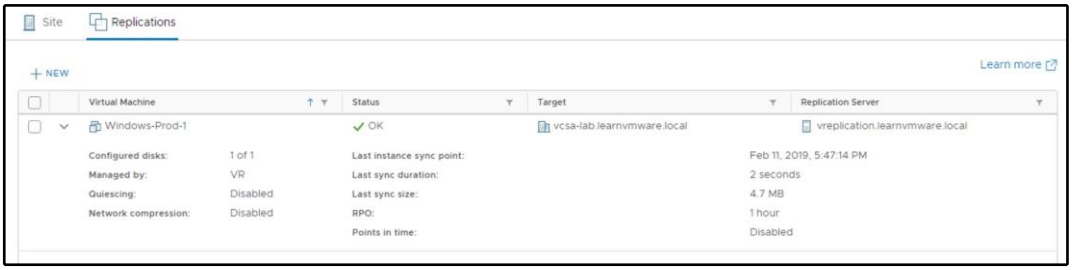
To configure vSphere Replication, follow these steps:

1. From the vSphere client, select the VM you need to replicate and select **All Site Recovery Actions** and **Configure Replication**.
2. The new UI will be launched.
3. First, the wizard will validate whether the VM can be configured for replication.
4. In the next step, you can select the replication target. As a replication target, you can use either the same vCenter server (the VM will be replicated to another local VM) or a remote vCenter server (the VM will be replicated to another site).
5. In the target datastore, you have to select on which datastore the data will be replicated.

6. The last step is to configure your replication option, especially **Recovery Point Objective** (**RPO**) and point in time snapshots, should you wish to use them:



Once the replication is configured, you can access the vSphere Replication UI to see the status of all replicated VMs and their last synchronization, and you have an option to recover from the replica:



Disaster recovery and disaster avoidance

A **disaster** is any event that halts business activity on a large scale. In most cases, we are talking about natural disasters, but there are also human-made disasters, and all of them can happen at any time without warning.

These disasters could impact technologies and IT services. Of course, there are other and more critical aspects, such as risk to human life, but for **Business Continuity**(**BC**), we put the main focus on business-critical services and applications.

DR provides BC in the event of a disaster, and may be just localized on equipment (such as a single server) or globally on an entire site. Business will be recovered by following a specific DR plan; it is just a subset of the **Business Continuity Plan** (**BCP**). In the case of a disaster that impacts an entire site or region, usually, the recovery process uses a remote location called a disaster recovery site.

DR is essential to ensure the continuation of business after a disaster. DR can also be required in several regulatory compliances. Effective DR is a critical part of a BCP must address the following three organizational requirements:

- **Minimize risks**: Having a BC plan does not eliminate all risks if you cannot be sure that the plan is reliable or practicable. The DR plan could be difficult to implement, and for several organizations, it may have some business impacts and possible risks.
- **Minimize downtime:** The consequences of extended downtime can be critical for business, recognition, and productivity. For most companies, a service disruption of ten or more days could be a total disaster and lead to the company closing.
- **Control costs:** Traditional disaster recovery plans are often limited in scope because of the cost, but you must find a trade off between costs and risk mitigation.

Although everyone realizes the importance of a DR plan, some organizations do not have the proper level of DR protection that they need. Only after a real disaster do they fully understand the importance of DR and the real impact that a disaster can have on their business.

Legacy solutions and processes to activate applications in the DR site usually require complex runbooks and manual procedures to execute the failover process. They may require highly specialized staff with vertical skills, large time investments, and high levels of coordination from several teams that are responsible for different layers of the infrastructure.

The main challenges that must be handled by DR in order to have a successful and effective plan are as follows:

- **Complexity**: Usually data center recovery plans are complex processes because, to guarantee the correct recovery of entire business services, they must deal with all the inter-dependencies between applications, hosts, networks, storage, and other infrastructural and organizational aspects.

- **Lack of predictable and reliable recovery**: Recovery plans documented in run books can be incomplete and may quickly fall out of sync with rapidly evolving deployments. Most enterprises test their recovery plan only twice a year or less.
- **High cost**: Legacy DR solutions require significant capital and operating expenditures. The DR site typically requires a dedicated duplicate server infrastructure. As defined by Gartner in *Survey Analysis: IT Disaster Recovery Management Spending and Testing Activities Expand in 2012, July 2012:*

> *"The net result is that legacy disaster recovery solutions are regarded as non-strategic and costly insurance policies with very questionable returns. At best, only a few mission-critical applications get the privilege of site-level protection."*

Two of the fundamental parameters that characterize a BC/DR plan are the **Recovery Point Objective** (**RPO**) and **Recovery Time Objective** (**RTO**):

- RPO refers to how much data you can lose. RPO usually refers to the time that has elapsed between last backup and the failure.
- RTO refers to how long it will take to recover from the failure. Usually, the duration of the restore procedure is referred as Recovery Time Objective.

In a perfect world, you would want to have those numbers near zero, but the cost of such a solution would be extremely high. In the real world, you need to find a balance between the cost of a BC/DR solution versus the potential risks of failure.

DR of a virtual data center

Protecting a virtual workload is much easier compared to protecting a physical data center, in a legacy way, for several reasons:

- **Virtualization provides encapsulation**: A VM is usually a set of files that can be easily managed.
- **Virtualization provides hardware independence**: Where a VM can run as it is (without any changes) on different hardware, maybe also with different sizing, but using the same hypervisor (or a compatible one).
- **Different types of data replication can be used**: From traditional storage array replication to VM replication.

- **On the DR site, you can, potentially, and for a limited number of workloads**: Use a single server that has enough computing and storage capacity.
- **Cloud Disaster Recovery as a Service (DRaaS)**: This solution is possible and practical.

All of the preceding benefits make it possible and convenient to provide a low RPO and a low RTO for the entire data center (or a large set of it), not just for the first tier of business-critical applications and services. Also, we have to consider that most companies are already leveraging virtualization, with a virtualization-first approach, also dictated for BC and DR requirements.

DR versus disaster avoidance

Disaster avoidance, as the name suggests, is a way to avoid or lessen the likelihood that a catastrophe will happen (through human blunders), or guarantee that, on the off-chance that, such a disaster occurs, the impact upon the association's innovation frameworks will be limited as far as possible.

The idea of disaster avoidance provides better resilience rather than good recovery, but to use it, you cannot rely only on infrastructure availability solutions, which are mostly geographically limited to a specific site; you also need to look at how to provide better application availability and redundancy in the wake of foreseeable disruptions.

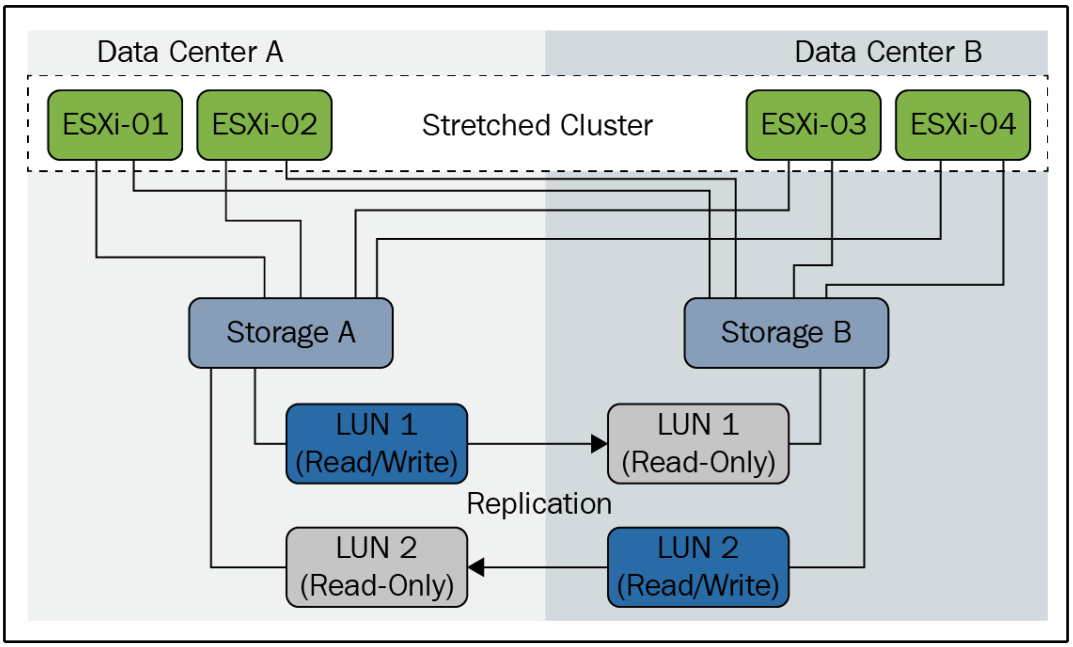
Multi-datacenter (or multi-region cloud) replication is one part of the solution. The second part is having active-active data centers or having applications spread between multiple sites that provide service availability.

Most new cloud-native applications are designed for this scenario. However, there are also some examples of traditional applications with HA concepts at the application level that can also work geographically, such as **DNS services**, **Active Directory Domain Controller** (**AD DC**), Exchange **database availability group** (**DAG**), and **SQL AlwaysOn** clusters. In all these cases, one system can fail, but the service will not be affected because another node will provide it. Although solutions such as Exchange DAG or SQL AlwaysOn rely on internal cluster services, applications designed with HA solutions usually use loosely coupled systems without shared components (except, of course, the network, but it can be a routed or geographical network).

An interesting examples of the infrastructure layer is a stretched cluster, or metro cluster.

DR versus stretched clusters

A stretched cluster, sometimes called a **metro cluster** or **metro storage cluster**, is a deployment model in which two or more host servers are part of the same logical cluster but are located in separate geographical locations, usually two sites. In a stretched cluster, the two groups of servers (in each site) are usually used to provide HA and load balancing features and capabilities:



This allows proactive behavior in order to avoid or minimize service outages, using disaster avoidance; if a disaster affects an entire site, the second one will manage all the resources and services. Although a stretched cluster can be used for disaster recovery and not only for disaster avoidance, there are some possible limitations on using a stretched cluster as DR as well:

- A stretched cluster can't protect you from site link failures and can be affected by the split-brain scenario.
- A stretched cluster usually works with synchronous replication; that means a limited distance, but also makes it difficult to provide multiple restore points with different timings.
- Bandwidth requirements are high, to minimize storage latency. So you not only need reliable lines, but also enough capacity.
- A stretched cluster can be more costly than a DR solution, but of course, can also provide disaster avoidance in some cases.

In most cases where a stretched cluster is used, there might be a third site acting as a traditional DR; in this way, a multi-level protection approach is used.

VMware solutions

In the past, business continuity was the first driver of virtualization; virtualization not only helps with server consolidation and driving down costs across IT organizations, but can also improve availability, resilience, and recoverability for business-critical applications and services.

VMware provides a holistic approach to protecting your IT environment and all applications running on the vSphere platform from a variety of factors that can cause application downtime, including unplanned events such as server failures and even planned events such as server maintenance. These solutions provide simple, cost-effective protection with a standard solution for all your applications and services.

VMware BC-related solutions cover the following:

- **Local availability**: Some products and technologies protect applications against the downtime of individual hosts. This includes vSphere HA and FT for unplanned downtime, as well as vMotion and storage vMotion for planned downtime.

- **Data protection**: There are solutions to back up entire VMs, including OSes, application binaries, and application data, in a simple non-disruptive manner. A third-party solution can use VMware storage APIs for data protection to enable native VM backup.
- **Disaster recovery**: vSphere Replication is an exciting addition to the vSphere platform, providing a cost-efficient and straightforward way to implement VM-based replication. For DR orchestration, vCenter SRM leverages vSphere and vSphere Replication (or storage-based replication) to protect applications against site failures and to streamline planned migrations.
- **Disaster avoidance**: **vSphere Metro Storage Cluster** (**vMSC**) is a configuration option, introduced in vSphere 5, that allows the use of stretched clusters.

You can find more details in **Mastering Disaster Recovery: Business Continuity and Virtualization Best Practices** at `http://download3.vmware.com/elq/img/EMEA/EMEA11122/pdf/VirtMngt_MasteringDisasterRecovery_WhitePaper_Q410_EN.pdf?cid=70180000000wCtz`.

VM Replication

VMware vSphere Replication is a BC/DR solution that enables you to replicate VMs to the same vCenter Server or even to a different vCenter Server running on another site. You can also utilize third-party service providers who offer disaster recovery to cloud solutions where the primary VM is replicated to the service provider environment.

vSphere Replication is a available for both vCenter Server for Windows and for vCenter Server Appliance, and it is included in every license starting from Essentials Plus.

Once you enable the replication, any VM can be replicated to the other site. If it is required (for example, because of low bandwidth between sites), the initial replica can be transported using offline media, and once it is at the targeted site, the delta data is transferred over the network. Only the changed blocks are transferred between sites using the **Change Block Tracking** (**CBT**) feature of vSphere.

You have the option defining your own RPO ranging from 5 minute to 24 hours depending on your needs, and you are able to use **multiple points in time** (**MPIT**), in which case you will be able to store up to 24 snapshots of each replicated VM.

Of course, this is not the only VM-based replication solution; several third-party products provide more features and capabilities. Most native backup products also have VM Replication features, and so, in this case, they could be cost-effective solutions, especially if you already have that specific backup product.

There are also some replication products that can replicate VMs not only across the same virtualization platform but also across different types of hypervisor such as **Zerto**. You can find more information about the technology at `https://www.zerto.com/solutions/use-cases/cross-hypervisor-replication-vmware-hyper-v-aws/`

Stretched cluster

A VMware vMSC configuration is a certified solution that combines vSphere clustering with array-based replication. Such solutions are commonly deployed in environments where the distance between data centers is limited since array-based replication requires low latency between sites.

vMSC infrastructures are implemented with the goal of reaping the same benefits that HA clusters provide to a local site, in a geographically dispersed model with two data centers in different locations.

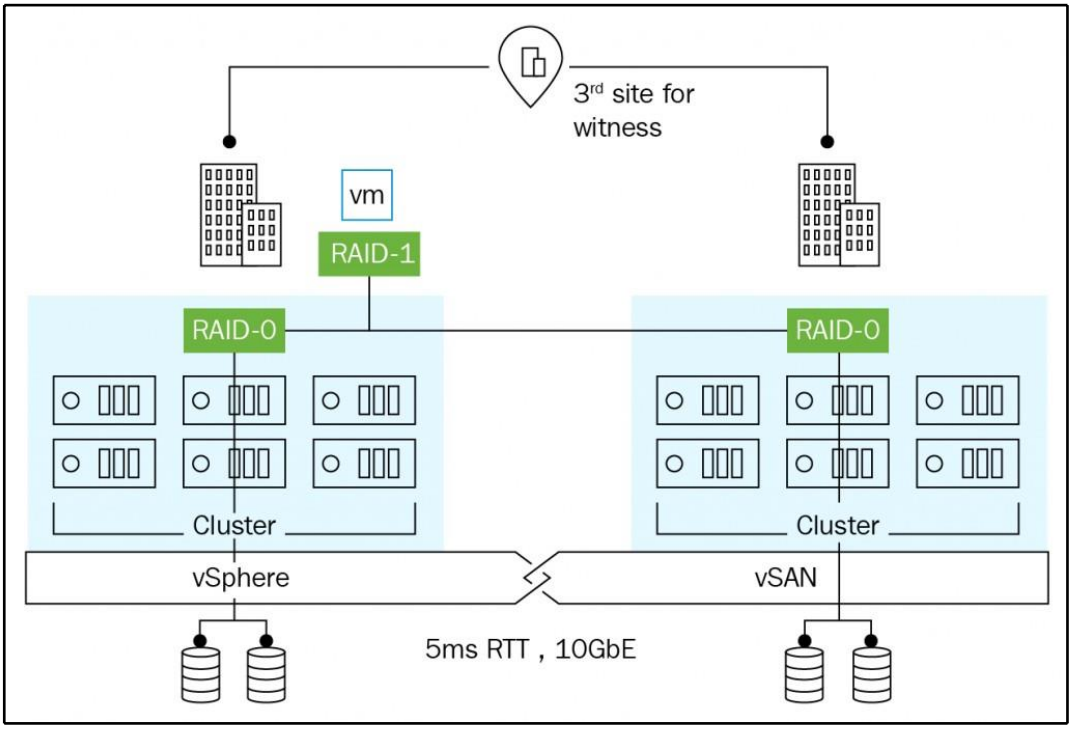
A vMSC infrastructure is necessarily a **stretched cluster**. The architecture is built on the premise of extending what is defined as *local* in terms of network and storage to enable these subsystems to span regions, resulting in a single logical infrastructure consisting of the resources from both of the sites.

VMware vMSC is just a configuration option for a vSphere cluster, where half of the virtualization hosts are on one site and the other half is on a second site. Both sites work in an active-active way, and common vSphere features such as vMotion and vSphere HA can be used.

The only restrictions are that vMotion must support higher latency (this is possible in the Enterprise Plus Edition), that all VMs must reside on the same layer 2 networks (that means a stretched network is needed, or some other network virtualization technique), and that the storage part can provide active-active access from both sites (there are several types of storage certified for vMSC).

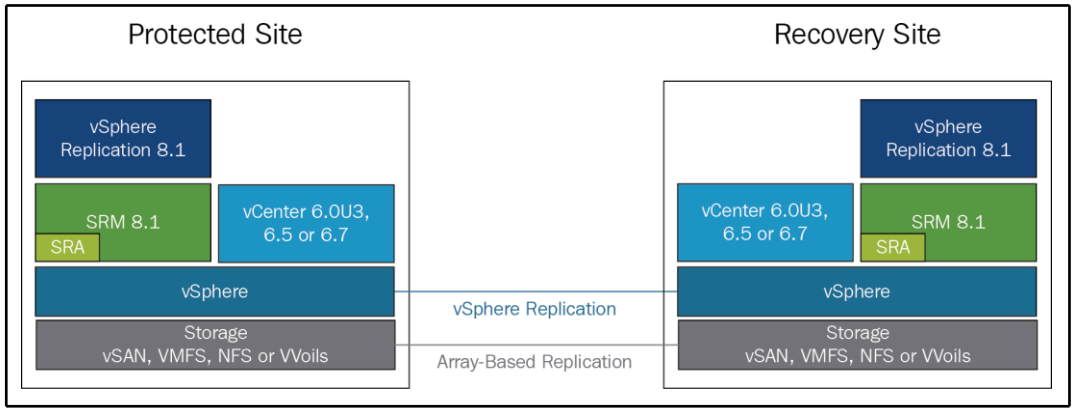
On account of a site catastrophe, vSphere HA will give start the VMs on the other site.

VMware vSAN in the Enterprise Edition can also provide the stretched shared storage part; in this way, it's possible to build a complete stretched cluster using vSphere as a core infrastructure component for both compute and storage components:



SRM

VMware's disaster recovery solution is called SRM. SRM is mechanization programming that incorporates a hidden replication innovation to give strategy-based administration, non-problematic testing, and computerized coordination of recuperation designs. It gives accessibility to the VMs when disasters occur:



SRM uses vSphere Replication and supports a broad range of array-based replication solutions available from various VMware storage partners. SRM integrates tightly with VMware vSphere Web Client and vCenter Server, and it also takes advantage of **software-defined datacenter** (**SDDC**) by integrating NSX and vSAN.

As written, cross-site replication at the VM or storage level is just the first step for DR. You will then need an entire set of rules to define how to recover your VMs in the case of the primary site failure. SRM delivers several features and functions, including centralized recovery plans, non-disruptive testing, and automated orchestration, both for fail-over (in the case of DR) and fail-back (in case the original site has been recovered).

Adding extended stockpiling to a SRM sending on a very basic level diminishes recuperation times; on account of a catastrophe, recuperation is a lot quicker because the extended stockpiling design that empowers synchronous information composes and peruses on the two destinations.

Please note that the SRM model for active-active data centers is fundamentally different from the model used in the VMware vMSC. SRM uses two vCenter Server instances, one on each site, instead of stretching the vSphere cluster across sites.

Adding SRM to a stretched storage deployment allows users to benefit from key features of SRM that are not present in vMSC, such as centralized recovery plans, orchestrated recovery, and non-disruptive automated testing.

16
Securing and Protecting Your Environment

One of the pillars of virtualization is the VM isolation property, which can protect the host layer from the VM effectively. Although some possible attacks have been found, virtualization remains an exciting approach to improve the security of your infrastructure. Securing and hardening your vSphere infrastructure should be considered one of the most important steps toward making your infrastructure as reliable as possible.
A new trend is now also to protect VMs from the underlying infrastructure; for example, in the case of a public cloud service, consumers may have some concerns about the security and privacy of their data. VMware offers different encryption mechanisms that make your data private no matter where they are being run.

This chapter will cover the following topics:

- Tuning and hardening guidelines
- vCenter and ESXi security
- Working with encryption and securing VMs

Security and hardening concepts in vSphere

Security is a complete process flow with an entire life cycle; depending on the model that will be used, the first part of the process is usually product-agnostic, but there is a part that's dependent on the different products and their features and capabilities.

Following VMware's vision, the five pillars of cyber hygiene are as follows:

- **Least privilege**: This is the standard and most reasonable approach, which applies to user accounts, service accounts, and services in general (for example, used ports).
- **Micro-segmentation**: Using NSX, it's finally possible to bring network control at the VM level with granular security rules. Considering also the new product, VMware AppDefense, VM security can be enforced at both network and application levels.
- **Encryption**: Data must be protected at each level, and for the physical level, encryption is the only way to ensure proper protection. We will discuss this later in the chapter.
- **Multi-factor authentication (MFA)**: Authentication is usually the weakest part, mostly due to passwords that are too simple (or passwords that are not changed periodically). We will discuss this later in the chapter.
- **Patching**: Keeping your software components up to date is crucial for the security aspect, but it's also essential for implementing new features. We have discussed this in *Chapter 12*, *Life Cycle Management, Patching, and Upgrading*.

Hardening vSphere

Hardening is the process of securing a system, a service, or an entire infrastructure, by reducing the attack surface and minimizing the possible vulnerabilities. VMware has built **Security Hardening Guides**, which can found at `https://www.vmware.com/security/hardening-guides.html`, to provide prescriptive guidance for customers on how to deploy and operate VMware products in a secure manner.

The vSphere 6.7 Security Configuration Guide is a spreadsheet file with several guidelines classified with a risk profile, useful as a checklist for tuning, with rich metadata for guideline classification and risk assessment. There are also some example scripts for enabling security automation. For more information on how to read them and how the guidelines have changed since the previous release, see `https://blogs.vmware.com/vsphere/2018/11/announcing-the-vsphere-6-7-update-1-security-configuration-guide.html`.

The vSphere 6.7 Security Configuration Guide isn't a compliance tool; it can be used for compliance, but it's not automatically enforced. It's mostly a set of guidelines that attempt to explain security risks. Also, the guidelines may or may not apply to specific customer cases.

Authentication and identity

The vCenter **Single Sign-On** (**SSO**) authenticates a user against the identity source (configured in the vCenter SSO). Identity sources define how and where to verify user credentials. vSphere supports several identity source types:

- **Local SSO domain**: Default SSO domain created during the installation of the PSC. This is a default identity source.
- **Active directory (native)**: When the PSC is joined to an AD domain, it is possible to use the domain or the forest as an authentication source using Kerberos authentication.
- **LDAP (active directory)**: Use this if you don't want to join the PSC to the AD domain, or if you are using a lightweight active directory.
- **LDAP (OpenLDAP)**: Use this if you have an open source LDAP server (such as OpenLDAP).
- **Local OS**: The user defined in the SAM (for a Windows-based PSC) or the `/etc/passwd` and `/etc/shadow` file (for a Linux-based PSC).

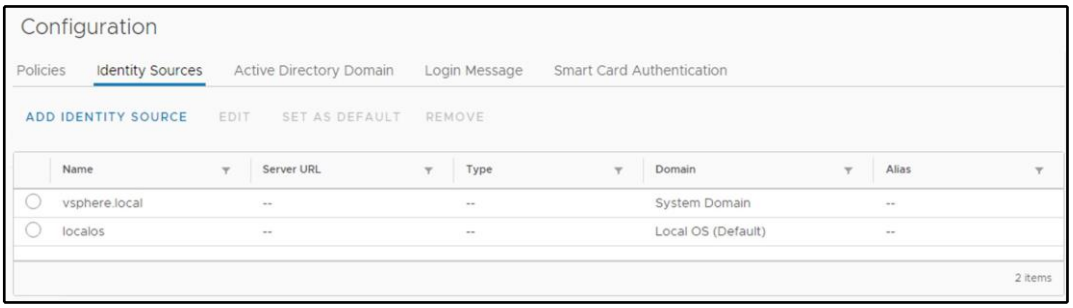
You can define as many identity sources as you need, but only one of them can be a default identity source. When an identity source is set as the default, you do not need to include the domain name as a part of the username (`user@mydomain.com`) as the domain name will be appended automatically, so all you need to do is to provide a username without a domain suffix.

SSO configuration

By default, only the administrator of the local SSO domain has permission to access a vSphere SSO configuration:

1. From the web client, select **Administration**
2. Under SSO, select **Configuration**

3. Switch to the **Identity Sources** tab, as shown in the following screenshot:



Password management

As vSphere is one of the most critical infrastructure components, it is critical to use strong passwords that are not easily guessed, and that is difficult for password generators. Password strength and complexity rules apply to all passwords, including hosts users (such as root).

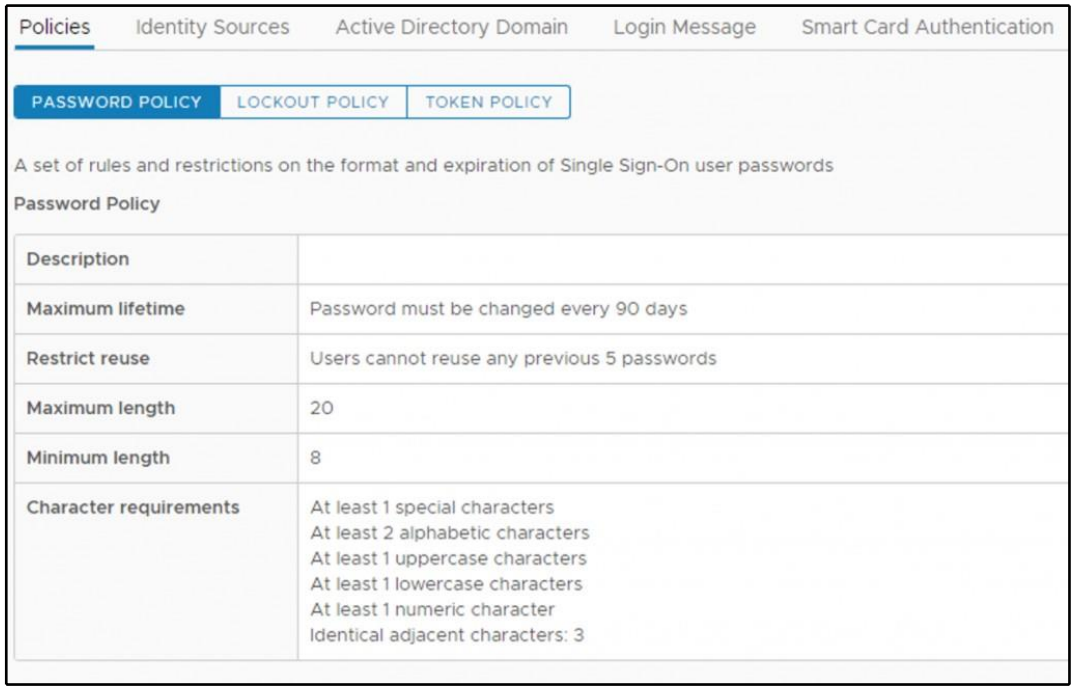
ESXi uses the `pam_passwdqc.so` plugin to set the strength and the complexity of the host's passwords. You can define the password quality using the host's advanced system settings, called `Security.PasswordQualityControl`.

ESXi 6.0 has presented another lockout feature; a limit of 10 failed logins is permitted before the account is locked. The account is unlocked after two minutes by default. In the host's events, you will see the following row:

```
Remote access to ESXi local user account 'LOGINNAME' has been locked
for 120 seconds after ### failed login attempts.
```

Account locking works for access through SSH and the vSphere web services SDK. It does not apply to the DCUI and ESXi Shell.

If you are using a local SSO domain, you can enforce password and lockout policies to force the users to use complex passwords with specific requirements:



If you are using AD users, both for hosts and vCenter, then the password policies are enforced by the AD GPO.

There are also password expiration rules for the virtual appliance local users, in case you are using vCSA for vCenter and/or the PSC components. Be sure also to check those settings:

1. Log in to the VAMI interface of the vCSA
2. Select the **Administration** tab from the navigator

3. Configure the desired options related to the password policy, as shown in the following screenshot:

Role-Based Access Control (RBAC)

The RBAC approach aims to limit individual users permissions based on their assignment within the organization by mapping the user accounts to the groups that are defined by the organization.

In VMware vSphere, you can find three core components of RBAC:

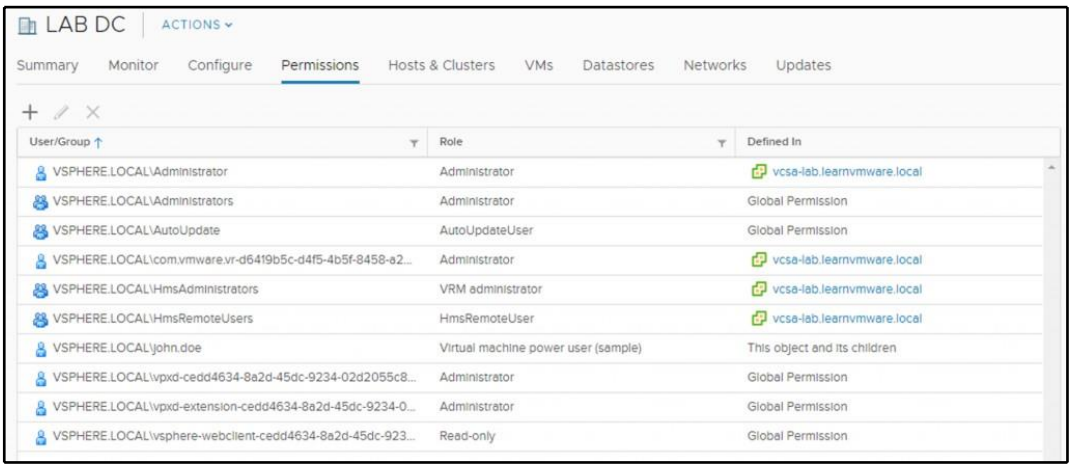
- **Roles**: A role is a specific subset of privileges based on the user's assignment.
- **Permissions**: Each task that can be performed is connected to a specific permission. To power off a VM, you need to have permission. Several permissions form a role.
- **Users and groups**: A role is mapped to a user and a particular vSphere object (such as a data center, cluster, or single VM).

Permissions are assigned according to the RBAC model, where you are matching the object (if the permission is not global), the user, or the group with the right role. A role is just a set of permissions, and you can use predefined roles or build (or copy) new ones.

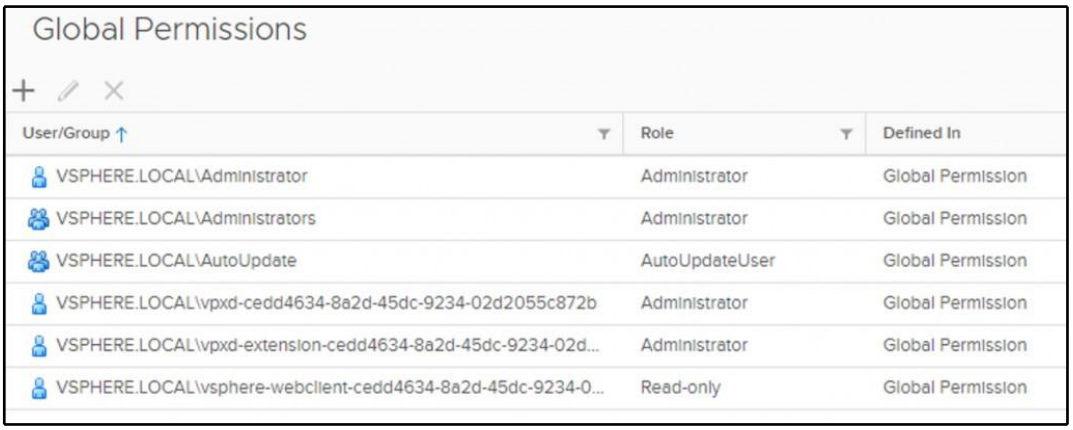
Try to avoid certain mapping roles to specific users. Always map a role to the group instead and add a user to the group.

Users and groups could be used to define vSphere permissions at two different levels:

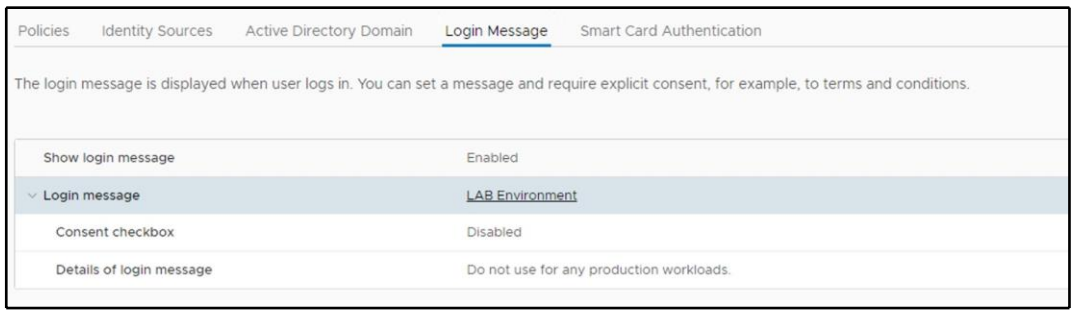
- **Inventory object level**: This is the traditional way to add permissions, by matching a user or group with a specific vSphere role on a specific object. It is useful if you need to delegate some management tasks:



- **Global level:** This is an option where you can define global permissions (with a specific role) on the entire infrastructure (at the PSC level, thus this global permission will be applied to all vCenter Servers linked to the PSC):



Starting with vSphere 6.0 Update 2, you can include a login banner with your environment. You can enable and disable the login banner from the SSO configuration, and you can require that users click an explicit **Consent checkbox**:



Active directory integration

The vCenter Server has an internal user database that allows you to add and manage users with the vSphere Web Client. User management and SSO is provided by the PSC, which has been available since vSphere 6.0. In a large environment, you might want to connect your virtualization infrastructure to a centrally managed AD.

MFA

MFA grants user access only after successfully presenting several separate pieces of evidence to an authentication mechanism, usually at least two of the following categories—knowledge (something they know), possession (something they have), and inherence (something they are).

Two-factor authentication (**2FA**) is a type of MFA where just two components are used. Starting with vSphere 6.0 Update 2, it is possible to have 2FA using the following:

- **Smart cards** (UPN-based **Common Access Card** (**CAC**))
- **RSA SecurID** token

vCenter SSO supports only native SecurID and does not support **Remote Authentication Dial-In User Service** (**RADIUS**).

Smart cards

A smart card is a small plastic card with an embedded integrated circuit chip that can be read by a smart card reader (many laptops may have one integrated). To enable smart card authentication for vCenter authentication, you must first set up your clients before users can log in using a smart card:

- **With vSphere 6.0**: Verify that the Client Integration Plugin is installed.
- **With vSphere 6.5 and 6.7**: Verify that the Enhanced Authentication Plugin is installed.

Then the configuration of the PSC is a little different in versions 6.0 and 6.5. For the latest version, before you can enable smart card authentication, you must correctly configure the reverse proxy from the command line on the PSC (or the vCenter if you have an embedded deployment). You have to create a trusted client **Certificate Authority** (**CA**) store that contains the trusted issuing CA's certificates for the client certificate.

For a Linux-based PSC, these are the possible commands:

```
cd /usr/lib/vmware-sso/
openssl x509 -inform PEM -in xyzCompanySmartCardSigningCA.cer >>
/usr/lib/vmware-sso/vmware-sts/conf/clienttrustCA.pem
```

Then you have to modify the `config.xml` file with the following changes:

```
<http>
 <maxConnections> 2048 </maxConnections>
 <requestClientCertificate>true</requestClientCertificate>
 <clientCertificateMaxSize>4096</clientCertificateMaxSize>
 <clientCAListFile>/usr/lib/vmware-sso/vmware-
sts/conf/clienttrustCA.pem</clientCAListFile>
 </http>
```

And finally, restart the service:

```
/usr/lib/vmware-vmon/vmon-cli --restart rhttpproxy
```

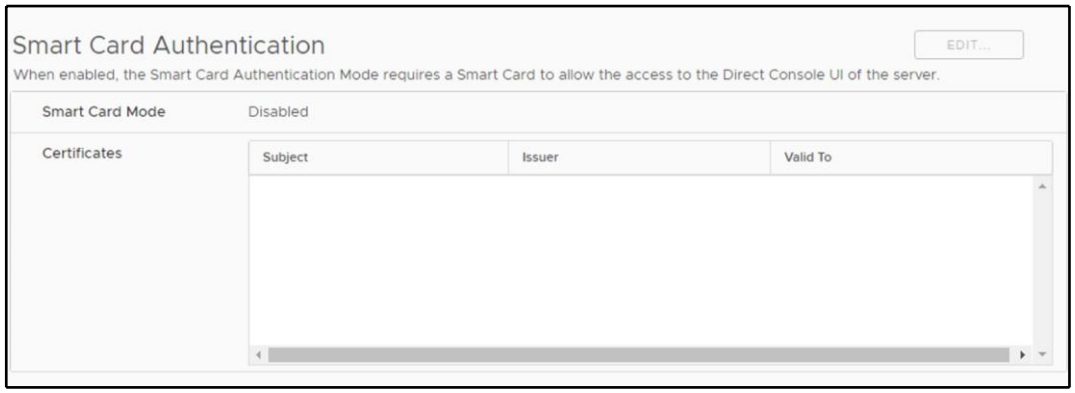
Then verify that an enterprise **Public Key Infrastructure** (**PKI**) is set up in your environment and that certificates meet the following requirements:

- A **User Principal Name** (**UPN**) must correspond to an AD account in the **Subject Alternative Name** (**SAN**) extension
- The certificate must specify client authentication in the **Application Policies** or **Enhanced Key Usage** fields, or the browser does not show the certificate

At this point, you can enable smart card authentication from the SSO configuration menu.

Starting with ESXi 6.0, it's also possible to use smart card authentication to log in to the ESXi DCUI by using a **Personal Identity Verification** (**PIV**), CAC, or SC650 smart card instead of specifying a username and password.

Under **Configure** | **System**, select the authentication services (described before for AD authentication) and you will see the current **Smart Card Authentication** status and a list of imported certificates:



In the **Smart Card Authentication** panel, you can click the **EDIT...** button and select the **Certificates** page to add trusted CA certificates, for example, root and intermediary CA certificates.

RSA SecurID

SecurID setup is supported only from the command line on vCenter Server version 6.0 or later. The configuration is well explained in the following blog post: `https://blogs.vmware.com/vsphere/2017/07/using-vcenter-login-banner-rsa-securid-support.html`.

Then the integration is quite simple on the web client authentication page:

RSA authentication manager requires that the user ID is a unique identifier that uses 1 to 255 ASCII characters. The characters ampersand (&), percent (%), higher than (>), less than (<), and single quote (`) are not allowed. Also, the RSA SecurID agent that is integrated into the PSC component of vCenter does not support PIN resets.

vCenter Server, ESXi, and VM hardening

VMware vSphere environments are sometimes deployed using the default configuration of many features and services and they are not regularly checked for potential improvements in terms of VMware security standards and best practices.

VMware regularly updates its hardening guides available at `https://www.vmware.com/security/hardening-guides.html` which provides essential information and recommendations on how to make the vSphere infrastructure more secure.

We have tried to pinpoint several of the most essential aspects of the vSphere infrastructure hardening in the following sections.

ESXi hardening

To protect the ESXi hosts against unauthorized intrusion and misuse, consider the following options for improving infrastructure security:

- **Limit user access:** This is done by restricting user access to the management interface and enforcing access security policies such as setting up password restrictions. Lockdown mode could be used to limit access to the hosts to all users. Otherwise, a centralized authentication could be useful to manage security groups and related roles (for example, with AD).
- **Limit shell access:** ESXi Shell (locally, but also through ESXi SSH access) has several privileged accesses to certain parts of the host. Therefore, they provide only trusted users with ESXi Shell login access. Usually, it is safe to keep both ESXi Shell and SSH access disabled to prevent direct access to the ESXi CLI. In this case, you can still use `esxcli` remotely or another remote CLI.
- **Limit services**: You can run ESXi essential services only. Some hardware vendors have specific agents that can run on ESXi hosts, but check their support and security level before running any third-party agents or services on ESXi hosts.
- **Limit network connections:** ESXi has a personal firewall (starting from ESXi 5.0) and, by default, is closed on most ports. When you enable a service, it also opens the right ports. Although you can manually open ports with the predefined firewall rules, and you can also build new custom ESXi firewall rules, it would be better to try to keep the ESXi firewall rules management entirely automatic. The personal firewall does not protect you from **Denial-of-Service** (**DoS**) attacks, so still keep your ESXi VMkernel interfaces on protected networks and still use perimeter firewalls.
- **Use secure connections:** By default, weak ciphers are disabled, and SSL secures all communication from clients. The exact algorithms used for securing the channel depend on the SSL handshake. VMware vSphere 6.0 introduces a certification authority to help in certification management. Starting with vSphere 6.5, the **Transport Layer Security** (**TLS**) protocol versions 1.0, 1.1, and 1.2 are enabled by default.
- **Patch your hosts:** Use only VMware sources to upgrade or patch ESXi hosts. VMware does not support upgrading these packages from any source other than a VMware source.

- **Check VMware Security Center**: VMware monitors all security alerts that could affect ESXi security and, if necessary, issues a security patch. If you regularly check the VMware Security Center site, you can find any alerts that might impact the environment.

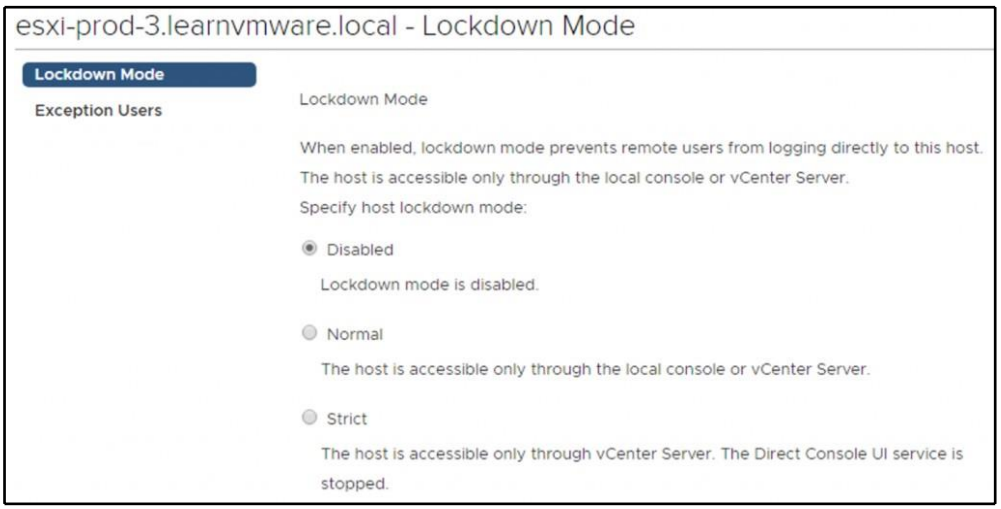
You can check the official guide, **General ESXi Security Recommendations**, at `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.security.doc/GUID-B39474AF-6778-499A-B8AB-E973BE6D4899.html`.

Lockdown mode

When you connect ESXi to vCenter, to increase the host's security, you can put the ESXi host in lockdown mode. Lockdown mode restricts remote users from directly logging in to this host. It can be accessed only through local console or an authorized centralized management application. It is possible to modify lockdown mode configuration in the host settings, or from the **Direct Console User Interface** (**DCUI**).

In vSphere 6.7, lockdown mode has multiple settings and a user exception list. This allows users and solutions to be excluded from the lockdown mode settings. The following are the different configuration options:

- **Disabled**: Lockdown mode is disabled.
- **Normal**: DCUI is not blocked. Privileged user accounts can still log in to the ESXi host console and exit lockdown mode.
- **Strict**: DCUI is stopped and is only accessible through vCenter:



Strict mode dramatically reduces the manageability of the hosts, because CLI commands cannot be executed from an administration server or script. There is an option to access the ESXi server even under strict lockdown mode but only for users defined in **exception users.** Users in this list retain their original permissions allowing them to interact with the ESXi. Typically, user accounts used for integration purposes, third-party solutions, or external applications are included in the exception users.

Networking

If you are using distributed virtual switches, some specific network security configurations can be managed only from the most advanced settings. For example, to enable the **Bridge Protocol Data Unit** (**BPDU**) filter, you must use a host advanced setting, `Net.BlockGuestBPDU`, as described in **KB 2047822**: **Understanding the BPDU Filter feature in vSphere**, at `https://kb.vmware.com/kb/2047822`.

Of course, the security policies (promiscuous mode, MAC address change, and forge packets) for the virtual switches are still relevant, but for distributed virtual switches, they are all rejected by default (starting with vSphere 5.1).

Virtual switches do not provide firewall functions (ESXi personal firewall works only on VMkernel ports); to implement micro-segmentation, you need solutions such as NSX, although you can achieve some necessary protection using filtering rules on the distributed vSwitch.

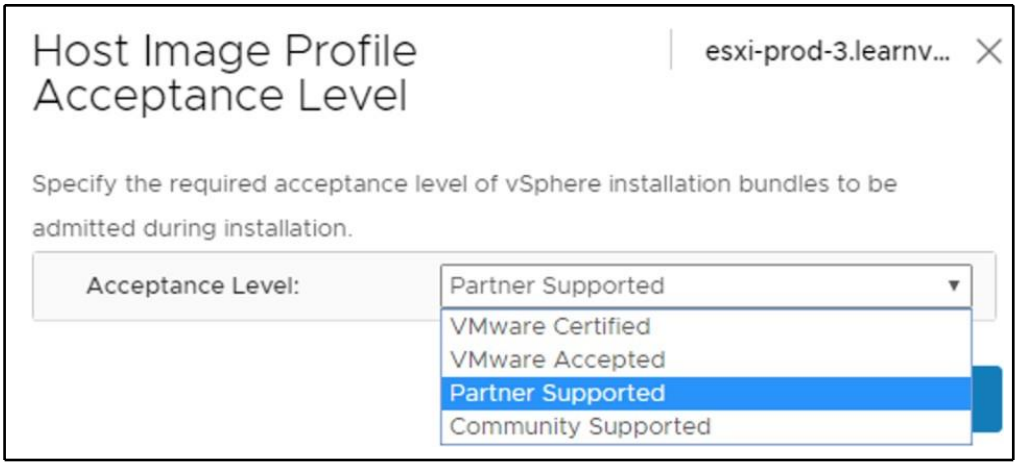
Transparent Page Sharing (TPS)

Recent academic research has demonstrated that it is theoretically possible to leverage TPS to gain unauthorized access to data under certain highly controlled conditions. For more information, see `https://blogs.vmware.com/security/2014/10/transparent-page-sharing-additional-management-capabilities-new-default-settings.html`.

For this reason, in vSphere 6.x, TPS is disabled across different VMs but is still working inside individual VMs. It is still possible to enable it on the entire ESXi, by following **KB 2097593: Additional Transparent Page Sharing management capabilities and new default settings** at `https://kb.vmware.com/kb/2097593`.

VIB acceptance level

By default, ESXi only allows signed **vSphere Installation Bundles** (**VIBs**), because an unsigned VIB represents untested code installed on an ESXi host. You can change the acceptance level for each host, in the **Configure** | **System** | **Security Profile** menu, under the **Host Image Profile Acceptance Level** option:



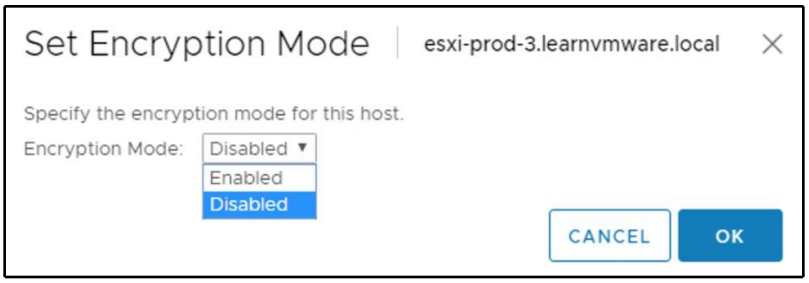
The host image profile supports the four acceptance levels:

- **VMware Certified**: VIBs created, tested, and signed by VMware.
- **VMware Accepted**: VIBs created by a VMware partner but tested and signed by VMware.
- **Partner Supported**: VIBs created, tested, and signed by a certified VMware partner.
- **Community Supported**: VIBs that have not been tried by VMware or a VMware partner. Community Supported VIBs are not upheld and don't have a computerized mark. To ensure the security and respectability of your ESXi, don't permit unsigned Community Supported VIBs to be introduced on your hosts.

Host encryption mode

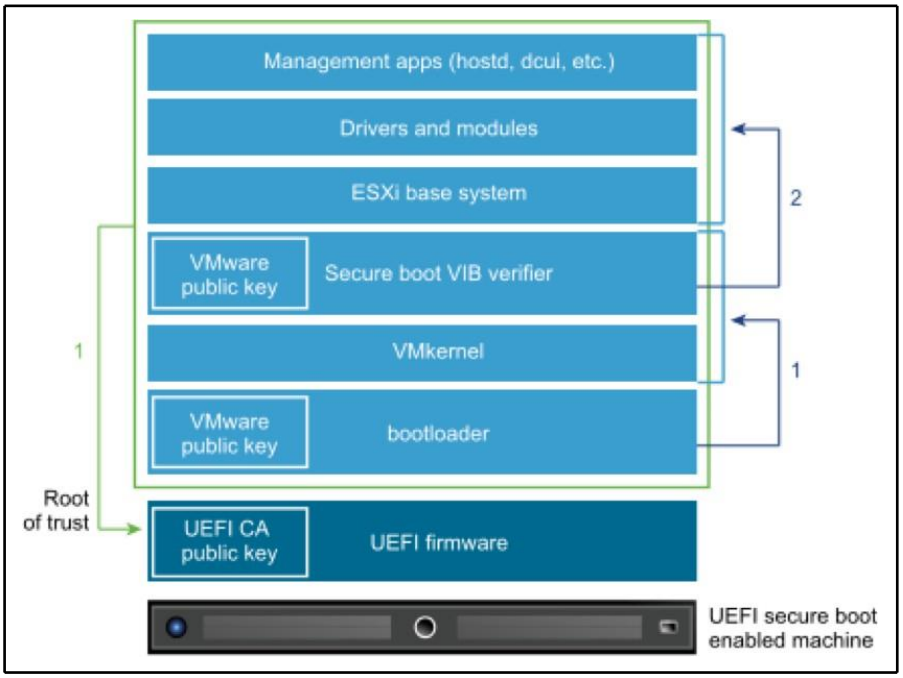
Encryption mode determines whether the host is ready to accept key material. When enabled, core dumps are always encrypted. Enable encryption mode only if the host is secured from unauthorized access to avoid leaking sensitive cryptographic data.

This setting can be configured from the **Configure** | **System** | **Security Profile** menu, under the **Host Encryption Mode** option:



ESXi Secure Boot

For ESXi, Secure Boot is possible using the digital signature of all VIB components, like a cryptographic assurance. By utilizing that computerized endorsement in the host UEFI firmware, at boot time, the approved ESXi VMkernel will in this way approve each VIB against the same certificate:



When ESXi Secure Boot is enabled, you will not be able to install unsigned code on ESXi forcibly.

For more information on how to enable this feature and some possible issues, for example, during the upgrade process, see the following post: `https://blogs.vmware.com/vsphere/2017/05/secure-boot-esxi-6-5-hypervisor-assurance.html`.

vCenter hardening

By using the vCSA, as also suggested by VMware, you can use the same VM hardening suggestions and also benefit from a hardened OS. By default, shell access is disabled. SSH can be enabled during deployment, but you still access the vCSA with a limited set of commands (anyway, enabling the full shell is quite easy).

Similar best practices to the ESXi hypervisor apply to the vCenter Server as well, with a few additional recommendations related to PSC:

- **Check password expiration**: The default vCenter SSO password lifetime is 90 days.
- **Configure NTP**: This ensures that all systems use the same relative time source (including the relevant localization offset). Synchronized systems are essential for vCenter SSO certificate validity, and the validity of other vSphere certificates.

VM hardening

The hardening guide describes a lot of specific VM options but, starting with ESXi 6.0 Patch 5, many of the VM advanced settings are now set to be *secure* by default. This means that the desired values in the Security Configuration Guide are the default values for all new VMs and you don't have to set them manually anymore.

For more information, see the blog post at `https://blogs.vmware.com/vsphere/2017/06/secure-default-vm-disable-unexposed-features.html`.

For VMs, several specific hardening operations should be considered:

- Use templates to deploy VMs
- Minimize use of the VM console
- Prevent VMs from taking over resources
- Disable unnecessary functions inside VMs

For more information, check the official documentation at `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.security.doc/GUID-14CCC8CD-D90D-4227-B2C3-0A93D3C023BA.html`.

It is recommended to disable or remove any virtual hardware that is not vital for the VM (such as the floppy drive). The same security principles as physical servers apply to the VMs:

- Protect the BIOS of the server with the password
- Patch the OS and application
- Enable **Secure Boot**
- Protect the server with the firewall (if connected to an unsecured network)

For virtual networking, NSX can provide the micro-segmentation capability to enforce network security directly at the VM virtual NIC level. Also, at VMworld 2017, a new product was announced—**VMware AppDefense**, a data center endpoint security product that protects applications running in virtualized environments. AppDefense works inside the VMs (as compared to NSX, which only works at the network level) and understands how applications are supposed to work regularly and monitors all changes to that behavior state that indicate a threat.

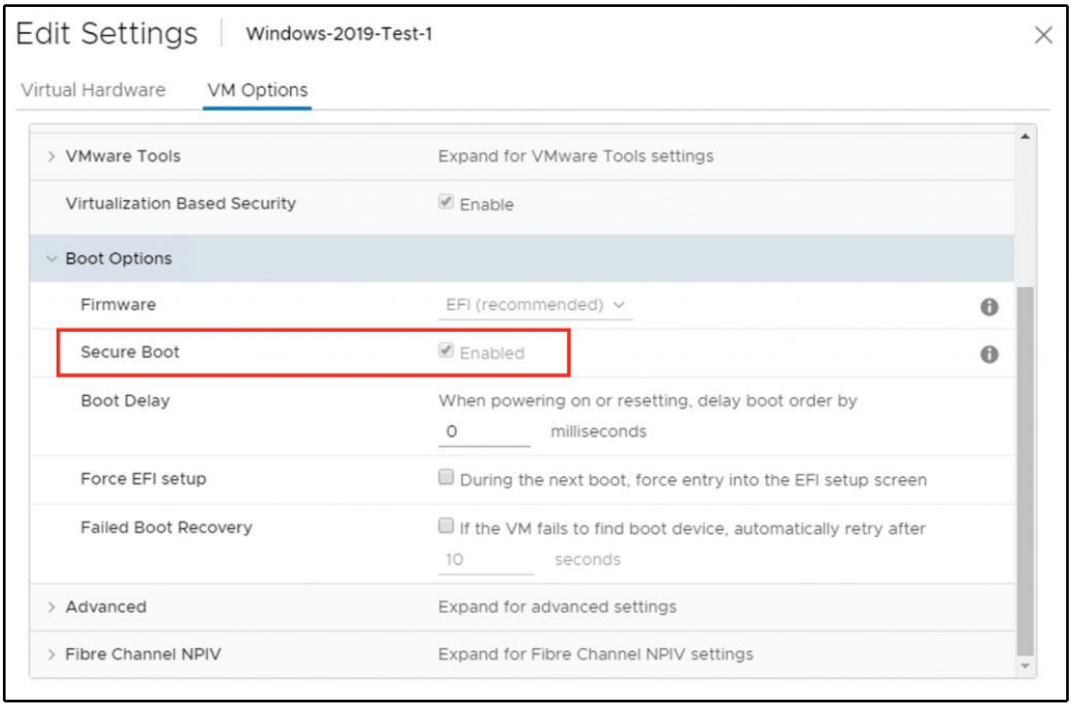
VM Secure Boot

In an OS that supports UEFI Secure Boot, each piece of boot software is signed, including the bootloader, the OS kernel, and OS drivers.
VM Secure Boot has some essential requirements:

- Virtual hardware version 13 or later
- EFI firmware in the VM boot options
- Guest OS that supports UEFI Secure Boot

Some examples of a supported OS are Windows 8 and Windows Server 2012 or newer, VMware ESXi 6.5 and Photon OS, RHEL/Centos 7.0, and Ubuntu 14.04.

You can enable Secure Boot using the vSphere Web Client in the **VM Options** section of the selected VM:



You cannot upgrade a VM that uses BIOS boot to a VM that uses UEFI boot. Only if a VM already uses UEFI boot and the OS supports UEFI Secure Boot can you enable Secure Boot.

Other security aspects

Several other aspects should be considered for security, such as **log management** and **system monitoring**; both of these are useful not only for the security of your environment but also for its manageability.

Another common aspect that will be discussed is certification management, which has been widely improved starting with vSphere version 6.0.

Log management

ESXi has run a syslog administration (`vmsyslogd`) that logs messages from the VMkernel and other framework parts to log records. The log destination can be configured from the vSphere Client; select the host and click **Configure** | **Settings** | **Advanced System Settings**. By default, the `Syslog.global.logDir` parameter is set to `/scratch/log`.

ESXi can be designed to store log documents on an in-memory filesystem. This happens when the host's `/scratch` registry is connected to `tmp/scratch`. When this is done, just a solitary day of logs is put away at once. For more information on ESXi partitions.

You can also set a Syslog Server, both with the GUI (under the advanced settings) or with the CLI, for example, from ESXi Shell:

```
esxcli system syslog config set –loghost tcp://SYSLOG_IP:514
esxcli system syslog reload
```

You can use more Syslog Servers using a comma, or also use SSL connections instead of plain TCP (or UDP); in this case, you must use the syntax `ssl://SYSLOG_SERVER:1514`.
For more information, see **KB 2003322**: **Configuring syslog on ESXi** at `https://kb.vmware.com/kb/2003322`.

You can use an external third-party Syslog Server or the following VMware solutions:

- **VMware Syslog Collector**: Included in vCenter Server. It supports TLS protocol versions 1.0, 1.1, and 1.2. However, it does not have a simple way to analyze logs.
- **VMware vRealize Log Insight server**: A dedicated product also used to correlate different logs and get to the root cause of issues more quickly and efficiently.

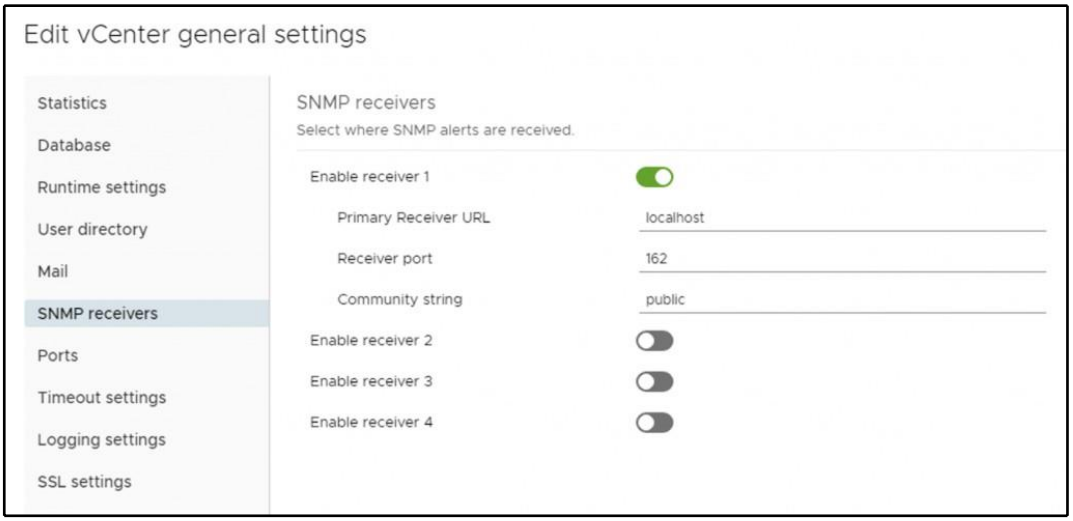
Monitoring protocols

By default, SNMP is not enabled on hosts, either as a service or as a configured node. If you want to enable the SNMP service, to use this protocol, then, for each host, the proper trap destination should be configured as the correct community.

Also on vCenter Server, you can enable sending traps on different alarms, but the SNMP receivers must be set in the general configuration. If SNMP is not being used, it might be better to keep it disabled; if it is not configured correctly, monitoring information can be collected from a malicious host that can then use this information to plan an attack.

To configure SNMP receivers, perform the following configuration:

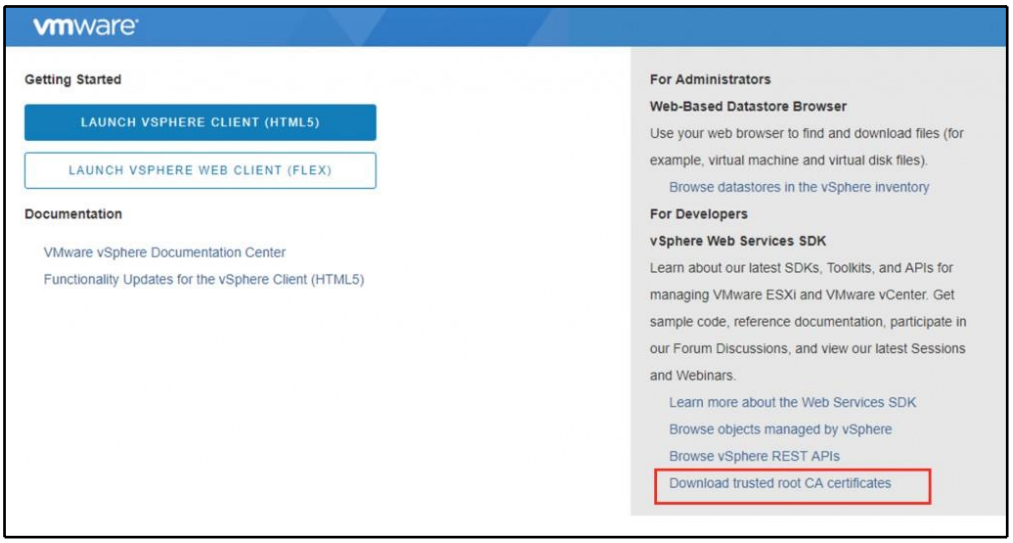
1. Select your vCenter server in the navigator and switch to the **Configure** tab.
2. Under **Settings**, switch to **General** and click the **Edit...** button.
3. Select the **SNMP receivers** section in the menu and configure the receiver URL (IP address), port, and SNMP community name:



Certification management

Starting with vSphere 6.0, the new PSC component includes not only the SSO part but also a certification authority, **VMware Certificate Authority** (**VMCA**), for certification management of all vSphere infrastructure components. This simplifies not only the certification management (with auto-enrollment for expired certificates) but also the trust between the different connections.

In this environment, the vSphere certificates are generated and issued by the VMCA and stored by the **VMware Endpoint Certificate Store** (**VECS**). However, to avoid browser warnings, you need to trust the VMware's CA by adding it in your certification chain. First of all, you need to get the CA root certificate. You can directly download it from the vCenter home page, under **Download trusted root CA certificates**:



You will download a simple download `.zip` file that contains both the CA certificate and the revocation list.

To import the certificate, you can use different approaches for a Windows system:

- **Import manually**: For Internet Explorer, Edge, and Chrome, you can double-click on the certificate and import it into the trusted CA. Firefox has a different certificates repository.
- **Import by using GPO**: Under **Computer Configuration** | **Windows Settings** | **Security Settings** | **Public Key Policies** | **Trusted Publishers**, you can import existing certificates. Be sure to import it into the **Trusted Root Certification Authorities** store.
- **Add as an intermediate CA**: In your existing CA authority.

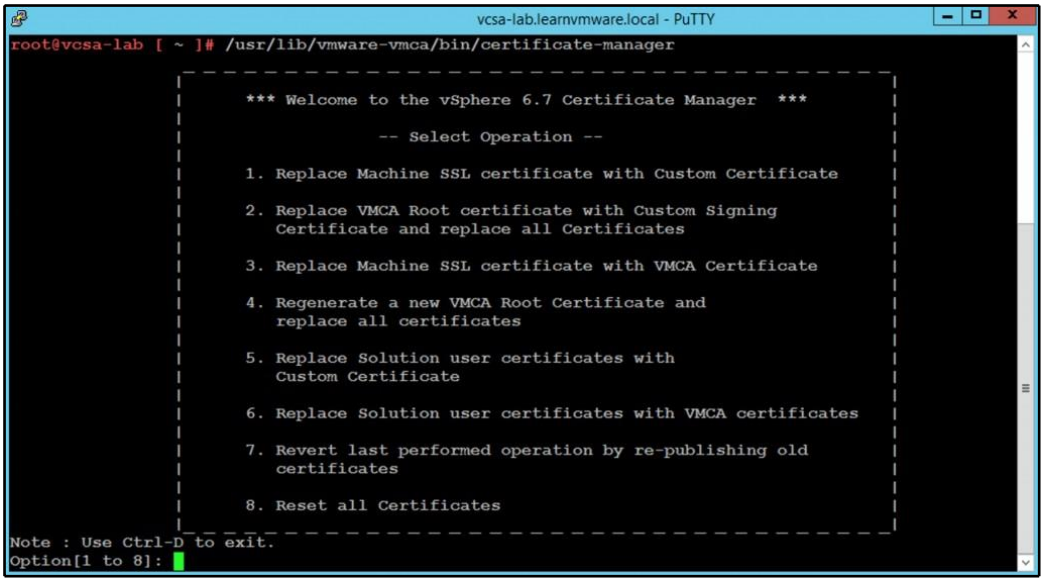
Otherwise, you can replace the CA certificate of VMCA, or don't use it at all and manage all the certificates as in the past. For more information, see **VMware KB 2097936**: **How to use vSphere 6.x Certificate Manager** at `https://kb.vmware.com/kb/2097936`.

If you have an existing PKI within your infrastructure, you can easily replace the VMCA root certificate (self-signed) by a new (signed) certificate from your enterprise authority. In this scenario, the VMCA certificate is an intermediate certificate. VMCA provisions vCenter Server components and ESXi hosts with certificates that include the full certificate chain.

VMCA can only be managed using the CLI. There is no UI available yet. If you need to access VMCA configuration utility, simply log in to the vCSA (or dedicated PSC) and issue the following command:

```
/usr/lib/vmware-vmca/bin/certificate-manager
```

The certificate manager will be displayed as follows:



Please be sure that you have enabled SSH access to the vCSA. You can check the configuration through the VAMI interface of the vCSA under the **Access** menu option.

Encryption options of the vSphere

Some vSphere infrastructures are located strictly on-premise, where the local IT department has full control over the entire infrastructure, but maintaining a hybrid infrastructure can become a security challenge.

Every time your data leaves your organization, for example, if part of the infrastructure is located in an external data center, you should always encrypt such data since it is the most valuable asset of every company.

VMware vSphere can be leveraged to encrypt data in different levels:

- **Encryption at rest**: Data is encrypted on the storage infrastructure, in the other words, where it resides
- **Encryption during transit**: Data is encrypted when transmitted over an unsecured channel

Protecting the data at rest

There are different possible options to store your data securely, which are as follows:

- **Encryption at storage physical level**: This is done by using **self-encrypting drives** (**SEDs**) using full disk encryption, also known as hardware-based **full-disk encryption** (**FDE**). **Opal Storage Specification** is a set of specifications for SEDs developed by the Trusted Computing Group. However, these types of disks are quite costly and also require controllers or storage that support this feature.
- **Encryption at storage logic level**: This is done by using vSAN encryption that uses an AES 256 cipher and eliminates the extra cost, limitations, and complexity associated with purchasing and maintaining SEDs. vSAN datastore encryption is enabled and configured at the datastore level. In other words, every object on the vSAN datastore is encrypted when this feature is enabled.
- **Encryption at VM level**: This is a new feature of the vSphere 6.5 Enterprise Plus edition. Previously, it was only possible with third-party products.
- **Encryption inside the VM**: Consider, for example, using Microsoft BitLocker, or using a Linux-encrypted filesystem (with `losetup`, `luks`, or other tools).

For more information, check the following guide, **How vSphere Virtual Machine Encryption Protects Your Environment**, available at `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.security.doc/GUID-8D7D09AC-8579-4A33-9449-8E8BA49A3003.html`.

VM encryption

A new feature introduced in vSphere 6.5 is the encryption of VMs, which secures the VMDK virtual disks (also `.vmx` and `swap` files are encrypted), making the stored data unreadable.

To get the benefits of encryption, you need to connect vCenter Server to a **Key Management Server** (**KMS**) that provides the necessary keys to encrypt and decrypt VMs using the **Key Management Interoperability Protocol** (**KMIP**) protocol. To establish the connection between KMS and vCenter Server, the KMS performs a certificate exchange.

The components required to allow VM encryption features are the following:

- **KMS**: Generates and stores the keys passed to the vCenter Server to encrypt and decrypt the VMs.
- **vCenter Server**: This is the only component that can log in to the KMS to obtain the keys and push them to ESXi hosts. KMS keys are not stored in vCenter Server, which keeps a list of key IDs only.

A KMS cluster configured in vCenter Server requires that all KMS instances added to the cluster are from the same vendor and must replicate keys. If you use different vendors in different environments, you can create a KMS cluster for each KMS specifying the default cluster. The first cluster added becomes the default cluster.

Be sure to use only a certified KMS provider. Some KMS providers are as follows:

- **HyTrust KeyControl**: `https://www.hytrust.com/`
- **IBM Security Key Lifecycle Manager**: `http://www-03.ibm.com/software/products/en/key-lifecycle-manager`
- **Thales Vormetric Data Security Manager**: `https://www.thalesesecurity.it/products/data-encryption/vormetric-data-security-manager`
- **Gemalto SafeNet KeySecure:** `https://safenet.gemalto.com/data-encryption/enterprise-key-management/key-secure/`

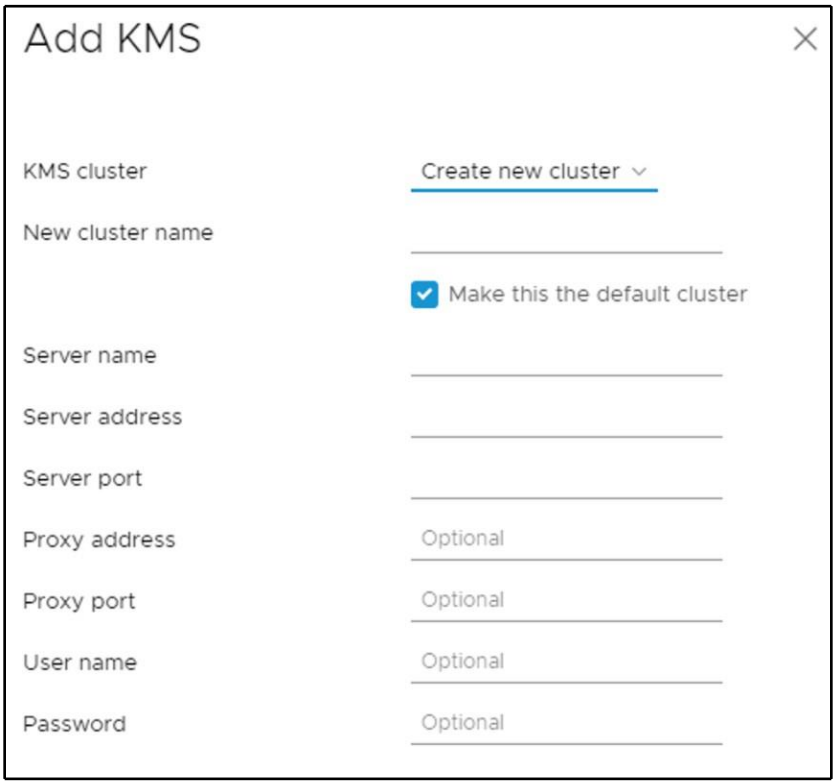
You can have a look at HCL for certified KMS providers at `https://www.vmware.com/resources/compatibility/search.php?deviceCategory=kms`.

A KMS is required to enable and use vSAN encryption as well. Multiple KMS vendors are compatible, including HyTrust, Gemalto (SafeNet), Thales e-Security, CloudLink, and Vormetric.

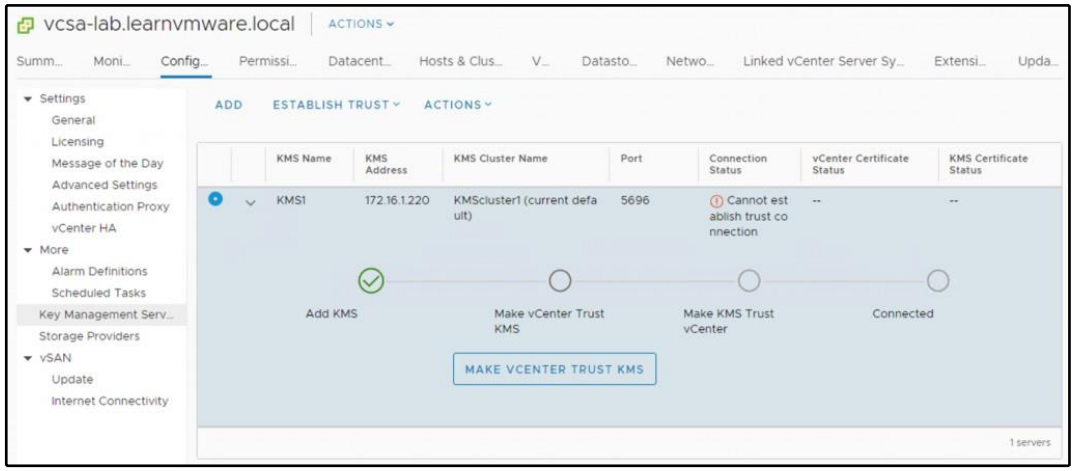
Access to the encrypted virtual disk requires the correct key owned only by the VM that manages the virtual disk. An unauthorized VM that tries to access the encrypted VMDK without the correct key will receive only meaningless data. No additional hardware is required for the encryption/decryption operation, and performance is improved if the processor used supports the AES-NI instruction set, because encryption is CPU-intensive. AES-NI should be enabled in your BIOS, and the VM needs to be powered off before proceeding.

To encrypt VMs, you first need to configure a KMS in vCenter Server:

1. From the vSphere Web Client, select **vCenter Server** in the inventory and select the **Configure** tab. Expand More and select **Key Management Servers** to access the KMS management section.
2. Click the Add KMS icon to add the KMS server (you must have one in your network). Specify the required parameters and click **OK** to save the configuration:

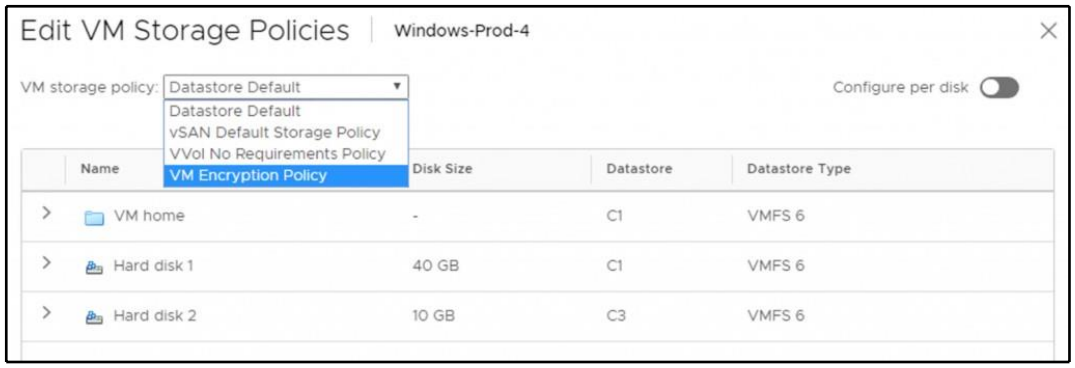


3. Once the KMS server is successfully added to vCenter Server, the **Connection Status** column is displayed as **Normal**. Once you have configured the KMS server, you can start encrypting VMs:



Change the storage policy of a VM by following this procedure:

1. From the vSphere Client, access **vCenter Server**, and right-click the VM to encrypt. Select **VM Policies** | **Edit VM Storage Policies**.
2. From the **VM storage policy** drop-down menu, select the **VM Encryption Policy** option to encrypt the VM and click **OK**:



3. When the encryption process has completed, the VM hardware area in the VMs **Summary** tab displays the **Encryption** field which indicates which components are encrypted.

Here are some recommendations for using encrypted VMs:

- PSC and vCenter Server VMs should not be encrypted.
- The support bundle used to decrypt a core dump is generated using the ESXi host key. If the host is rebooted, the host key may change, and the support bundle can no longer be generated with a password and you might not be able to decrypt core dumps located in the support bundle as well. For this reason, if the host crashes, you should retrieve the support bundle as soon as possible.
- Since `.vmx` files and `.vmdk` descriptor files contain the support bundle, you should not edit these files; otherwise, the VM becomes unrecoverable.

Encryption and decryption of a VM can also be performed using PowerCLI:

- To encrypt a VM, run the following command:

```
Get-VM -Name <vmname> | Enable VM encryption
```

- To decrypt a VM, use the following command:

```
Get-VM -Name <vmname> | Disable VM encryption
```

Encrypted VMs can be a potential challenge for native backup programs, but there is a way to permit backup of those encrypted files in a clear format, to permit indexing and granular restore. Several backup products already support this feature.

Also, you have to consider the following caveats:

- vSphere FT, vSphere Replication, and content library do not work with VM encryption.
- Snapshot operations have some limitations; for example, you cannot select a snapshot of the VM's memory checkbox.
- Cloning an encrypted VM or performing a storage vMotion operation and changing the disk format may not work.
- You cannot encrypt a VM and its disks by using the **Edit Settings...** menu. You have to use the storage policy.
- When you detach a disk from a VM, the storage policy information for the virtual disk is not retained.
- OVF export is not supported on an encrypted VM.

- You can use vSphere VM encryption with IPv6 in mixed mode, but not in a pure IPv6 environment.
- The vCenter Server becomes more critical; only vCenter Server has credentials for logging in to the KMS. Your ESXi hosts do not have those credentials. vCenter Server obtains keys from the KMS and pushes them to the ESXi hosts.

If you want to try VM Encryption, for PoC or Dev and Test environments, you can follow William Lam's post, **KMIP Server Docker Container for evaluating VM Encryption in vSphere 6.5**, available at `https://www.virtuallyghetto.com/2016/12/kmip-server-docker-container-for-evaluating-vm-encryption-in-vsphere-6-5.html`.

Protecting data in motion

Protecting the stored data is only a part of the data security; you also need to encrypt or make secure the network connections and how data is moved. Data in motion is trickier to protect. The best way is always to use secure channels and communication.

At the VM level, it is a problem that is addressed and managed as in any physical environment. Do not only use VLAN (or VXLAN) to segregate traffic, but use the right network traffic rules (in this case, NSX can help with micro-segmentation) and try to avoid clear text network communication.

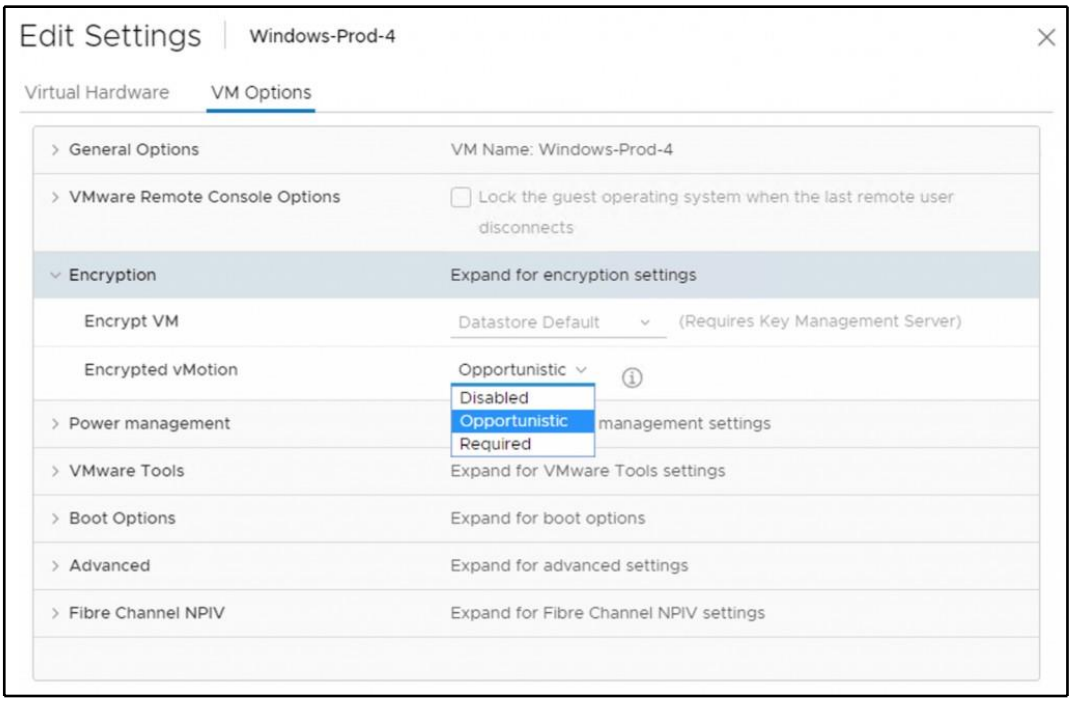
However, you have also the infrastructure to consider. VMware vSphere management traffic is already on SSL connections since version 3.5, but other types of traffic are usually not encrypted, such as vMotion (until vSphere 6.5), or FT logging or storage traffic based on IP, such as **iSCSI** or **NFS traffic**.

If you need to transfer data over an unsecured channel, always use network encryption such as **MACsec** or **IPsec**.

Encrypted vMotion

The vMotion encryption feature isn't merely an encryption of the entire network channel for the vMotion traffic. There aren't certificates to manage.

The encryption happens on a per-VM level; when the VM is migrated, a randomly generated, one-time-use 256-bit key is generated by vCenter (it does not use the KMS). In addition, a 64-bit nonce (an arbitrary number used only once in a crypto operation) is also generated. The encryption key and nonce are packaged into the migration specification sent to both hosts. At that point, all the VM vMotion data is encrypted with both the key and the nonce, ensuring that communications can't be used to replay the data:



Three options regarding encrypted vMotion are available:

- **Opportunistic**: If the source and destination ESXi host supports **Encrypted vMotion**, Encrypted vMotion will be used. If one of the hosts does not support encrypted vMotion, regular (unencrypted) vMotion will be used.
- **Required**: Both source and destination ESXi host must be capable of encrypted vMotion. If the host is non-compliant, the vMotion will fail. In other words, encrypted vMotion will always be used.
- **Disabled**: No encrypted vMotion will be used at all, only regular (unencrypted) vMotion will be used.

17
Analyzing and Optimizing Your Environment

In this chapter, we will show how it is possible to monitor and optimize your vSphere environment. Here, we will look at **virtual machine** (**VM**) optimization through the vSphere approach, as well as through the guest OS approach.

This chapter focuses on monitoring different critical resources, such as computing, storage, and networking, across the ESXi hosts, resource pools, and clusters. Other tools, such as vRealize Operations and third-party tools, will also be described briefly.

In this chapter, we will cover the following topics:

- How to monitor and optimize your vSphere environment
- VM optimization
- The importance of log management
- vRealize Operations
- Third-party monitoring tools

Monitoring a virtual environment

So, *how can we monitor the environment?* You can use two ways—first, through an OS VM approach and second, through a third-party tools approach. We can use two views to monitor the environment:

- **Inside the guest OS tools**: Task Manager, top or monitoring agents installed inside of the guest OS
- **Outside the guest OS tools**: vCenter Server performance charts or ESXTOP

For specific tasks, such as long-term monitoring, GUI tools are usually more powerful, but for quick identification of bottlenecks or troubleshooting, I usually prefer **command-line interface** (**CLI**). As we are covering mainly the VMware world, we will not cover how to monitor your VMs from the guest OS perspective, but we will instead focus on vSphere monitoring itself.

vSphere monitoring

There are many options related to monitoring your vSphere environment, but most commonly you will use the performance monitoring capabilities of vCenter Server. There is an option to monitor your VMs directly from the ESXi web client, but as you have already learned, ESXi hypervisor does not contain a database, thus performance data is only available for the past 60 minutes. Anything older is automatically discarded. On the other hand, with vCenter Server, you can store performance data for years.

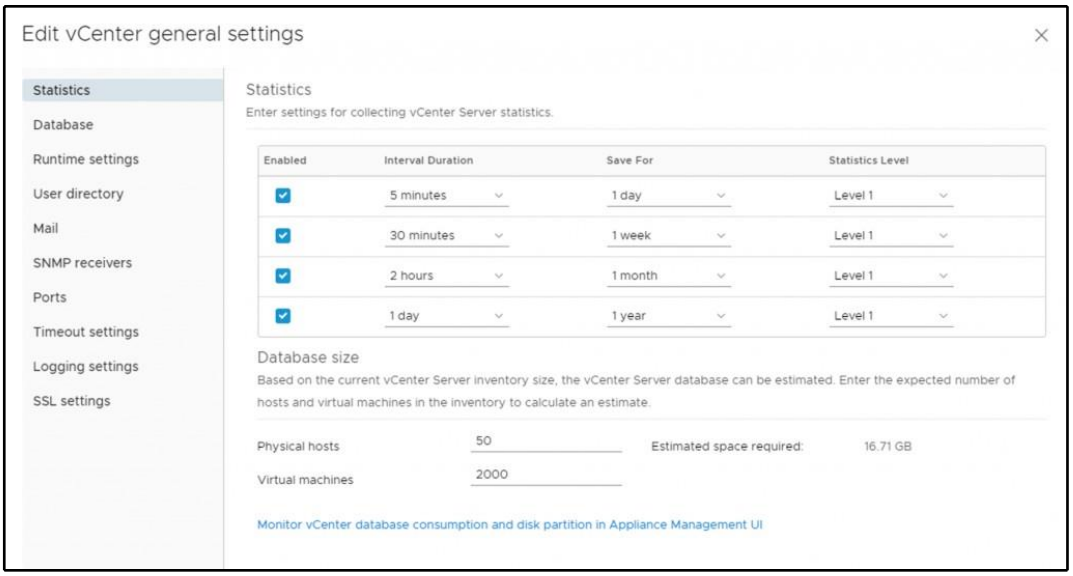
vCenter Server statistics levels

Statistics levels determine the overall size of the performance data within vCenter Server. Based on your preferences, you can alter how long the data will be stored for as well as the individual metrics for different interval durations.

There are five interval durations when working with performance graphs:

- **Real-time**: This data is not stored in the database, but the individual ESXi host is queried when such data is requested. Every 20 seconds, the sample of data is retrieved.
- **Last day**: Real-time statistics are aggregated to 5-minute intervals and stored in the database.
- **Last week:** The last day's statistics are aggregated to 30-minute intervals and stored in the database.
- **Last month**: The last day's statistics are aggregated to 2-hour intervals and stored in the database.
- **Last year**: The last month's statistics are aggregated to 24-hour intervals and stored in the database.

For each duration, you can configure the *statistic level from 1 to 4*. Each statistic level contains a different set of performance counters. With **Level 1**, you get the most useful set of performance counters, and with **Level 4** you get a complete set of all performance counters available in vCenter Server. Of course, the statistic level will affect the overall size of the performance data that will be stored, as shown in the following screenshot:

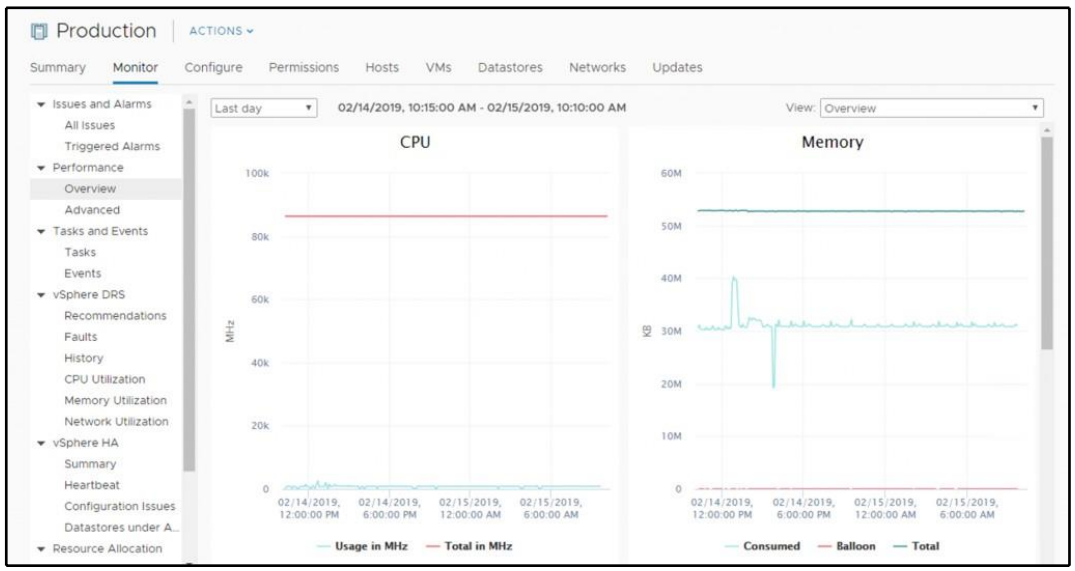


For more information about statistic levels, feel free to check the official documentation at `https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-25800DE4-68E5-41CC-82D9-8811E27924BC.html`.

Performance monitoring with vCenter Server

Each object can be monitored from vCenter Server no matter the type. You can monitor your ESXi hypervisor, single VM, specific resource pool, cluster, or even data center object here.

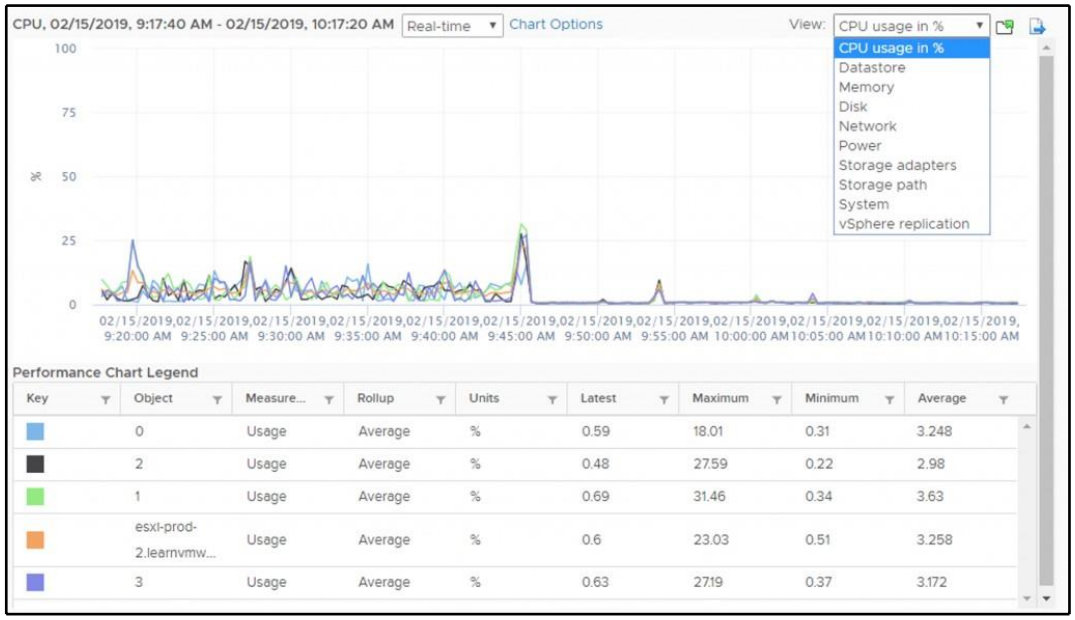
Based on what object you select, different metrics are available based on such item, but for all objects, you can access performance monitoring using the **Monitor** tab and the **Performance** option, as shown in the following screenshot:



For all objects, the following options are available:

- **Overview**: Based on the selected object, the most useful graphs are displayed. There will be different overview content for the data center object and the VM object.
- **Advanced**: Using this option, you can drill down to the individual components of the selected object.

Using the **Advanced** option, you can gain access to different component monitoring options. When you switch to the **Advanced** mode, you can select what view you are interested in. Based on the view, different performance counters related to such a view will be displayed. In the following screenshot, you can see the available views for the ESXi hypervisor object:



Key	Object	Measure...	Rollup	Units	Latest	Maximum	Minimum	Average
	0	Usage	Average	%	0.59	18.01	0.31	3.248
	2	Usage	Average	%	0.48	27.59	0.22	2.98
	1	Usage	Average	%	0.69	31.46	0.34	3.63
	esxi-prod-2.learnvmw...	Usage	Average	%	0.6	23.03	0.51	3.258
	3	Usage	Average	%	0.63	27.19	0.37	3.172

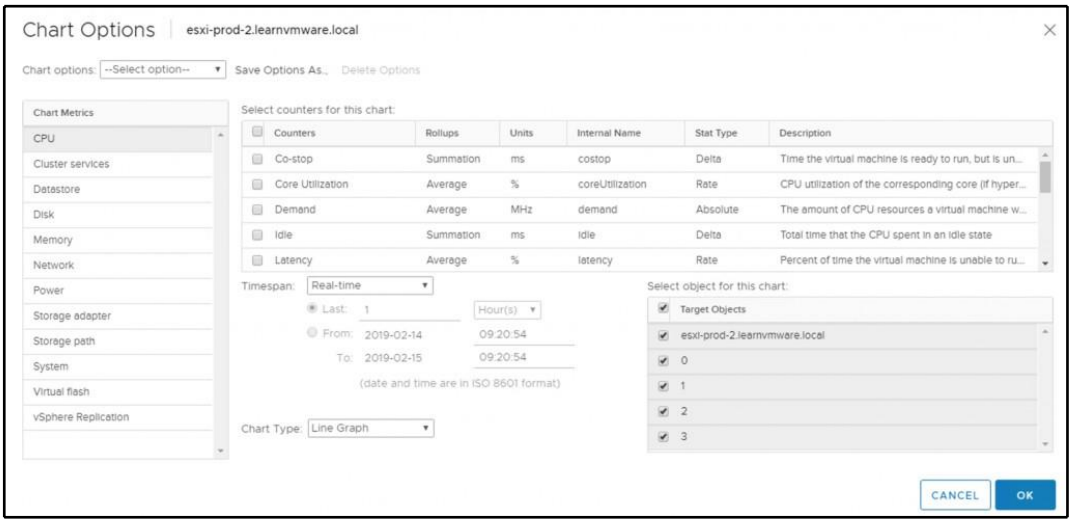
Are you looking for a specific performance counter that is not displayed? Don't worry, only preselected performance counters are displayed by default, but you have an option to select individual performance counters in each view by clicking on the **Chart Options** link.

Here, you can see all of the available performance counters, and you can adjust the graph based on your needs.

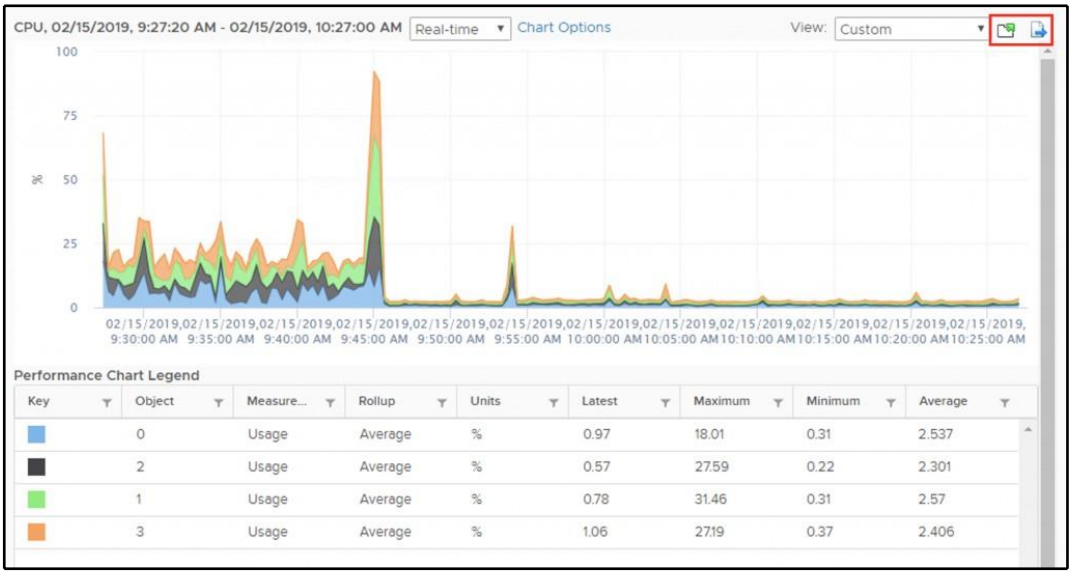
The following options are available:

- **Chart Metrics**: You can switch to the different components you are interested in.
- **Select counters for this chart**: Individual performance metrics are available.
- **Timespan**: You can either select existing a preset (**Real-time**, **Last day/week/month/year**), or you can specify a custom range.
- **Chart Type**: Sometimes different chart types are more useful than the default line graph.
- **Select the object for this chart**: You can specify individual objects here. For example, you can monitor only a single physical CPU or even a core for the ESXi hypervisor. Another example might be an individual **virtual NIC** (**vNIC**) of the VM.

An example of **Chart Options** for ESXi hypervisors is shown in the following screenshot:



Once you define your custom chart, you have an option to either export the graph, or open the graph in a new window (indicated by two small icons in the top right-hand side of the chart), as shown in the following screenshot:



Using export, you can export the chart directly from the vCenter Server in the following formats:

- PNG
- JPG
- SVG
- CSV

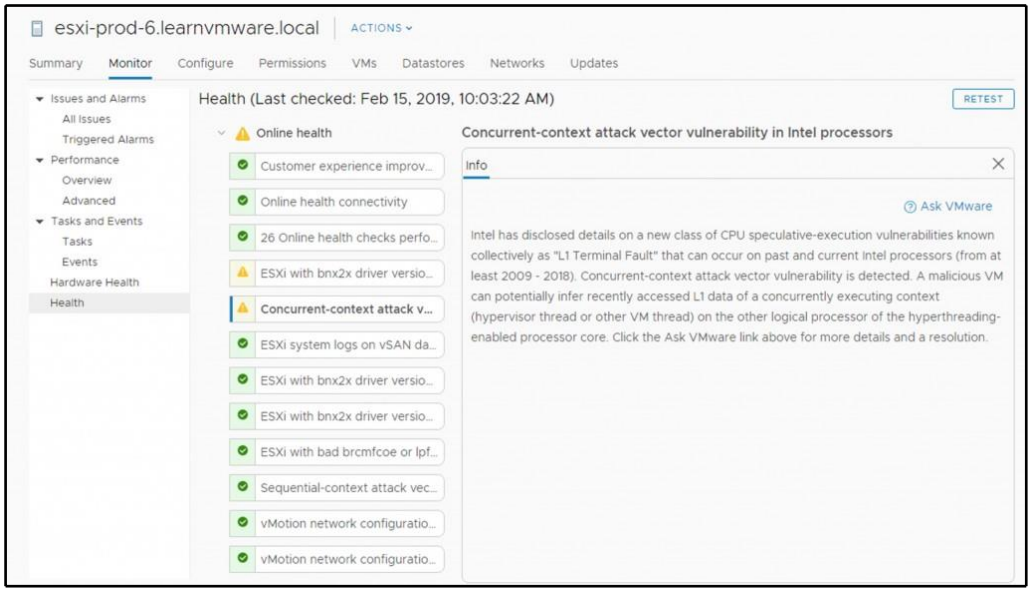
Sometimes, it's also handy to open the graph in a new window for a more detailed view without being limited by the window size of the vCenter Server UI.

ESXi health

There is a new option integrated into vCenter Server 6.7, and that is the online health check of your ESXi hypervisor.

Using this option, you can automatically check against VMware recommendations regarding the physical hardware and driver versions and several vSphere configuration options.

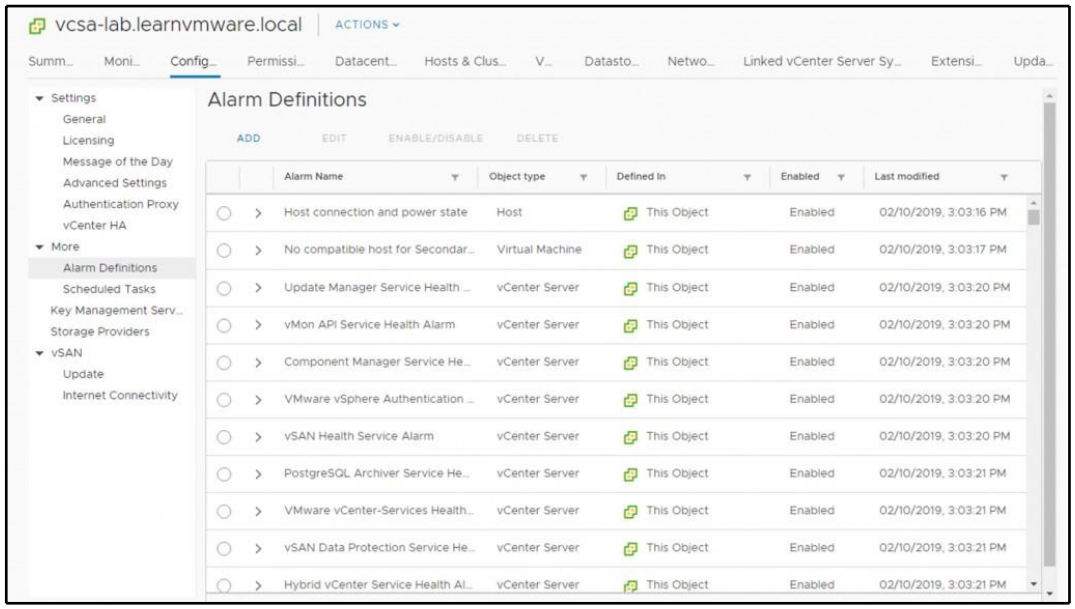
Not only can you see what is not correct in your infrastructure, but you can also display the details of such a warning, as well as having an option to open the related KB using the **Ask VMware** option, as shown in the following screenshot:



Working with alarms

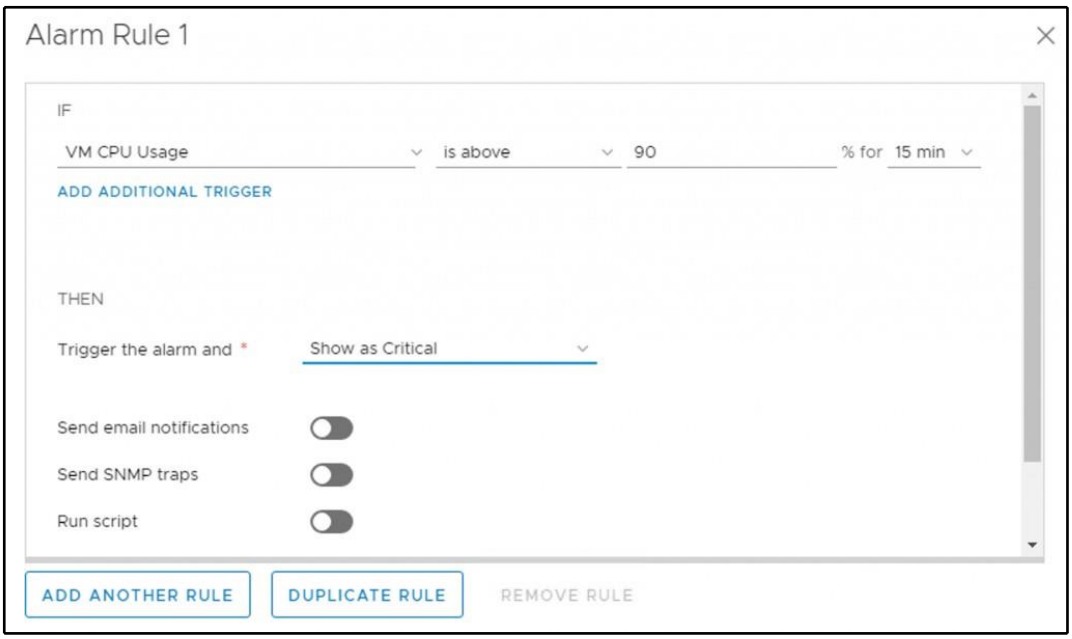
vCenter Server contains several predefined alarms, but there is an option to define your custom alarms as well. This feature is sometimes overlooked, although it provides a comprehensive option to better monitor your vSphere environment.

Default alarms are enabled by default, but you have an option to disable a particular alarm if needed. You can check the exact configuration of the alarm. Alarms are defined at the vCenter level. To access the configuration, select your vCenter Server and, from the **Monitor** tab, switch to the **Alarm Definitions** option, as shown in the following screenshot:



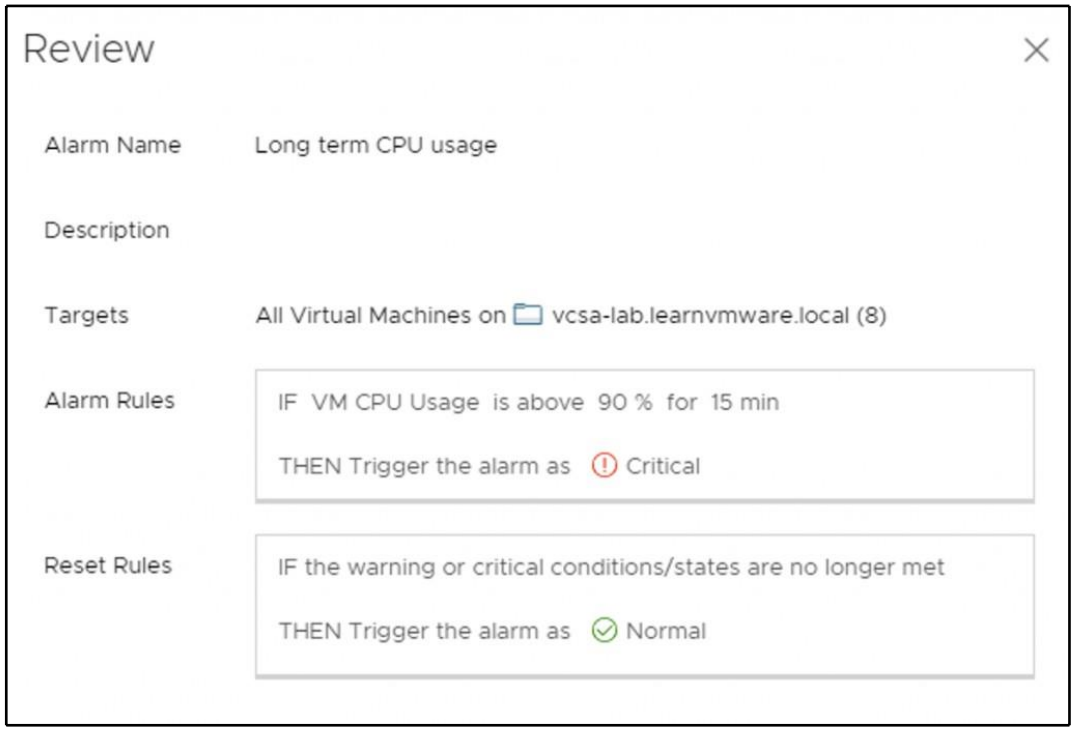
To create a custom alarm, perform the following steps:

1. Click on the **ADD** button in **Alarm Definitions.**
2. Provide the name of the alarm, description, and target type. Based on the target type, different rules that could trigger the alarm will be available.
3. Define the rule that will trigger the alarm. For example, you can raise the alarm when the CPU usage of the VM is above 90% for more than 15 minutes, as shown in the following screenshot:



4. Using the **THEN** clause, you can select whether the object will be in a warning state or critical, and you can also specify to send an email notification, SNMP trap, or even run a custom command.
5. In **Reset Rules**, you can configure what condition will bring the object back to normal state and again, an email, SNMP trap, or script can be run once the condition has changed to **Normal**.

6. Review the alarm definition and save the configuration, as demonstrated in the following screenshot:



CLI monitoring

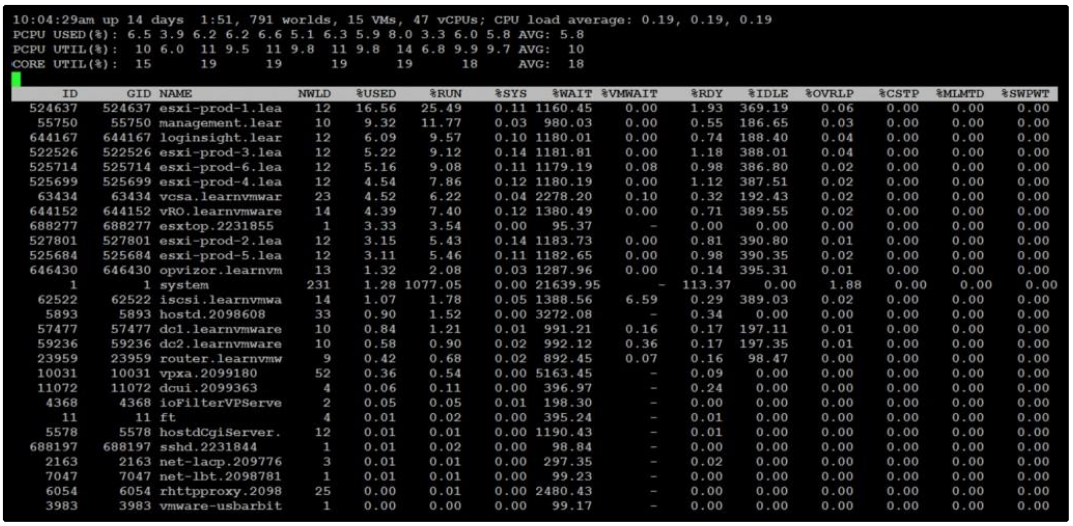
CLI monitoring is usually used for troubleshooting, or when you need to see specific performance counters in real-time. There are two options that you can use for CLI-based monitoring:

- ESXTOP
- PowerCLI

ESXTOP

The `esxtop` command is available at the hypervisor level, so you need to connect to your ESXi hypervisor over **Secure Shell** (**SSH**) first, before you can issue the command.

Once you are logged-in, all you need to do is issue the `esxtop` command from the command line, as shown in the following screenshot:



The windows are automatically refreshed every 5 seconds so you can see what exactly is going on.

ESXTOP has a number of different views you can switch to:

- CPU view
- Interrupt view
- Memory view
- Network view
- Disk adapter view
- Disk device view
- VM disk view
- Power management
- vSAN

To switch to the different view, hit *h* to see what views are available and their corresponding shortcuts, as demonstrated in the following screenshot:

```
Interactive commands are:

fF      Add or remove fields
oO      Change the order of displayed fields
s       Set the delay in seconds between updates
#       Set the number of instances to display
W       Write configuration file ~/.esxtop60rc
k       Kill a world
e       Expand/Rollup Cpu Statistics
V       View only VM instances
L       Change the length of the NAME field
l       Limit display to a single group

Sort by:
        U:%USED         R:%RDY          N:GID
Switch display:
        c:cpu           i:interrupt     m:memory        n:network
        d:disk adapter  u:disk device   v:disk VM       p:power mgmt
        x:vsan
```

You also have an option to sort the items based on different metrics. Again, all the available commands are available from the help view, which you can switch to using the *h* key.

There are a lot of metrics available through ESXTOP, and Duncan Epping wrote a great series of blog posts about ESXTOP available at `http://www.yellow-bricks.com/esxtop/`.

You can also check the following article about interpreting ESXTOP counters at `https://communities.vmware.com/docs/DOC-9279`.

PowerCLI

If you prefer PowerCLI, you can use it for CLI-based monitoring as well. From my perspective, it is an excellent tool because it allows you to monitor not only an individual ESXi hypervisor, but you can connect directly to vCenter Server to monitor multiple objects in your inventory.

So, let's start with the monitoring itself.

As a first step, you need to connect to your ESXi hypervisor or vCenter Server, as follows:

```
Connect-VIServer
```

If you are not sure what parameters are available with a particular command, you can always check the help of a command using the `Get-Help` command.

As a next step, you will probably want to know what statistics are available for a particular object through the following command:

```
Get-StatType -Entity VMname
```

Based on your statistics-level configuration, different metrics will be available, as follows:

```
cpu.usage.average
cpu.usagemhz.average
cpu.ready.summation
mem.usage.average
disk.usage.average
net.usage.average
sys.uptime.latest
disk.used.latest
disk.provisioned.latest
disk.unshared.latest
```

Lastly, you can retrieve particular metric using the `Get-Stat` command, as follows:

```
Get-Stat -Entity VMname -Disk -IntervalSecs 30
```

A similar output will be retrieved:

```
MetricId Timestamp Value Unit Instance
 ------- --------- ----- ---- --------
 disk.usage.average 2/15/2019 11:16:20 AM 51 KBps
 disk.usage.average 2/15/2019 11:16:00 AM 46 KBps
 disk.usage.average 2/15/2019 11:15:40 AM 165 KBps
 disk.usage.average 2/15/2019 11:15:20 AM 81 KBps
 disk.usage.average 2/15/2019 11:15:00 AM 67 KBps
 disk.usage.average 2/15/2019 11:14:40 AM 60 KBps
 disk.usage.average 2/15/2019 11:14:20 AM 52 KBps
```

For more information, feel free to visit the official PowerCLI reference guide available at `https://www.vmware.com/support/developer/PowerCLI/PowerCLI651/html/`.

VM optimization

VM configuration is one of the most crucial decisions you can make in your environment. There are several recommendations and best practices that you should follow to optimize your VM hardware and improve the performance of the infrastructure.

These recommendations are generic, and at your environment, you might not apply all of them. There is nothing like a golden rule here, but the first recommendation is to know your workloads.

Using the default VM templates

Based on the guest OS you want to run inside the VM, make sure you have selected the best possible virtual hardware. VMware optimizes the default VM templates, so you should stick with them.

Based on the OS family and the guest OS version, the wizard will automatically choose the best set of components for the VM.

With every new vSphere version, several improvements and features are available. vSphere 6.7 uses virtual hardware 14, and for example, a persistent memory device is available for the VMs.

Using only the necessary virtual hardware

You should check that you are using only the necessary virtual hardware that your VM requires. Do you need a virtual floppy drive more than one SCSI controller? If not, remove those devices. This will optimize the VM, so that it uses less memory and fewer CPU cycles from the underlying ESXi host, which, in turn, will help you to achieve higher consolidation ratios.

Choosing the correct virtual network adapter

Some older legacy adapters have been replaced over time, and it is essential to upgrade them in your VM. Also, for high-performing VMs, it is strongly recommended to use **paravirtualized devices** instead of emulated devices. For example, **VMXNET Generation 3** (**VMXNET3**) is the latest paravirtualized adapter with multi-queue support, IPv6 offloads, and MSI/MSI-X delivery interruption providing the best possible network performance for your VMs.

VMware tools

Always use the latest version of VMware tools. VM tools are a suite of **device drivers** and **management components** that are installed in the guest OS after OS installation. Although your VMs can run without VM tools, it is strongly suggested to install VM Tools as well as checking for the new versions that might improve some functions of the VM. Every time you upgrade your underlying ESXi hypervisors, you should also upgrade VM tools inside each VM.

Paravirtual SCSI (PVSCSI) storage controller

The PVSCSI storage controller that is used to attach virtual disks in your VM is the most advanced performing storage adapter available today. Compared to traditional emulated LSI storage controllers, it provides **higher I/O** and **lower CPU utilization** for the VM.

Please note that some guest OSes do not include the driver by default, so when you start the installation without the driver, the installer might directly complain that there is no disk to install the system.

Don't use snapshots in production

Snapshots are great for short-term tasks, such as performing some change or upgrade, but they could have a significant impact on the performance. Keep in mind that if the VM has a snapshot, you can't resize the virtual disk, and based on the amount of data written to the snapshot file, the merge of such a snapshot can take a lot of time.

Don't oversize your VMs

Of course, if your VM has fewer resources that it needs, it might affect the performance, but there is a problem with oversized VMs as well.

The VMware ESXi CPU scheduler allocates physical CPU time slots to vCPUs in VMs. If your VMs are configured with multiple vCPUs, the CPU scheduler must wait for physical CPUs to become available. Unused vCPUs will continue to consume system resources even when the system isn't using them. If other single-vCPU VMs use the system, your multi-vCPU VM will have to wait for CPU time.

VMware OS Optimization Tool (OSOT)

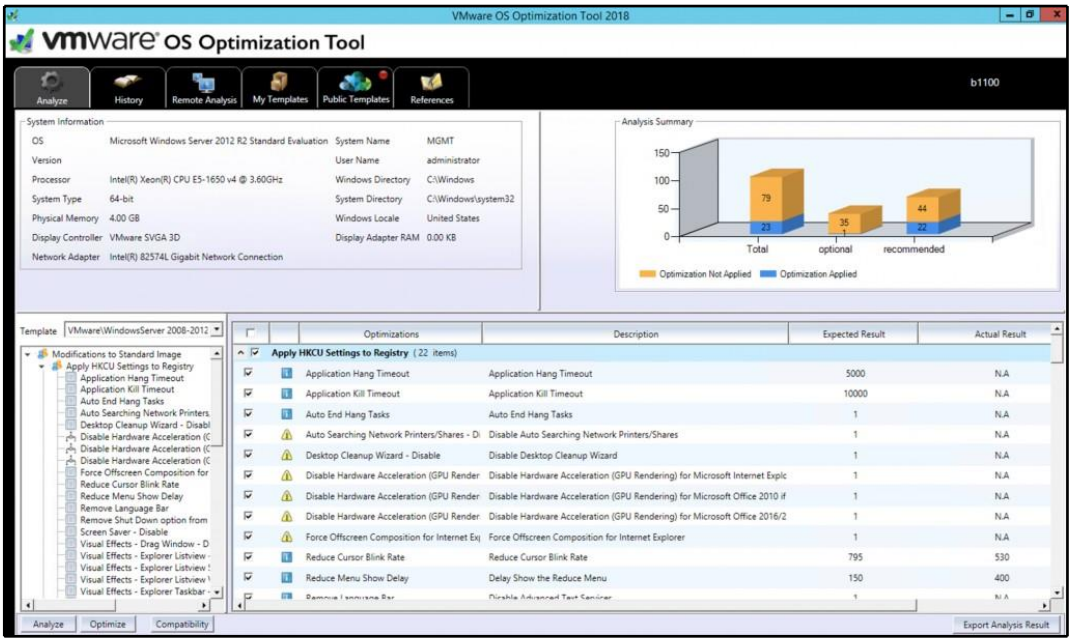
Some of the optimizations can be easily performed by yourself, but some of them – especially optimizing the guest OS—might be tricky.

The optimization tool includes customizable templates to enable or disable Windows system services and features, per VMware recommendations and best practices and across multiple systems. Since most Windows system services are enabled by default, the optimization tool can be used to disable unnecessary services and features to improve performance quickly.

You can perform the following actions using the VMware OSOT:

- Local analyze/optimize
- Remote analyze
- Optimization history and rollback
- Managing templates

Based on the selected template, different services or registry keys will be changed during the optimization, as shown in the following screenshot:



You can download the VMware OSOT for free at `https://labs.vmware.com/flings/vmware-os-optimization-tool`.

Log management

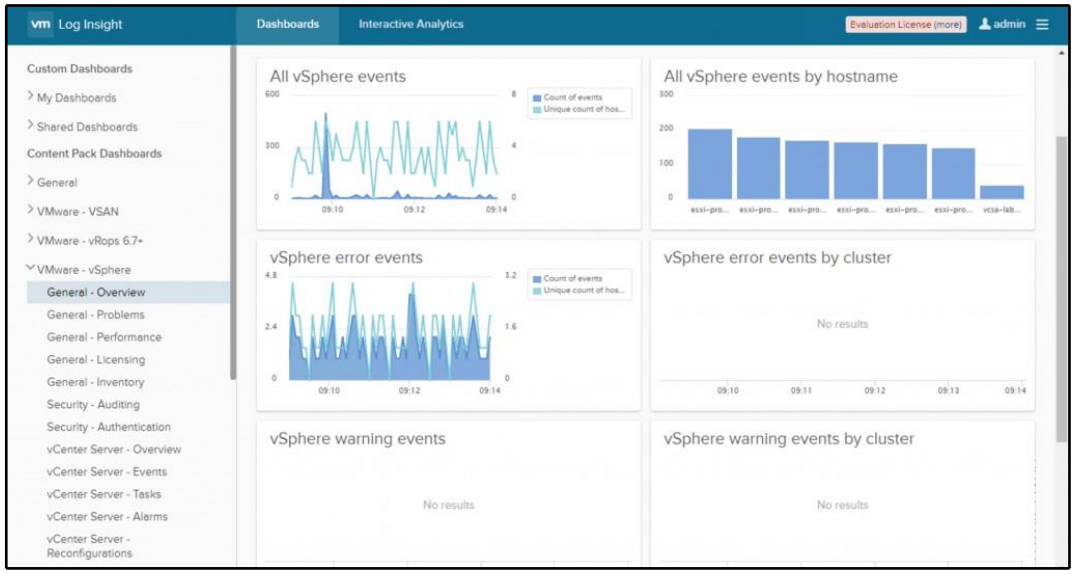
We will cover different log files available for troubleshooting in *Chapter 18, Troubleshooting Your Environment*. However, for now, I would like to focus on the importance of long-term log and event collection of your environment. You should not only monitor your vSphere infrastructure, but other components of the physical infrastructure or even VMs to get a better understanding of what is going on within your environment.

Manual log and event checking might be useful for troubleshooting, but you should consider deploying some centralized tools that can be used for long-term analysis of the environment.

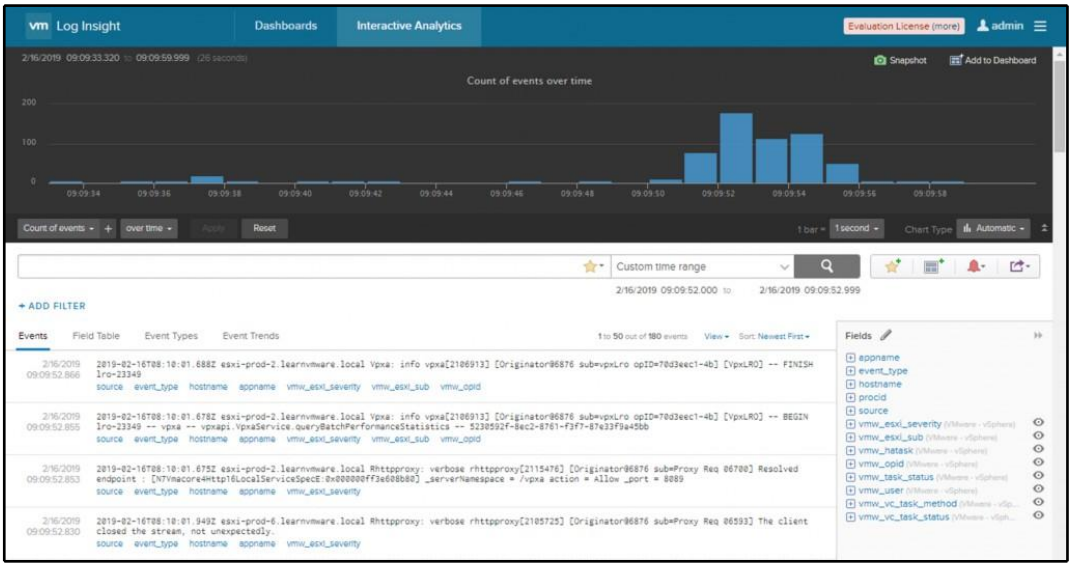
vRealize Log Insight

VMware itself develops this tool, which can be used as a single location to collect and analyze all logs (not only VMware-related ones). You can find additional information about vRealize Log Insight at `https://www.vmware.com/products/vrealize-log-insight.html`.
vRealize Log Insight is available as an **OVA appliance**, so the installation to the environment is a pretty easy task.

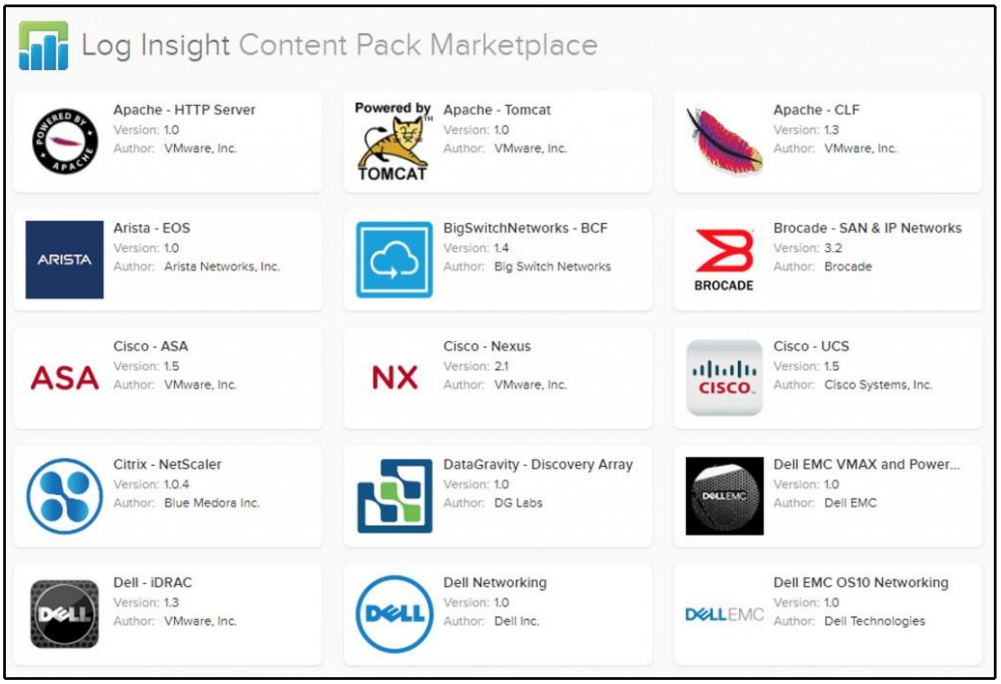
Once Log Insight is installed, you can easily integrate different solutions to Log Insight to start digesting all the logs and events. Based on your integrations, different views and dashboards will be available, as demonstrated in the following screenshot:



The great power of vRealize Log Insight is the **Interactive Analytics** view, where you can quickly drill down through your infrastructure and discover anything you wish, as shown in the following screenshot:



You can also use this tool for Microsoft, Veeam, Cisco, or other solutions only if the content pack is installed from the marketplace:



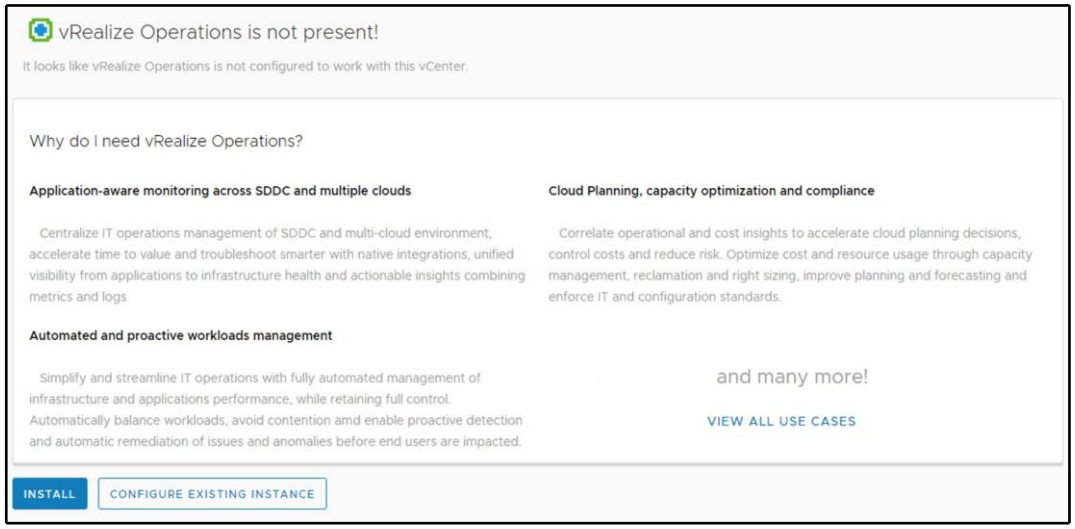
vRealize Operations

vRealize Operations Manager helps you to monitor and report your environment. vRealize Operations is a powerful tool which collects complex information about all objects in your VMware environment. Although long-term monitoring is available in vCenter Server itself, with vRealize Operations, you can get a more in-depth understanding of your environment, as well as different predictions based on your workloads.

For more information, refer to the following link: `https://www.vmware.com/products/vrealize-operations.html`.

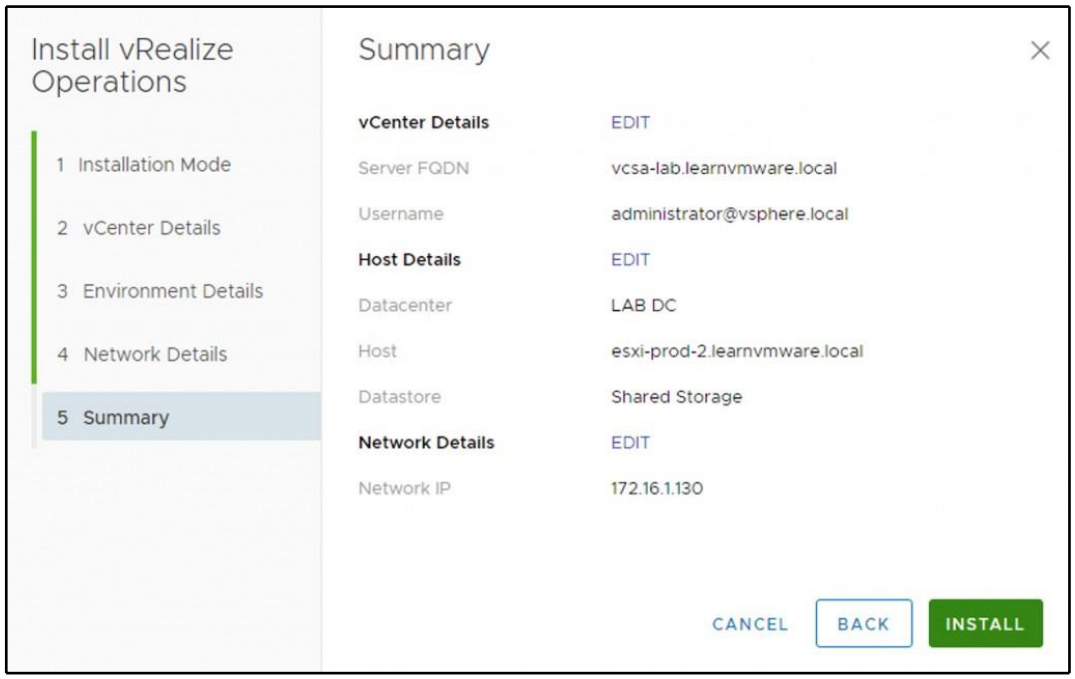
vRealize Operations installation

vRealize Operations is fully integrated into the vSphere 6.7, and you can start the installation directly from the vSphere Web Client by selecting **vRealize Operations** from the menu. If no vRealize Operations are deployed yet, you will have an option to start the installation from there:

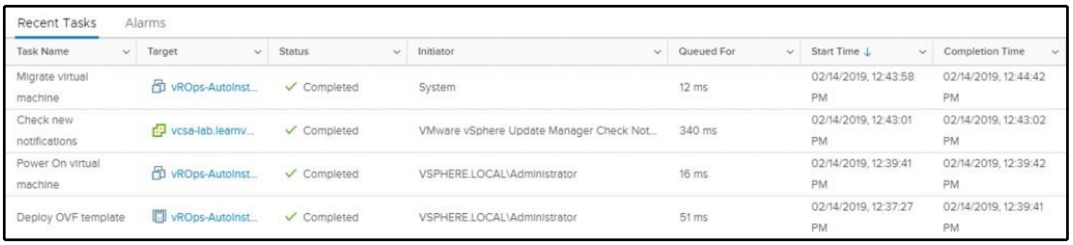


The installation procedure of vRealize Operations is as follows:

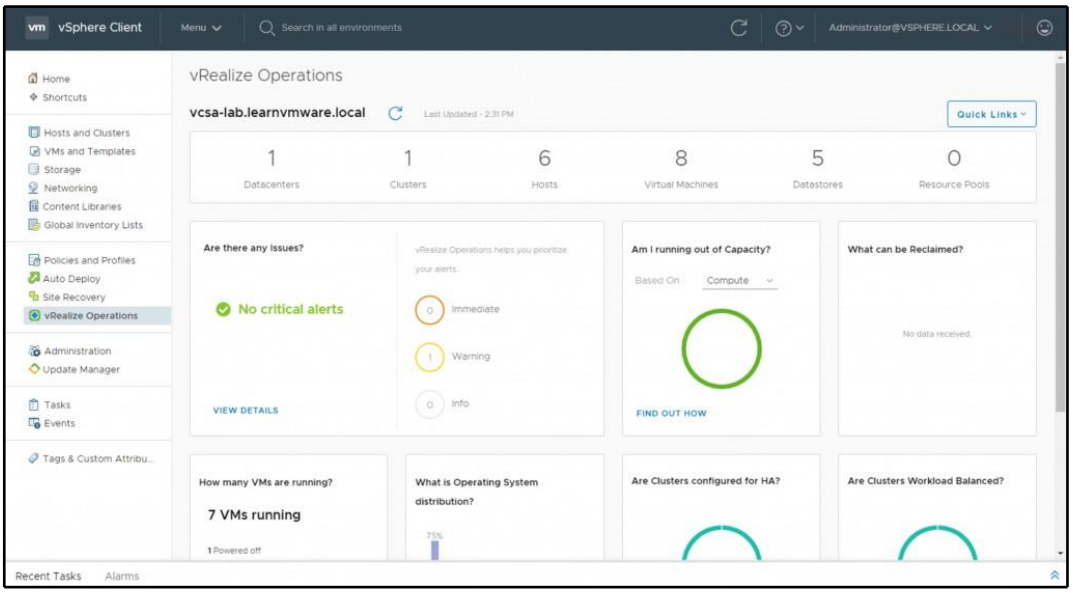
1. Navigate to the **vRealize Operations** menu and select **Install.**
2. You have the option to select either an online or an offline install type. With online installation, you do not need to download anything from the internet; the installer will do that for you.
3. In the next step, you need to connect to the vCenter Server providing the FQDN, username, and password.
4. In **Environment Details**, you need to select on which data center or cluster the new vRealize Operations appliance will be deployed, on which datastore the VM will be stored, and to which port group you should connect vRealize Operations.
5. In the last step, all that remains is the network configuration, such as IP address, gateway, or subnet mask.
6. In the **Summary** view, you have an option to verify everything and start the installation, as shown in the following screenshot:



Once you hit **INSTALL**, you might notice that several tasks will be executed in the environment, as shown in the following screenshot:



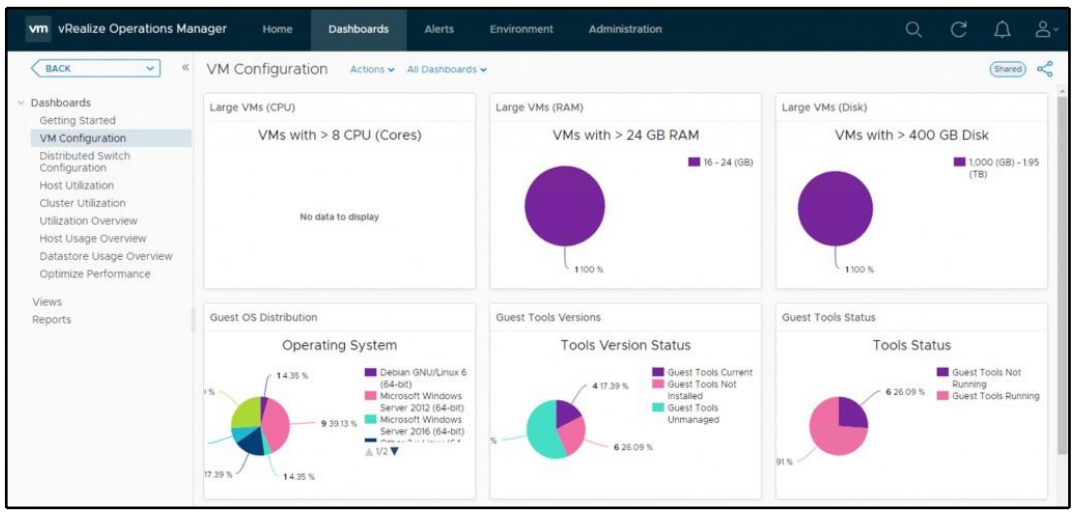
Once the installation is finished, you will have an option to open vRealize Operations directly from the vSphere client, as shown in the following screenshot:



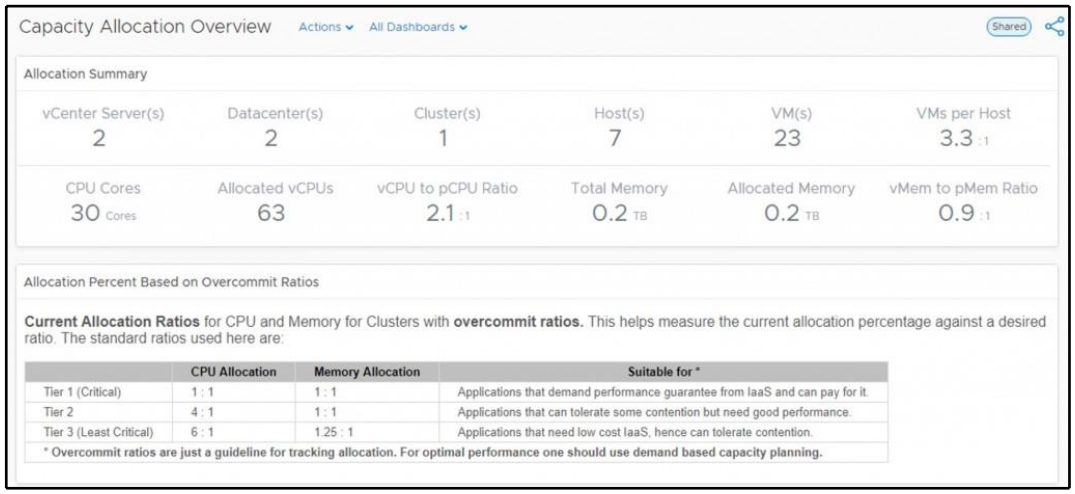
vRealize Operations analytics

There are many different views that you can have a look at using vRealize Operations, so let's show you several interesting views and dashboards that will be available to you:

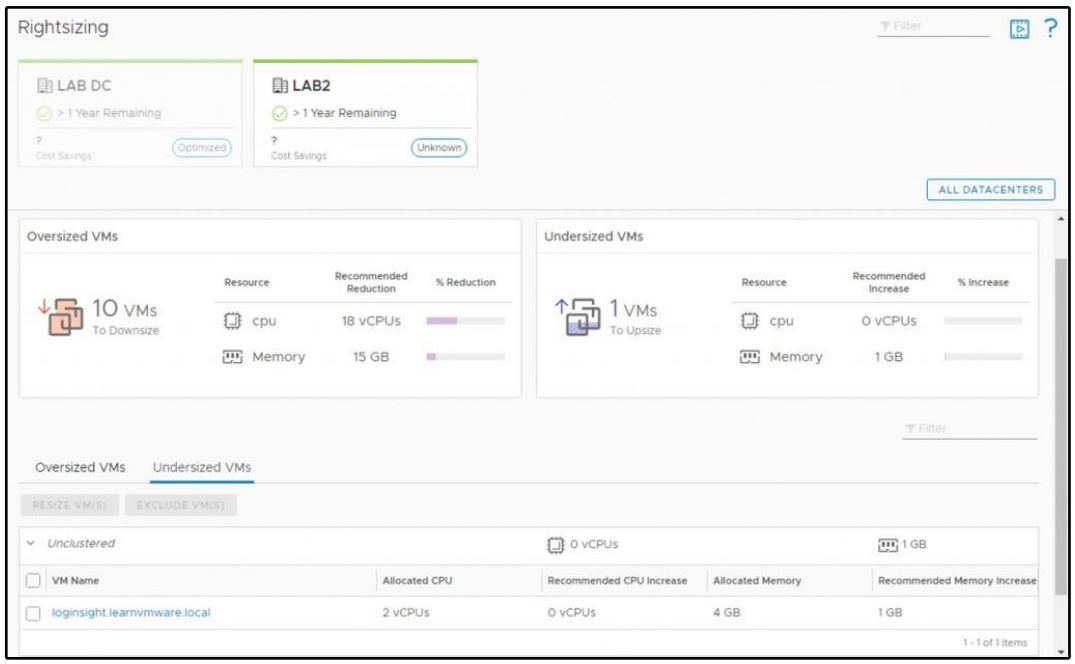
- **VM Configuration dashboard**: You can find exciting information about your VM configuration in the following screenshot:



- **Capacity Allocation**: Certain resource over-subscriptions can affect your vSphere environment. You can find your **current over subscriptions** and **physical-to-virtual resource mapping** in this report, as shown in the following screenshot:



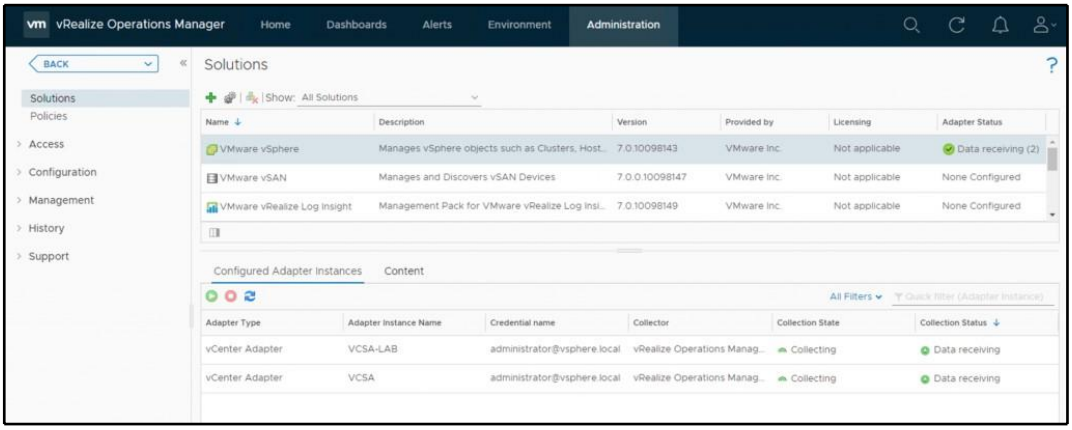
- **Rightsizing**: *Are your VMs oversized or undersized? Should you increase CPU or memory allocation to specific VMs, or, on the other hand, have you configured more resources than the VMs require?*



vRealize Operations integrations

vRealize Operations can be integrated with different products as well as allowing you to monitor the infrastructure end-to-end.

For such integration, management packs are used. For VMware solutions, management packs are already included in vRealize Operations, and all you need to do is configure the integration. vRealize Operations also supports the installation of third-party management packs, as you can see in the following screenshot:



These management packs are distributed as `.pak` files, which you can obtain from your vendor, or you can have a look at the VMware marketplace located at `https://marketplace.vmware.com/vsx/?contentType=1` to see whether there is a management pack for your solution.

We could write a book just about vRealize Operations, but the best way to learn the product is to download the trial and play with it.

Other monitoring tools

If you wish, you can use only VMware products to monitor and manage your environment. However, there are other options available as well. It is beyond the scope of this chapter to describe all of them, so let's stick with a few tools I have worked with and would recommend.

Veeam ONE

Veeam is a well-known player in the backup and availability world that also provides monitoring tools. The core product is called **Veeam ONE**, and it is distributed as an executable file that you directly install on the top of any Windows-based VM.

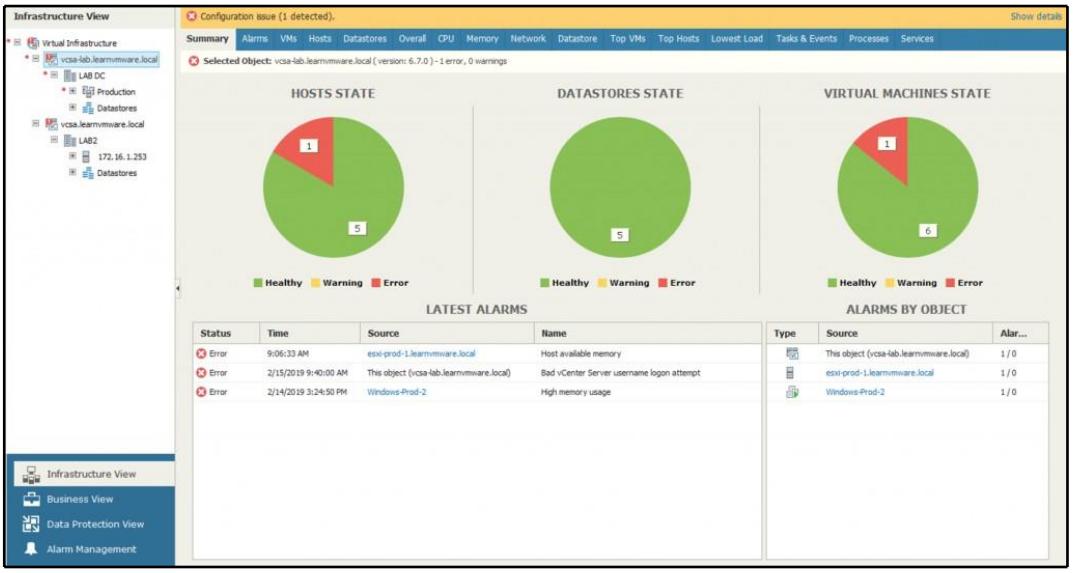
Veeam ONE integrates seamlessly into your entire IT environment, providing complete visibility into virtual and Veeam-protected cloud and physical workloads. It provides monitoring, reporting, and intelligent tools to help your business with the automation and control you need to maintain availability, by protecting against potential problems before operational impact.

The following functionality is available with Veeam ONE:

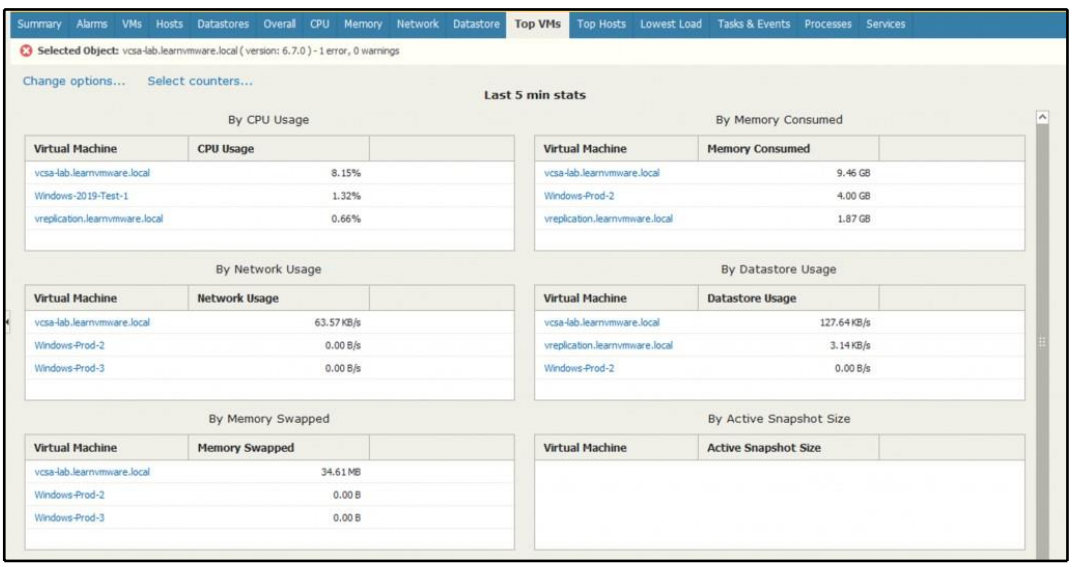
- Real-time dashboards with drill-down views in one click
- More than 200 pre-set alarms based on best practices
- Detailed heat maps with deep visibility into backup repositories and proxies
- Enhanced business view to easily group and monitor the health states of VMs and agents
- Extensive KB connected to each alarm
- Detailed information to help you isolate root cause and quickly resolve issues
- Specific alarm dashboards designed to reduce discovery and troubleshooting times
- Dashboards for backup infrastructure performance and trends
- Dashboards for top consumers by any resource

Veeam ONE comes in two different editions: the Community edition with a limited set of functionality, and the full-featured product for large environments. You can have a look at the comparison at `https://www.veeam.com/products-edition-comparison.html`.

Different dashboards and views are available in Veeam ONE. The overall health of the environment is shown in the following screenshot:



The top VMs in the inventory are shown in the following screenshot:



Opvizor

Opvizor's Performance Analyzer is a performance monitoring and optimization product for virtual environments, specializing in VMware vSphere. Performance Analyzer presents a dashboard that is simultaneously comprehensive and easy to understand and use, but it is also extremely customizable.

Opvizor Performance Analyzer is distributed as an OVA appliance, which you can download from the Opvizor website at `https://www.opvizor.com/how-it-work`.

Again, different views and dashboards are available once the Performance Analyzer digests enough data.

18
Troubleshooting Your Environment

Although VMware vSphere is a reliable platform, sometimes unexpected events occur. In this chapter, we will show you how it is possible to troubleshoot and repair your VMware infrastructure. You might not become a troubleshooting master, but you will learn enough to be ready for any situation that might arise. This chapter will cover the native tools used to troubleshoot performance issues and other issues to improve the virtual environment and workloads.

In this chapter, we will cover the following topics:

- What is troubleshooting?
- Troubleshooting a virtual environment
- Logs
- Troubleshooting vSphere components

What is troubleshooting?

Troubleshooting (**TRBL**) is a complete process where you (in the role of VMware administrator) identify an issue, try to find the origin of the problem, and define the way to resolve it.

The main steps involved during the troubleshooting process are therefore the following:

1. Defining the problem
2. Identifying the cause of the problem
3. Resolving the problem

The complexity of VMware environments is that different layers are involved, and the problem could impact any of the component for different reasons:

- Hardware failures
- Software problems
- Network problems
- Resources contention
- Mistakes in configuration

A big mistake that occurs quite often is considering TRBL only when your environment has failed, for example, with a **Purple Screen of Death** (**PSOD**) error. NO! TRBL is about all problems, and you should start TRBL when there is a problem or when users report problems in terms of performance, reliability, or usability.

The first step of every TRBL process is collecting all the symptoms. Here, you must be careful because the symptoms and the origin of the problem can be entirely different. This stage is crucial for gathering additional information to define the problem.

The typical questions may be—*Can the problem be reproduced? What is the scope? Did the system change before we got notification of the problem? Is the problem documented in the VMware Knowledge Base (KB)?*

When you have all of this information, you can start TRBL from the following three components:

- You start on the VM OS level and continue down to the hardware
- You start at the hardware level and continue up to the VM OS level
- You can start in the middle, at the VMkernel level, and continue up or down

After identifying the cause, you must specify the level of the problem to be fixed for your production environment, assigning a priority:

- **High**: Resolve as fast as a possible
- **Medium**: Resolve during the first possible window
- **Low**: You can wait for the next maintenance window

Solutions levels can be classified as follows:

- **Short**: Typical workaround
- **Long**: Reconfigure or change the advanced configuration

A problem's solution may require the use of different solutions together. But I think the theory is done with, and we can start with some real examples of how to troubleshoot your production environment.

Troubleshooting a virtual environment

From my perspective, GUI tools are not the ideal tools to use for TRBL. You will mostly stick with the logs and CLI commands to dig down into your environment if something goes wrong.

Although some GUI tools might be handy, such as **vRealize Operations** or **vRealize Log Insight** , we will focus on the CLI tools for troubleshooting.

CLI tools

The CLI is the most useful option for TRBL. There are a lot of CLIs available that are used for TRBL in a VMware environment:

- **vSphere ESXi shell**: `esxcli`, which is the new CLI
- **vSphere command-line interface vCLI**: `esxcfg-*`, which is the old CLI

Both CLIs can be used directly from an ESXi host, and the ESXi shell must be enabled in the **Direct Console User Interface** (**DCUI**). You can access the DCUI from the physical console of the host, and also remotely by using a specific hardware vendor card, such as iDRAC for Dell or iLO for HP. When you want to use the CLI using a **remote SSH** session (**PuTTY** is a popular SSH client you can use), SSH protocol must be enabled. The second place where the configuration can be set is through vCenter Server in the Security Profile tab.

esxcli commands

How do we use esxcli? Follow these steps:

1. When you write the basic `esxcli` command and press *Enter*, you will see all the possible commands:

```
[root@esxi-prod-1:~] esxcli
 Usage: esxcli [options] {namespace}+ {cmd} [cmd options]

Available Namespaces:
```

```
device Device manager commands
esxcli      Commands that operate on the esxcli system
itself
            allowing users to get additional information.
fcoe        VMware FCOE commands.
graphics    VMware graphics commands.
hardware    VMKernel hardware properties and commands for
            configuring hardware.
iscsi       VMware iSCSI commands.
network     Operations that pertain to the maintenance of
            networking on an ESX host. This includes a wide
            variety of commands to manipulate virtual
networking
            components (vswitch, portgroup, etc) as well as
local
            host IP, DNS and general host networking
settings.
...
```

We can continue by adding the other namespaces to the ESXi command. Another namespace is `network`, for example. `esxcli` is just a kit composed of a sequence of commands and namespaces.

2. Type `esxcli network` and press *Enter*:

```
[root@esxi-prod-1:~] esxcli network
Usage: esxcli network {cmd} [cmd options]

Available Namespaces:
ens         Commands to list and manipulate Enhanced
Networking
            Stack (ENS) feature on virtual switch.
firewall A set of commands for firewall related
operations
ip          Operations that can be performed on vmknics
multicast   Operations having to do with multicast
nic         Operations having to do with the configuration
of
            Network Interface Card and getting and updating
the
            NIC settings.
port        Commands to get information about a port
sriovnic    Operations having to do with the configuration
of
            SRIOV enabled Network Interface Card and
getting and
            updating the NIC settings.
vm          A set of commands for VM related operations
```

```
diag       Operations pertaining to network diagnostics
....
```

3. To find out the IP addresses of all of the VMkernel ports, use the following `esxcli` command:

```
[root@esxi-prod-1:~] esxcli network ip interface ipv4
address list
Name IPv4 Address IPv4 Netmask IPv4 Broadcast Address Type
Gateway DHCP DNS
---- ------------ ------------ ------------- -------
---- ----------- -------
vmk0 172.16.1.11 255.255.255.0 172.16.1.255 STATIC
172.16.1.254 false
vmk1 172.16.2.1 255.255.255.0 172.16.2.255 STATIC 0.0.0.0
false
vmk2 192.168.101.11 255.255.0.0 192.168.255.255 STATIC
0.0.0.0 false
vmk3 192.168.101.12 255.255.0.0 192.168.255.255 STATIC
0.0.0.0 false
```

4. Once you have found the VMkernel port IP address, you need to know where the TCP/IP stack is. To achieve this, type the following command:

```
[root@esxi-prod-1:~] esxcli network ip interface list
vmk0
 Name: vmk0
 MAC Address: 00:50:56:a9:8a:3f
 Enabled: true
 Portset: vSwitch0
 Portgroup: Management Network
 Netstack Instance: defaultTcpipStack
 VDS Name: N/A
 VDS UUID: N/A
 VDS Port: N/A
 VDS Connection: -1
 Opaque Network ID: N/A
 Opaque Network Type: N/A
 External ID: N/A
 MTU: 1500
 TSO MSS: 65535
 RXDispQueue Size: 1
 Port ID: 33554438
```

To avoid problems in the production environment, I recommend trying all available CLI commands for storage, devices, and so on in a lab environment or using the VMware **Hands-On Lab** (**HoL**). VMware HoL is free and available online at `http://labs.hol.vmware.com`.

esxcfg-*

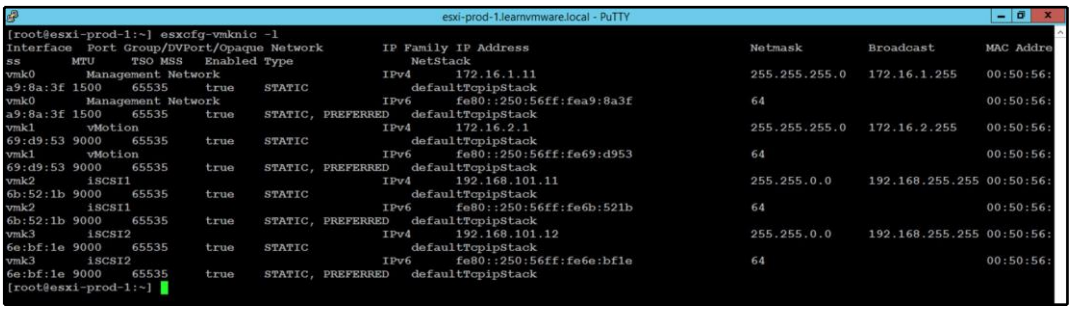
There is also another set of CLI commands that start with `esxcfg-*`. These commands are a little bit older, but still very useful. They can be used from the ESXi host's shell.

In the following example, you can see a duplicated CLI for the same output as `esxcli`. Yes, any CLI duplicates `esxcli`.

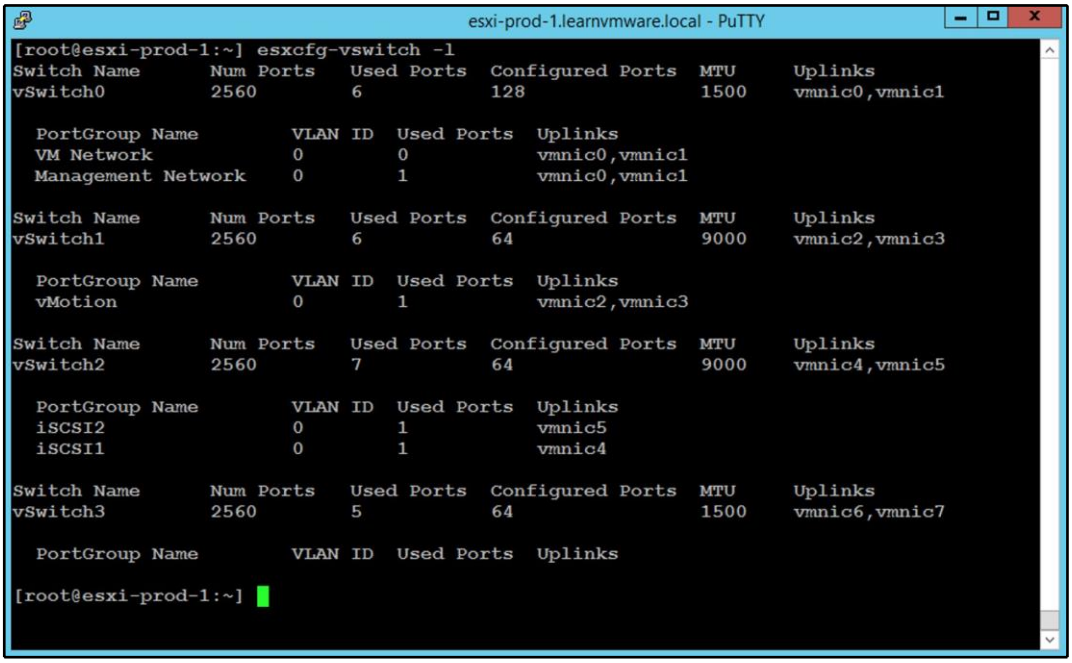
To see all the available commands, you can type `esxcfg-` ,followed by a double hit of the *Tab* button. You will be able to see all possible `esxcfg-` CLI commands:

```
[root@esxi-prod-1:~] esxcfg-
esxcfg-advcfg esxcfg-hwiscsi esxcfg-ipsec esxcfg-nas esxcfg-resgrp
esxcfg-swiscsi esxcfg-vswitch esxcfg-dumppart esxcfg-info esxcfg-
module esxcfg-nics esxcfg-route esxcfg-vmknic
esxcfg-fcoe esxcfg-init esxcfg-mpath esxcfg-rescan esxcfg-scsidevs
esxcfg-volume
```

Using the `esxcfg-vmknic –l` command, you can find all `vmk` IP address, and the net stack in a more natural way than with `esxcli`:



Next, an excellent CLI to use is `esxcfg-vswitch –l`, which lists all vSwitches and vDSes:



Ruby vSphere console

The next CLI that is very useful in real life is the Ruby vSphere console. The Ruby vSphere console is a part of the vCenter Appliance 6.7, and is accessible by typing `rvc`.

You can install `rvc` in your notebook or use a Docker container. The `rvc` command is often used for TRBL vSAN. You can find a great post about it here: `https://blogs.vmware.com/kb/2016/10/tips-tricks-ruby-vsphere-console-rvc-managing-virtual-san-environment.html`:

```
Command> rvc
Install the "ffi" gem for better tab completion.
WARNING: Nokogiri was built against LibXML version 2.9.4, but has
dynamically loaded 2.9.8
Host to connect to (user@host): localhost
Using default username "administrator@vsphere.local".
password:
Welcome to RVC. Try the 'help' command.
0 /
1 localhost/
>
```

Once you are connected, you can browse the available commands using `help`:

```
> help
Namespaces:
alarm
basic
cluster
connection
core
datacenter
datastore
device
diagnostics
...
```

To see commands in a namespace, use `help namespace_name`. To see detailed help for a command, use `help namespace_name.command_name`.

Here's a list of the available commands for VMs:

```
help vm
Commands:
annotate: Change a VM's annotation
answer: Answer a VM question
bootconfig: Alter the boot config settings
clone: Clone a VM
create: Create a new VM
extra_config: Display extraConfig options
find: Display a menu of VMX files to register
ip: Wait for and display VM IP addresses
kill (kill, k): Power off and destroy VMs
...
```

vim-cmd

The `vim-cmd` command could be a very good CLI when you need to start a VM, for example, the vCSA:

```
[root@esxi-prod-1:~] vim-cmd
Commands available under /:
hbrsvc/ internalsvc/ solo/ vmsvc/
hostsvc/ proxysvc/ vimsvc/ help
```

For example, with the `vmsvc` namespace, you can manage the power status of a VM:

```
[root@esxi-prod-1:~] vim-cmd vmsvc/
Commands available under vmsvc/:
acquiremksticket get.snapshotinfo
acquireticket get.spaceNeededForConsolidation
createdummyvm get.summary
destroy get.tasklist
device.connection getallvms
device.connusbdev gethostconstraints
device.ctlradd message
device.ctlrremove power.getstate
device.disconnusbdev power.hibernate
device.diskadd power.off
device.diskaddexisting power.on
…
```

The first step is to list all VMs because we need the **VM identificator** (**VMID**) to power it on:

```
[root@esxi-prod-1:~] vim-cmd vmsvc/getallvms
Vmid Name File Guest OS Version Annotation
19 Windows-Prod-1 [C2] Windows-Prod-1/Windows-Prod-1.vmx
windows8Server64Guest vmx-14
20 Windows-Prod-4 [C1] Windows-Prod-4/Windows-Prod-4.vmx
windows8Server64Guest vmx-14
21 Windows-2019-Test-1 [Shared Storage] Windows-2019-
Test-1/Windows-2019-Test-1.vmx windows9Server64Guest vmx-14
```

When we know the correct VMID, we can power on the VM:

```
[root@esxi1:~] vim-cmd vmsvc/power.on 19
```

With `vim-cmd`, you can try a lot of further commands. In the following example, the following command is used to get network information for VM `19`:

```
[root@esxi-prod-1:~] vim-cmd vmsvc/get.network 19
Networks:

(vim.Network.Summary) {
 network = 'vim.Network:HaNetwork-VM Network',
 name = "VM Network",
 accessible = true,
 ipPoolName = "",
 ipPoolId = <unset>
}
```

vcsa-cli

The next CLI is part of the vCSA, and you can use the `api` command. To see all available commands, you can use `help api list`:

```
Command> help api list
Supported API calls by this server:
com.vmware.appliance.version1.access.consolecli.get
com.vmware.appliance.version1.networking.ipv6.list
com.vmware.appliance.version1.access.consolecli.set
com.vmware.appliance.version1.networking.ipv6.set
com.vmware.appliance.version1.access.dcui.get
com.vmware.appliance.version1.networking.proxy.delete
com.vmware.appliance.version1.access.dcui.set
com.vmware.appliance.version1.networking.proxy.get
com.vmware.appliance.version1.access.shell.get
com.vmware.appliance.version1.networking.proxy.set
com.vmware.appliance.version1.access.shell.set
com.vmware.appliance.version1.networking.proxy.test
...
```

See the following example:

```
Command> api com.vmware.appliance.health.mem.get
Health: green
```

In real life, a VMware administrator needs other commands on the vCSA to restart components, such as the vSphere Web Client:

```
Command> service-control --list
vmware-updatemgr (VMware Update Manager)
vmafdd (VMware Authentication Framework)
vmware-eam (VMware ESX Agent Manager)
vmcam (VMware vSphere Authentication Proxy)
…
```

Restarting the vSphere Web Client service is quite easy from the command line:

```
Command> service-control --stop vsphere-client
Command> service-control --start vsphere-client
```

You should also know basic Linux commands such as `tail`, `vi`, `more`, `less`, `grep`, and `ls` for the TRBL process.

PowerCLI

We can't forget **PowerCLI**, of course. PowerShell lovers will love PowerCLI or PowerNSX. As you already know, you must first log in to the ESXi or vCenter server:

```
Connect-VIServer IP/FQDN
```

You can easily get information about hosts with the `Get-VMHost` command:

```
PS C:\> Get-VMHost

Name ConnectionState PowerState NumCpu CpuUsageMhz CpuTotalMhz
MemoryUsageGB MemoryTotalGB Version

---- -------------- ---------- ------ ---------- ---------- ------
-------------- ------
esxi-prod-5.learn... Connected PoweredOn 4 98 14396 5.997 11.999 6.7.0
esxi-prod-4.learn... Connected PoweredOn 4 142 14396 3.440 11.999
6.7.0
esxi-prod-6.learn... Connected PoweredOn 4 75 14396 1.437 11.999 6.7.0
esxi-prod-2.learn... Connected PoweredOn 4 870 14396 10.978 11.999
6.7.0
esxi-prod-1.learn... Connected PoweredOn 4 309 14396 1.565 11.999
6.7.0
esxi-prod-3.learn... Connected PoweredOn 4 156 14396 6.032 11.999
6.7.0
```

I think that your brain is now full of CLI and commands. However, at this moment, it is not essential to remember every CLI. We need to know and remember what is possible and which CLI can be used for the different parts of the TRBL process.

Logs

For TRBL, it is vital to know where the logs are located. In vCSA, log files are stored in `/var/log/`.

If you login to the vCSA, bash shell is not accessible directly after the login. To access the bash shell, you need to issue the `shell` command first:

```
Command> shell
Shell access is granted to root
root@VCSA-lab [ ~ ]#
root@VCSA-lab [ ~ ]# cd /var/log/vmware
root@VCSA-lab [ /var/log/vmware ]# ls
```

Detailed log locations and descriptions can be found in the VMware KB at `https://kb.vmware.com/s/article/2110014?language=en_USr=2Quarterback.validateRoute=1KM_Utility.getArticleData=1KM_Utility.getArticleLanguage=2KM_Utility.getArticle=1.`

ESXi host logs

The ESXi host log files are very similar to vCSA logs and can be found in the `/var/log/` directory of the host:

```
[root@esxi-prod-1:/var/log] ls -lah
total 444
drwxr-xr-x 1 root root 512 Feb 16 15:48 .
drwxr-xr-x 1 root root 512 Feb 11 20:14 ..
-rw-rw-rw- 1 root root 82 Feb 16 15:48 .vmsyslogd.err
-rw-r--r-- 1 root root 10.0K Feb 16 15:48 .vmsyslogd.err.1
drwxr-xr-x 1 root root 512 Feb 11 20:12 EMU
lrwxrwxrwx 1 root root 21 Feb 16 15:48 Xorg.log ->
/scratch/log/Xorg.log
lrwxrwxrwx 1 root root 21 Feb 16 15:48 auth.log ->
/scratch/log/auth.log
-rw-rw-rw- 1 root root 56.7K Feb 11 20:14 boot.gz
lrwxrwxrwx 1 root root 22 Feb 16 15:48 clomd.log ->
/scratch/log/clomd.log
lrwxrwxrwx 1 root root 29 Feb 16 15:48 clusterAgent.log ->
/scratch/log/clusterAgent.log
lrwxrwxrwx 1 root root 33 Feb 16 15:48 cmmdsTimeMachine.log ->
/scratch/log/cmmdsTimeMachine.log
lrwxrwxrwx 1 root root 37 Feb 16 15:48 cmmdsTimeMachineDump.log ->
/scratch/log/cmmdsTimeMachineDump.log
-rw-r--r-- 1 root root 36.1K Feb 12 09:17 configRP.log
-rw-r--r-- 1 root root 0 Feb 11 20:12 cryptoloader.log
lrwxrwxrwx 1 root root 24 Feb 16 15:48 ddecomd.log ->
/scratch/log/ddecomd.log
lrwxrwxrwx 1 root root 25 Feb 16 15:48 dhclient.log ->
/scratch/log/dhclient.log
lrwxrwxrwx 1 root root 20 Feb 16 15:48 epd.log -> /scratch/log/epd.log
...
```

When you use the `ls –lah` command, you may notice an important key point about ESXi logs; all logs on the host are symbolic links to `/scratch/log/`. When you install the ESXi host on an SD card, a warning message related to non-persistent storage may appear when the installation has completed. To fix this, you must create a datastore and redirect the logs to that datastore.

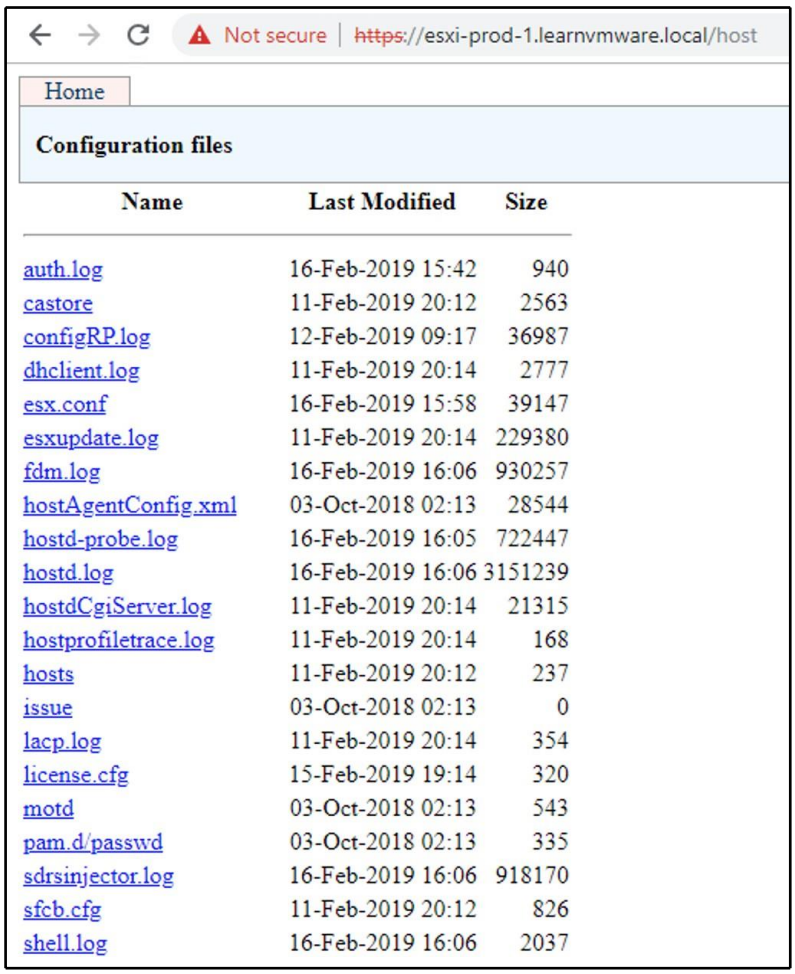
Also, check out **VMware KB 2032823**: **System logs are stored on non-persistent storage at the address** at `https://kb.vmware.com/s/article/2032823?language=en_US`.

As a first step, I recommend checking the `vmkernel.log` log file:

```
[root@esxi-prod-1:/var/log] tail -f vmkernel.log
2019-02-16T15:55:37.094Z cpu3:2317032)Swap: vm 2317014: 5104: Finish
swapping in migration swap file. (faulted 0 pages). Success.
2019-02-16T15:55:37.144Z cpu2:2099363)Config: 703: "SIOControlFlag2" =
0, Old Value: 1, (Status: 0x0)
2019-02-16T15:55:38.486Z cpu0:2317158)DLX: 4319: vol 'Shared Storage',
lock at 10395648: [Req mode 1] Checking liveness:
2019-02-16T15:55:38.486Z cpu0:2317158)[type 10c00002 offset 10395648 v
1968, hb offset 3571712
gen 25, mode 1, owner 5c61c947-9f3a90a4-5da3-005056a97911 mtime 47191
num 0 gblnum 0 gblgen 0 gblbrk 0]
2019-02-16T15:57:17.473Z cpu2:2097177)ScsiDeviceIO: 3068:
Cmd(0x459a40b33c40) 0x1a, CmdSN 0x8fdf from world 0 to dev
"mpx.vmhba1:C0:T0:L0" failed H:0x0 D:0x2 P:0x0 Valid sense data: 0x5
0x20 0x0.
2019-02-16T15:58:28.574Z cpu2:2315918)WARNING: UserSocketInet: 2266:
python: waiters list not empty!
2019-02-16T15:58:28.575Z cpu2:2315918)WARNING: UserSocketInet: 2266:
python: waiters list not empty!
2019-02-16T15:58:28.968Z cpu0:2317573)WARNING: MemSchedAdmit: 1226:
Group vsanperfsvc: Requested memory limit 0 KB insufficient to support
effective reservation 10596 KB
```

There are also other ways to check ESXi logs. One way is using the DCUI from the ESXi console.

Another way is using a web browser and pointing it at your ESXi host using `https://ESXi_IP/host`. After providing the correct ESXi credentials, you will see something like the following screenshot:



Troubleshooting vSphere components

Now that we have learned a lot about CLI or GUI commands and tools, we can start with specific TRBL aspects.

TRBL can be focused on different infrastructural parts, such as the ESXi hosts or the vCenter Server, the network, storage, or can be directly performed at the VM level. Depending on the different issues, it could be better to adopt a bottom-up or top-down approach.

Troubleshooting the vCenter Server

TRBL problems and errors with vCenter Server and ESXi or clusters can be straightforward. One possible issue could be only that some services are no longer working.

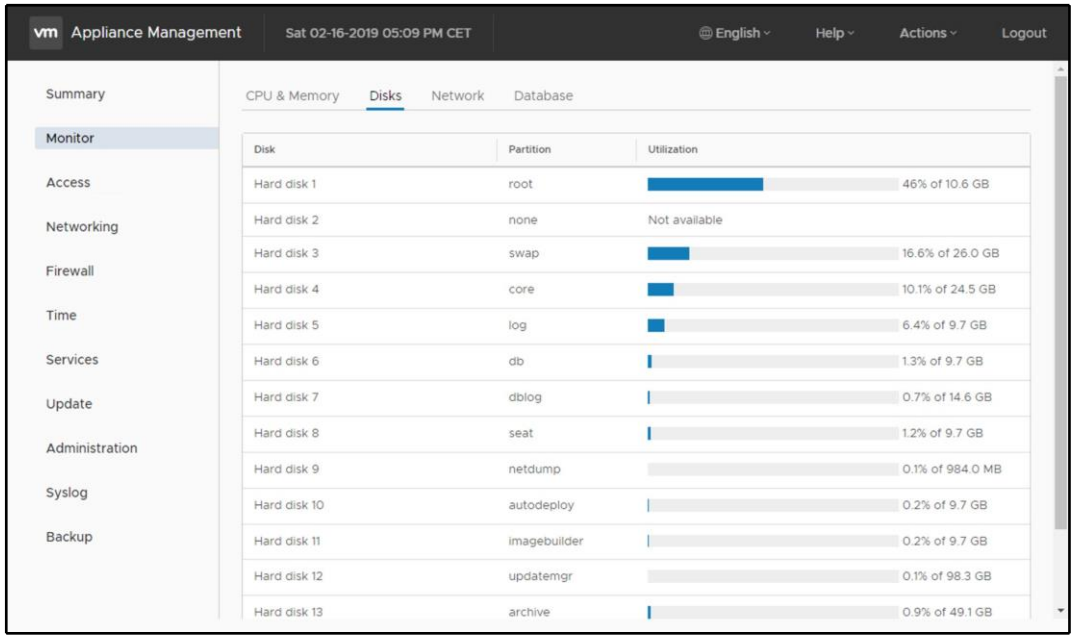
The first action to perform during the troubleshooting of a problem that has occurred in the vCenter Server is to restart the service using the `service-control` command:

```
Command> service-control --list
vmware-vpostgres(VMware Postgres)
vmware-imagebuilder(VMware Image Builder Manager)
vmware-cm (VMware Component Manager)
vmware-vpxd (VMware vCenter Server)
...

Command> service-control --stop vmware-vpxd and service-control --
start vmware-vpx
```

When you see a problem in `vmkernel.log`, you can check the next important component, that is, the vCenter Server database. Typical problems are disk capacity, CPU, and RAM. With vCSA, the only database that can be used is PostgreSQL, but with the Windows version of vCenter Server, both Microsoft SQL Server and Oracle can be used.

For the Windows-based vCenter Server, you may use the available monitoring tools as a part of the database that's used, such as the Management Studio for MS SQL or SQL Developer for Oracle. When you are using the vCSA, life is more much more comfortable, and you can use the new management console, **vCenter Server Appliance Management Interface** (**VAMI**), which is accessible through the browser at `https://VCFQDN or IP:5480`:



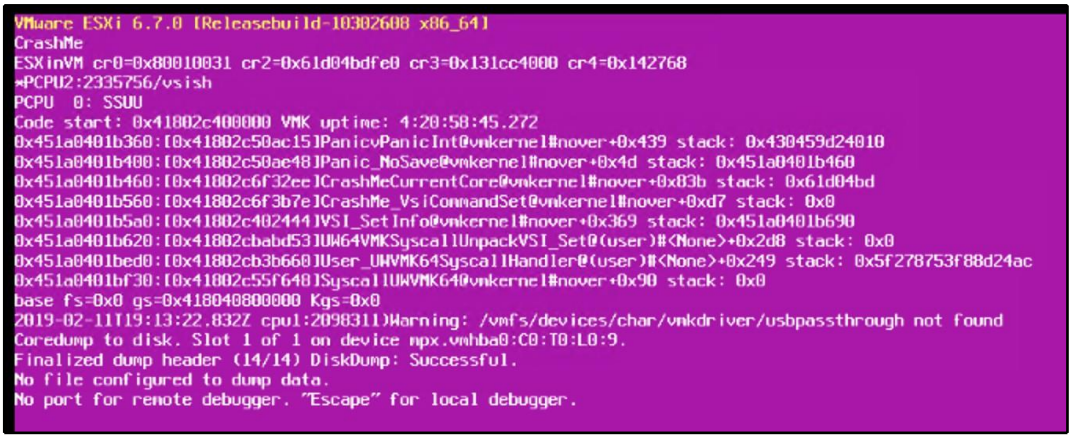
To verify the disk usage, you can go to the **Monitor** and **Disks** tab of the VAMI and check the reported value.

The next step to solve problems related to the database is to try and restart PostgreSQL using the `service-control` command. Next, it is usually useful to restart the problematic vSphere Web Client.

You can use any backup feature that's available so that the vCSA can restore the database.

Troubleshooting the ESXi host

A typical problem you may face with the ESXi host is the fatal PSOD error, as shown in the following screenshot:



This crash is a problem that can be related to CPU, RAM, modules, hardware, or a software bug. When you see this error through iDRAC or iLO, you should take a photo or screenshot to support VMware. You can also try checking the VMware KB to find a resolution. *Did you experience PSOD after an ESXi upgrade?* A possible quick solution to fix this error is to perform an ESXi downgrade. This problem may also occur when you change RAM or firmware in HBA. All of these situations can be potentially problematic for ESXi.

Another way to troubleshoot the problem is to gather maximum information from PSOD, such as the most recent changes in the environment, and restart the ESXi host to create the log bundle requested by the VMware support. The log bundle can be created with a GUI or a CLI with the `vm-support` command.

Does the PSOD occur only on one ESXi host or all ESXi hosts with Qlogic FC HBA? Is it a specific ESXi build that's affected by the problem? You must know this information because it can help you resolve the root problem.

The worst situation you can face is when the VMkernel is in the stopped status, and the ESXi host is not responding. When the VMkernel is busy and doesn't work correctly, as a possible solution, you can try to reboot the host. After rebooting the ESXi, it is very important to gather logs and performance statistics for the support.

You can initiate PSOD using the following command:
`vsish -e set /reliability/crashMe/Panic 1`

Troubleshooting cluster HA or DRS

Problems with cluster HA or DRS may occur at any time. If you have a problem with vSphere HA, you must first check the HA logs stored in `/var/log/fdm.log`. A typical problem can occur during the installation of the FDM agent (HA agent) to the ESXi host. The relevant logs are located in `/var/log/esxupdate.log`.

If you have a problem with the `/root` partition space, it is possible to control the partition through the CLI with the `vdf` command.

Other possible problems in the cluster can be related to VMkernel ports, VMs reservation on the target host, and incorrect network time. If the time is not synced in the VMware environment, you can have issues. Another typical vMotion issue is often due to misconfiguration of the IP address or VLANs.

To check the VMkernel, you can use the `ping` command, but when using the vMotion stack, you should use the `++netstack=vmotion` parameter.

DRS can also cause a problem with vMotion. Keep in mind that DRS uses vMotion to balance resources. Sometimes, a problem is simply due to DRS misconfiguration (manual or fully automated setup) or related to DRS rules.

Troubleshooting a virtual network

Every administrator around the world may have problems with the network connection, and the first action to begin the TRBL process is the use of the `ping` command. Yes, an easy ping can help you, but you have to ping from all directions: ESXi to vCSA and vCSA to ESXi. A typical possible TRBL scenario is when the network is misconfigured, the VLAN is not set up correctly, NIC teaming is not configured correctly, a port on a switch may be down, or there is a hardware problem with a VMNIC or with a physical switch.

The following CLIs can help you identify the problem:

- `esxcfg-vswitch`
- `esxcfg-vmknic`
- `esxcli network`
- `esxcfg-nics`

In TRBL you must, of course, understand what **vSphere Standard Switch** (**vSS**) and **vSphere Distributed Switch** (**vDS**) are. The problem can be anywhere, from the virtual machines to the physical network, software, or hardware.

A very good CLI command for TRBL network is `esxcli network`. For example, a command that can be used to enable or disable a VMNIC is `vmnic`, followed by the `up` or `down` parameters:

```
esxcli network nic down -n vmnic2
esxcli network nic up -n vmnic2
```

Network cards can be checked and listed using the `esxcli network nic list` command:

```
[root@esxi-prod-1:/] esxcli network nic list
Name PCI Device Driver Admin Status Link Status Speed Duplex MAC
Address MTU
Description
------ ------------ ------- ------------ ----------- ----- ------ ---
---------        ____ ---------------------------------------------------------------
vmnic0 0000:0b:00.0 nvmxnet3 Up Up 10000 Full 00:50:56:a9:8a:3f 1500
VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic1 0000:13:00.0 nvmxnet3 Up Up 10000 Full 00:50:56:a9:ac:78 1500
VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic2 0000:1b:00.0 nvmxnet3 Up Up 10000 Full 00:50:56:a9:2a:5c 9000
VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic3 0000:04:00.0 nvmxnet3 Up Up 10000 Full 00:50:56:a9:5c:de 9000
VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic4 0000:0c:00.0 nvmxnet3 Up Up 10000 Full 00:50:56:a9:47:f1 9000
VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic5 0000:14:00.0 nvmxnet3 Up Up 10000 Full 00:50:56:a9:07:f3 9000
VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic6 0000:1c:00.0 nvmxnet3 Up Up 10000 Full 00:50:56:a9:93:db 1500
VMware Inc. vmxnet3 Virtual Ethernet Controller
vmnic7 0000:05:00.0 nvmxnet3 Up Up 10000 Full 00:50:56:a9:7c:20 1500
VMware Inc. vmxnet3 Virtual Ethernet Controller
```

A typical problem that's due to misconfiguration is the selection of a bad virtual machine's port group. Changing to the correct VLAN fixes this problem.

If the management network is misconfigured (the change results in losing the host connection to the vCenter Server), the change is rolled back automatically to the previous configuration.

For additional information, check out **VMware KB 2032823**: **Understanding network rollback and recovery in vSphere 5.1 and later** at `https://kb.vmware.com/s/article/2032908`.

Troubleshooting storage

When you want to resolve problems with storage quickly, you must understand the architecture. There is a big difference between NFS and VMFS filesystems, or if DAS, FC, FCoE, iSCSI, or the new vSAN or VVOL are used. Two great friends will be `esxcli storage` and `esxcli iscsi` for particular use with iSCSI storage.

A problem you may need to analyze is the space occupied on the datastore. Use the `df -h` command to do so:

```
[root@esxi-prod-1:/var/log] cd /vmfs/volumes/
[root@esxi-prod-1:/vmfs/volumes] df -h
Filesystem Size   Used    Available Use%   Mounted on
VMFS-6    499.8G  135.0G  364.8G    27%    /vmfs/volumes/Shared
Storage
VMFS-6    49.8G    5.5G    44.2G    11%    /vmfs/volumes/C1
VMFS-6    49.8G    9.4G    40.4G    19%    /vmfs/volumes/C2
VMFS-6    49.8G    5.6G    44.1G    11%    /vmfs/volumes/C3
VMFS-6  49.8G  1.6G  48.1G  3%  /vmfs/volumes/C4
vfat      249.7M  155.3M  94.4M     62%    /vmfs/volumes/c931af73-
                                           7283cfde-5f11-b0b5ca4e48fc
vfat  285.8M  174.2M  111.6M  61%  /vmfs/volumes/5c60274d-
                                          50968070-edd0-005056a95cde
vfat     4.0G     19.8M   4.0G      0%    /vmfs/volumes/5c602753-
                                          c9175297-8143-005056a95cde
vfat     249.7M   155.3M 94.4M     62%    /vmfs/volumes/4a41ca7c-
                                          4c35da2c-e33e-88f6e0f74792
```

Troubleshooting VMs

The last component you may need to troubleshoot is the VMs. Typical issues are related to power-on, delete, misconfiguration, and resources.

To list the files belonging to a specific VM, use the `ls -lah` command:

```
[root@esxi-prod-1:/vmfs/volumes/5c601d26-2c3ab9f8-
e0ab-005056a95cde/VCSA-lab.learnvmware.local] ls -lah
total 36349248
drwxr-xr-x 1 root root 88.0K Feb 16 15:50 .
drwxr-xr-t 1 root root 76.0K Feb 14 12:07 ..
-rw-r--r-- 1 root root 92 Feb 16 15:20 VCSA-
lab.learnvmware.local-1af96f43.hlog
-rw------- 1 root root 10.0G Feb 11 20:35 VCSA-
lab.learnvmware.local-1af96f43.vswp
-rw------- 1 root root 12.0G Feb 16 16:16 VCSA-lab.learnvmware.local-
flat.vmdk
-rw------- 1 root root 8.5K Feb 14 15:22 VCSA-
lab.learnvmware.local.nvram
-rw------- 1 root root 546 Feb 16 15:20 VCSA-
lab.learnvmware.local.vmdk
-rw-r--r-- 1 root root 0 Feb 10 13:48 VCSA-lab.learnvmware.local.vmsd
-rwxr-xr-x 1 root root 4.1K Feb 16 15:20 VCSA-
lab.learnvmware.local.vmx
-rw------- 1 root root 0 Feb 16 15:20 VCSA-
lab.learnvmware.local.vmx.lck
-rwxr-xr-x 1 root root 4.1K Feb 16 15:20 VCSA-
lab.learnvmware.local.vmx~
lab.learnvmware.local_8-flat.vmdk
-rw------- 1 root root 546 Feb 11 20:13 VCSA-
lab.learnvmware.local_8.vmdk
-rw------- 1 root root 10.0G Feb 11 20:14 VCSA-
lab.learnvmware.local_9-flat.vmdk
-rw------- 1 root root 548 Feb 11 20:13 VCSA-
lab.learnvmware.local_9.vmdk
-rw-r--r-- 1 root root 222.3K Feb 16 07:20 vmware-185.log
-rw-r--r-- 1 root root 223.8K Feb 16 08:00 vmware-186.log
-rw-r--r-- 1 root root 222.3K Feb 16 08:50 vmware-187.log
-rw-r--r-- 1 root root 228.4K Feb 16 11:15 vmware-188.log
-rw-r--r-- 1 root root 228.2K Feb 16 13:50 vmware-189.log
-rw-r--r-- 1 root root 226.0K Feb 16 15:20 vmware-190.log
-rw-r--r-- 1 root root 183.7K Feb 16 16:09 vmware.log
-rw------- 1 root root 110.0M Feb 16 15:19 vmx-VCSA-
lab.learnvmware.local-452554563-1.vswp
```

During the TRBL process, `vmware.log` is the virtual machine's log file, which helps you to understand the problem better. In this log, you will see all the details about the problem. The name of the log file, `vmware.log`, is the same for each VM.

Many problems are due to resources, resource pools, and vApp, thus you should be very careful. If it is not a requirement, don't use reservations or limits for VMs.

Although TRBL can fix most problems, there are situations where restoring a VM from the backup is the only possible solution. Backup is an essential part of vSphere management.

19
Building Your Own VMware vSphere Lab

In this chapter, we will focus on different techniques that can be used to enhance your VMware vSphere skills. The majority of the chapter will look at various aspects of building your lab, including different approaches you can take and the pros and cons of each.

After that, we will cover one particular solution in more detail, which is having a vSphere lab environment running on the dedicated physical server in the data center. We will learn how to install and configure different components of the lab.

Running your vSphere lab should be one of your primary concerns, since, with your lab, you can test any component of VMware vSphere and gain the required experience and knowledge.

In this chapter, we will cover the following topics:

- The importance of lifelong learning
- Choosing the right platform
- Software components and licensing
- Architecture and logical design
- A detailed implementation guide

The importance of lifelong learning

It is always advantageous for professionals in the IT field to have a home lab environment, because it enables you to experiment with different technologies. It is always a great learning experience to build a lab from scratch by yourself. Building your own VMware vSphere lab is much easier than you might think and it will enable you to extend your VMware skills and further your career.

Why build a lab?

There are many reasons why someone would want to build a virtualization lab. Two of the most common reasons are the following:

- **Exam study**: Before you apply for an exam, you need to test everything and be entirely sure about how different components interact. You also need to be confident with the different configuration options and have an awareness of how to perform different tasks.
- **Hands-on learning**: One of the most common reasons for running a lab is to be able to gain real hands-on experience. You can, of course, read books or watch videos about different components of VMware vSphere, but being able to configure and maintain the environment by yourself is an essential skill.

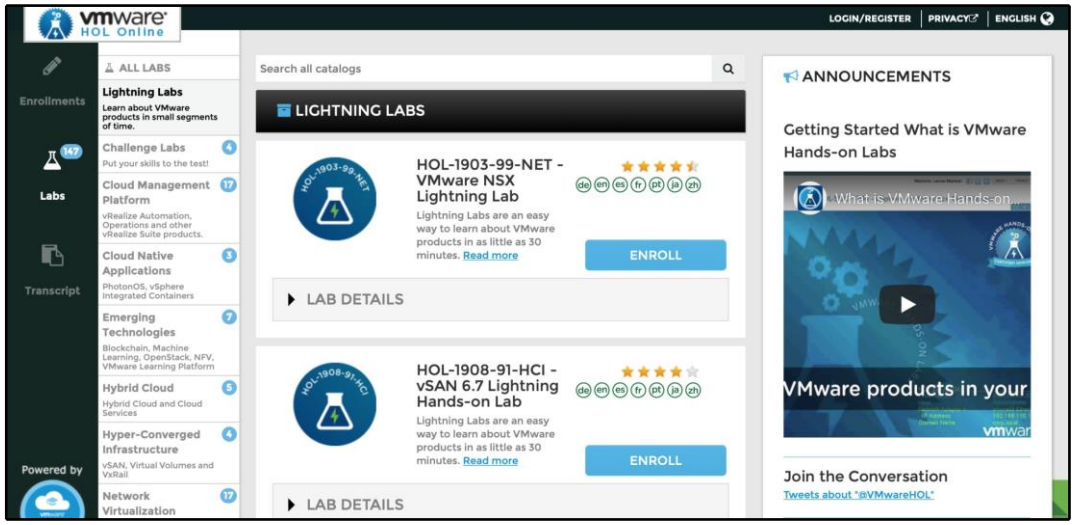
If you don't wish to run your own lab, there are other options. We'll take a look at a few of these in the following sections.

VMware Hands-On Lab (HOL)

If you need to test specific products without running an environment, you can check out **VMware Hands-On Labs** (**HOLs**), which can be found at the following link: `https://labs.hol.vmware.com/HOL/catalogs/catalog/1212`.

Here, you can find numerous instant-access labs that you can use to evaluate certain products. The nice thing about HOLs is that the infrastructure is ready within a few minutes and you get a comprehensive lab guide that will guide you through different scenarios. Moreover, it's free!

The VMware HOLs catalog looks as follows:



VMware forums

If you are looking for unofficial support, or you would like to discuss a particular situation within your VMware environment, you can also check out VMware Communities at the following link: `https://communities.vmware.com/welcome`. You can find numerous forums here based on the products with which you are facing some issues or difficulties.

The most useful forums include the following:

- **VMware vSphere**: `https://communities.vmware.com/community/vmtn/vsphere/content?filterID=contentstatus[published]~objecttype~objecttype[thread]`
- **vCenter Server**: `https://communities.vmware.com/community/vmtn/vcenter/content?filterID=contentstatus[published]~objecttype~objecttype[thread]`
- **VMware NSX:** `https://communities.vmware.com/community/vmtn/nsx`
- **VMware vSAN:** `https://communities.vmware.com/community/vmtn/vsan/content?filterID=contentstatus[published]~objecttype~objecttype[thread]`

Blogs

A lot of useful information can be found on different blogs. I would strongly recommend the following blogs so that you don't miss any new features or configuration walk-throughs.

There are two kinds of blogs:

- **Official**: Run by different VMware departments
- **Unofficial**: Run by freelancing VMware consultants or sometimes even by VMware employees

The following are official blogs:

- **VMware vSphere Blog**: `https://blogs.vmware.com/vsphere/`
- **VMware VROOM! Blog**: `https://blogs.vmware.com/performance/`
- **VMware PowerCLI Blog**: `https://blogs.vmware.com/powercli/`
- **Network Virtualization**: `https://blogs.vmware.com/networkvirtualization/`

A number of personal blogs you should check out are listed here:

- **Yellow Bricks**: `http://www.yellow-bricks.com`
- **virtuallyGhetto**: `https://www.virtuallyghetto.com`
- **CormacHogan**: `https://cormachogan.com`
- **ESXvirtualization**: `https://www.vladan.fr`
- **LearnVMware**: `https://learnvmware.online`
- **VirtualGeek**: `https://virtualgeek.typepad.com`
- **vNinja**: `https://vninja.net`
- **VMGuru**: `https://vmguru.com`

Choosing the right platform

A VMware lab can come in many forms. You can use standard rack servers installed in your basement, you can use your desktop PC, you can run several small PCs as an Intel **Next Unit of Computing** (**NUC**) to host your ESXi servers, you can rent a physical server from a number of service providers, or you can use a cloud environment to host your virtual ESXi hypervisors.

The aim is to find the sweet spot between cost and functionality. Of course, it would be nice to run your blade chassis with multiple physical servers and fiber-channel storage, but the cost of such a solution would probably be too high.

It is important to note that whatever your decision, in most cases, it won't be supported by VMware at all. You can't expect any support from the VMware support team and you will be responsible for any problems you have with your lab. In order to get official support, you need to have all hardware components listed in the **Hardware Compatibility List** (**HCL**). However, community support is still available. If you run into any problems, you can try asking a question on the public VMware forums at the following link: `https://communities.vmware.com/index.jspa`.

No matter what platform you choose, you should carefully plan the resources the platform will require and any additional hardware you might need. Think about the following questions:

- Which CPU and memory resources do you need?
- What is the size of the storage and what is its performance like?
- Do you need additional physical components, such as switches or storage arrays?
- Where will the lab be located?
- What will the energy consumption of your lab be like?

Another factor might be the time span of the lab or **proof of concept** (**PoC**):

- **Short-term**: You only need to run the lab for a few weeks or months to test the individual components or for exam preparation
- **Long-term**: You would like to run the lab for months or even years so that you have an environment that you can come back to whenever you need it

The duration of the lab project or PoC will, of course, affect the overall cost of the solution. Let's take a look at the different options for your lab and their pros and cons.

Standard rack servers

This kind of server is commonly used by enterprise companies in their data center. You might get lucky and find refurbished servers at a fraction of the original cost. Usually, when companies are refreshing their hardware infrastructure, they sell older servers.

The advantages of using a standard rack server are as follows:

- They are similar to production servers
- They have adequate compute resources
- They are easy to scale by adding more servers

The disadvantages of using a standard rack server are as follows:

- They are expensive
- They consume a lot of power (this is especially the case for older hardware)
- They require additional physical network infrastructure
- They are large in size
- They are noisy

You can find many refurbished servers on eBay at `https://www.ebay.com/b/Computer-Servers/11211/bn_886971` or on some specialized sites including `https://www.bargainhardware.co.uk/refurbished-servers`.

Desktop PC

For a small lab, even your home PC could be enough, but it depends on what you want to test. Do you need to run a single ESXi server? If so, a VMware Workstation will do the trick. Once you get serious, however, you might encounter problems with the resources of your home PC.

Today, traditional home PCs might have 16 GB of memory, with some more expensive configurations having up to 32 GB. To run a single vCenter Server appliance, you will need 10 GB of memory. If you require additional ESXi servers, nested virtual machines, vRealize Operations, or even NSX, you will reach the resource limits quite quickly.

The advantages of using a Desktop PC are as follows:

- They are cheap
- They are often already available
- There is no need for additional infrastructure

The disadvantage of using a Desktop PC is that you will often not have enough resources available for more complex setups.

Small, dedicated PCs

A lot of people use NUCs or similar platforms for their labs. In this case, you have several dedicated small servers, each usually equipped with 32 GB of memory and local SSD storage.

The advantages of using small, dedicated PCs are as follows:

- They are small
- They are easy to scale by adding more servers

The disadvantages of using small, dedicated PCs are as follows:

- They are expensive
- They require additional physical network infrastructure
- You will have limited upgrade options for CPU and memory

Cloud-based solutions

If you do not want to invest in any physical hardware, you might be interested in a cloud-based option. In this case, you are renting virtual resources like any other customer on the public cloud, but inside the virtual machine, a virtualized ESXi server is running.

This approach might be ideal for short-term projects, where you pay only for resources that you have used over a period of time. For a long-term project, however, the price might not be that attractive when compared to physical servers.

The advantages of using cloud-based solutions are as follows:

- They can be deployed within minutes
- They can be cheap for short-term projects
- They are software defined, so there is no need for any additional infrastructure

The disadvantages of using cloud-based solutions are as follows:

- They can be expensive for long-term projects
- The solution is delivered as is, and you can't customize some of the components

Note that you can't run ESXi inside a virtual machine. The service provider must explicitly support this function (nested virtualization). If you try to install an ESXi hypervisor on an Amazon AWS EC2 instance, for example, it won't work correctly. Several companies specialize in nested environments. These include Ravello (`https://cloud.oracle.com/en_US/ravello`) and VMlabs (`https://www.vmlabs.io`).

A dedicated server in a data center

In my opinion, this is the optimal solution. You can either host your server in the data center, or you can rent a server, depending on your requirements. When you are running your own dedicated server, you have complete control of the environment, and you can tweak the server as you need to.

In this situation, nested virtualization is used. First, the ESXi hypervisor is installed on the physical server and then you create several other virtual machines that will be used to host your virtual ESXi servers.

The advantages of using a dedicated server in a data center are as follows:

- You have complete control of the servers
- There is no need for any additional physical network infrastructure because everything is virtualized
- It can be cheap for short-term projects
- You will have sufficient compute resources

The disadvantage of using a dedicated server in a datacenter is that it can be cheap for long-term projects.

There are many service providers that you can use to rent a physical server. I tend to use servers from OVH (`https://www.ovh.com`).

Software components and licensing

Now, when you have chosen your ideal hardware platform, you need to think about which software licenses you need in order to install and run the environment successfully.

There are many components that you need to license, especially if you want to build everything from scratch. In some cases, cloud-based solutions do not need to be licensed, because the licenses are already included in the service price.

Let's cover the most common software components that you might need to use in your lab.

VMware licensing

Let's start with VMware itself. Most products can be tested for free with the evaluation version through the Eval center at the following link: `https://www.vmware.com/try-vmware.html`.

The following products can be downloaded as 60-day eval versions:

- **VMware vSphere**: `https://www.vmware.com/go/evaluate-vsphere-en`
- **VMware vSAN**: `https://www.vmware.com/go/try-vsan-dl-en`
- **VMware Horizon**: `https://www.vmware.com/go/try-horizon-view-dl-en`
- **vRealize Operations**: `https://www.vmware.com/go/try-vrealize-ops-dl-en`
- **vRealize Log Insight**: `https://www.vmware.com/go/try-log-insight`
- **Site Recovery Manager**: `https://www.vmware.com/go/try-srm`

However, some products can't be downloaded. VMware NSX and vRealize Automation, for example, are no longer available for download.

For some VMware products, you can test solutions already deployed for you using the cloud offerings located at the following link: `https://cloud.vmware.com`.

VMware EVALExperience

This is an excellent program that you can join through **VMware User Group** (**VMUG**) membership. Standard VMUG membership is free, and it gives you several benefits, which you can check at the following link: `https://www.vmug.com`.

There is a paid membership as well, called VMUG Advantage, which allows you to access the following additional benefits:

- EVALExperience
- 20% discount on VMware training classes
- 20% discount on VMware certification exams
- 35% discount on VMware certification exam preparation workshops (VCP-NV)

- 35% discount on VMware lab connect
- $100 discount on VMworld attendance
- Extended trials of VMware cloud services

Membership of the advantage program costs $200 per year, but it is worth it. The most exciting offering is the EVALExperience program. VMware's EVALExperience gives you exclusive access to 365-day evaluation licenses for a selection of VMware solutions, for personal use in a non-production environment. It includes the following products:

- VMware vCenter Server v6.x Standard
- VMware vSphere ESXi Enterprise Plus with Operations Management (six CPU licenses)
- VMware NSX Enterprise Edition (six CPU licenses)
- VMware vRealize Network Insight
- VMware vSAN
- VMware Site Recovery Manager
- VMware vRealize Log Insight
- VMware vRealize Operations
- VMware vRealize Automation 7.3 Enterprise
- VMware vRealize Orchestrator
- VMware vCloud Suite Standard
- VMware Horizon Advanced Edition
- VMware vRealize Operations for Horizon
- VMware Fusion Pro 11
- VMware Workstation Pro 15

For the price of $200, you get the licenses for all vSphere products you might ever need, which is an impressive offering. Plus, it saves you having to reinstall the environment every 60 days!

If you would like to join the program, visit the following link: `https://www.vmug.com/Join/EVALExperience`.

Windows licensing

You will probably use Windows Servers within your inventory as well as some Microsoft applications such as the SQL database. Fortunately, Microsoft has an eval program as well. This allows you to run any software component for 180 days to test it. You can download different software at the following link: `https://www.microsoft.com/en-us/evalcenter/`.

The following two components are likely to be the most interesting:

- **Windows Server:** `https://www.microsoft.com/en-us/evalcenter/evaluate-windows-server-2019`
- **SQL Server:** `https://www.microsoft.com/en-us/evalcenter/evaluate-sql-server-2017-rtm`

Note that you can download the older versions as well, so if you prefer to stick with Windows Server 2012 R2, you can still download it from the Evaluation Center.

Other software components

Once you have obtained the core licenses, it is time to start thinking about the whole environment, including how you will interconnect the core components and which additional licenses or software you might need.

Storage

Which storage solutions will you use? There are many options for how to provide storage to your ESXi infrastructure:

- Windows Server with file services (iSCSI target)
- Open source Linux-based storage solutions such as **FreeNAS** (`https://www.freenas.org`)
- Software-defined storage solutions such as HPE VSA or EMC ScaleIO

Networking

Depending on how your infrastructure will be interconnected with the outside network, you might need to run some firewall appliances that provide access to the lab, **Network Address Translation** (**NAT**) and routing features, or even to the VPN connection.

Many vendors offer either free products or at least evaluation versions of virtual appliances:

- **Cisco Cloud Services Router 1000V Series**: `https://www.cisco.com/c/en/us/products/routers/cloud-services-router-1000v-series/index.html`
- **Juniper vSRX Virtual Firewall**: `https://www.juniper.net/us/en/products-services/security/srx-series/vsrx/`
- **VyOS**: `https://vyos.io`
- **MikroTik RouterOS**: `https://mikrotik.com/software`

You might even try to simulate complex physical networks to test some advanced networking features such as leaf-spine design or VXLANs. To do this, you will need to deploy a specific virtual machine that will act as a switch for your environment.

Cumulus Linux is one of the most frequently deployed network operating systems that can be installed within the virtual machine as well. This can be found at the following link: `https://cumulusnetworks.com/products/cumulus-linux/`.

Architecture and logical design

You might have realized by now that building a lab isn't as simple as you might think. For those who are interested in how I built the lab that I used during the writing of this book, you can follow this detailed guide.

I chose to rent a dedicated server from OVH. This was for the following reasons:

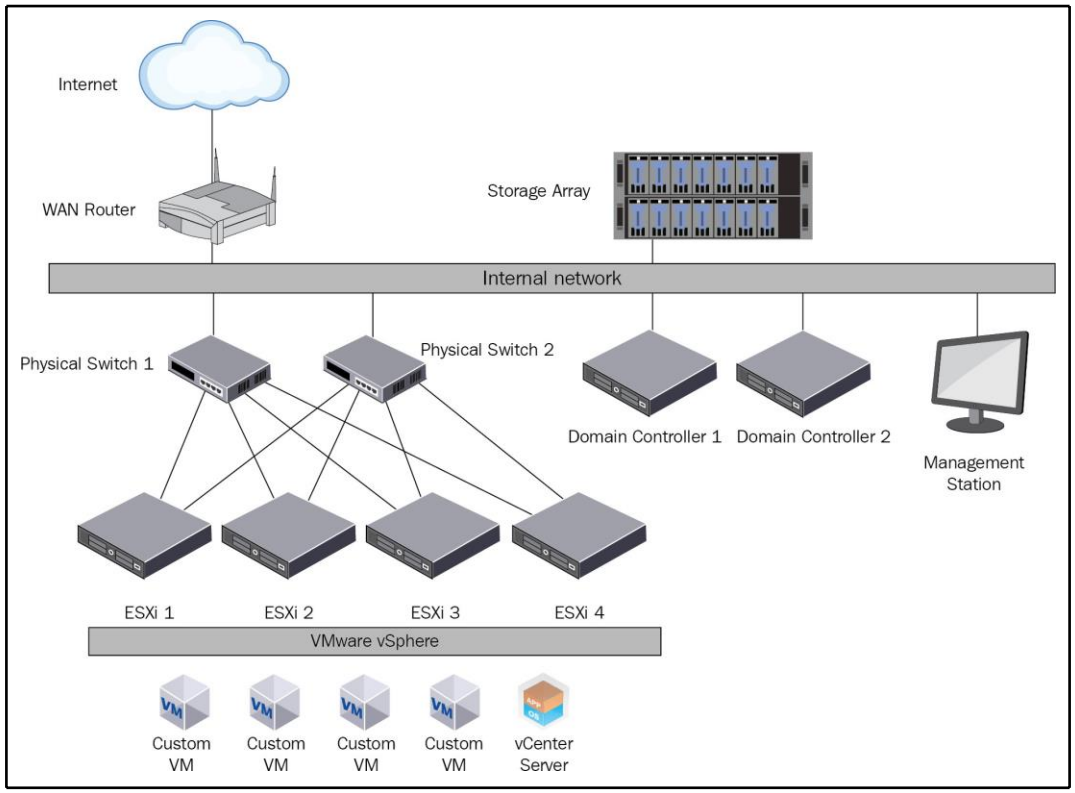
- For my on-site training, I needed a lot of resources (256 GB of RAM or more)
- I did not need to run the server all the time
- I did not want to pay any setup fees
- I needed to be able to order as many public IP addresses as necessary for the project
- There were months when I didn't need to run the lab at all
- There were other months in which I needed a different hardware platform with different resources
- I needed to access the lab from anywhere

Based on these requirements, I decided that it did not make sense to use my hardware because the costs would have been too high. The cloud solution did not work for me either, because I needed to have complete access to the environment and I needed to be able to tweak it as much as necessary.

Let's take a look at the architecture of my lab and its logical design.

The architecture of the lab

I used a single physical server. The first ESXi server is installed on the top of the server. I call the ESXi hypervisor **MasterESXi**. This MasterESXi then hosts multiple VMs, some of which are used to provide infrastructure services, such as DNS, **Active Directory** (**AD**) services, or iSCSI storage, to the lab. Other VMs are used as nested ESXi hypervisors. Let's take a look at the overall architecture:



Let's now take a look at the different components of the lab.

The Master ESXi hypervisor

This is an ESXi hypervisor running on the bare metal server in the data center. There is no shared storage at all, only local storage that is used for virtual machines. This ESXi hypervisor is used to host all virtual infrastructure machines, such as the **Active Directory Domain Controller** (**AD DC**), the management station, the iSCSI server, and the virtual router. All virtual ESXi servers are running on top of the Master ESXi server.

iSCSI storage

As you already know, we need to provide shared storage to our ESXi servers if we want to use different cluster features such as **HA** or **DRS**. As a result, I have a virtual machine that hosts the iSCSI target service. All virtual ESXi servers are connected to that storage over iSCSI.

Virtual router

You need to provide management access to the environment. You also need to configure the routing between different subnets within your lab. My virtual router supports VLANs, so I can quickly test multiple port groups with different VLAN tags. You can also use a virtual router to provide services such as a DHCP server or NAT.

Management station

I prefer to install all necessary software on a management station (or a jump-host server if you prefer) so that I do not need to install any software directly to my laptop or home PC. Another reason for using a management station is that it is run within the environment and you are accessing the station over RDP remotely. This means that the interconnection within the lab is much faster than over the internet.

AD

In many enterprise environments, there is an AD in place. It would be good to test all the integrations between your vSphere environment and AD. Also, if you need to test different roles and permissions, it is useful to test these against AD (or any other LDAP server) instead of the local SSO domain.

IP address plan

In the lab, there will be multiple IP subnets and port groups. Let's take a look at the different sections of the IP address plan. As a DNS server, use the IP addresses of the primary and secondary domain controllers. In my case, these are `172.16.1.1` and `172.16.1.2`.

Management network

The management network is used for management communication between different components:

Management network	172.16.1.0/24	IP address	DNS name
vSphere infrastructure			
	ESXi-prod-1	172.16.1.11	esxi-prod-1.learnvmware.local
	ESXi-prod-2	172.16.1.12	esxi-prod-2.learnvmware.local
	ESXi-prod-3	172.16.1.13	esxi-prod-3.learnvmware.local
	ESXi-prod-4	172.16.1.14	esxi-prod-4.learnvmware.local
	vCenter Server	172.16.1.100	vcsa.learnvmware.local
Internal Infrastructure			
	Domain Controller 1	172.16.1.1	dc1.learnvmware.local
	Domain Controller 2	172.16.1.2	dc2.learnvmware.local
	Management workstation	172.16.1.250	mgmt.learnvmware.local
Network infrastructure			
	Default gateway	172.16.1.254	

vMotion network

This is the dedicated network for vMotion:

vMotion network	172.16.2.0/24	IP address
VMware vSphere		
	ESXi-prod-1	172.16.2.1
	ESXi-prod-2	172.16.2.2
	ESXi-prod-3	172.16.2.3
	ESXi-prod-4	172.16.2.4

iSCSI network

This is the dedicated network for iSCSI traffic between ESXi hypervisors and the shared storage array:

iSCSI network	192.168.0.0/16	IP address
VMware vSphere	ESXi1 iSCSI1	192.168.100.11
	ESXi1 iSCSI2	192.168.100.12
	ESXi2 iSCSI1	192.168.100.21
	ESXi2 iSCSI2	192.168.100.22
	ESXi3 iSCSI1	192.168.100.31
	ESXi3 iSCSI2	192.168.100.32
	ESXi4 iSCSI1	192.168.100.41
	ESXi4 iSCSI2	192.168.100.42
	Storage SP1	192.168.10.1
	Storage SP2	192.168.10.2

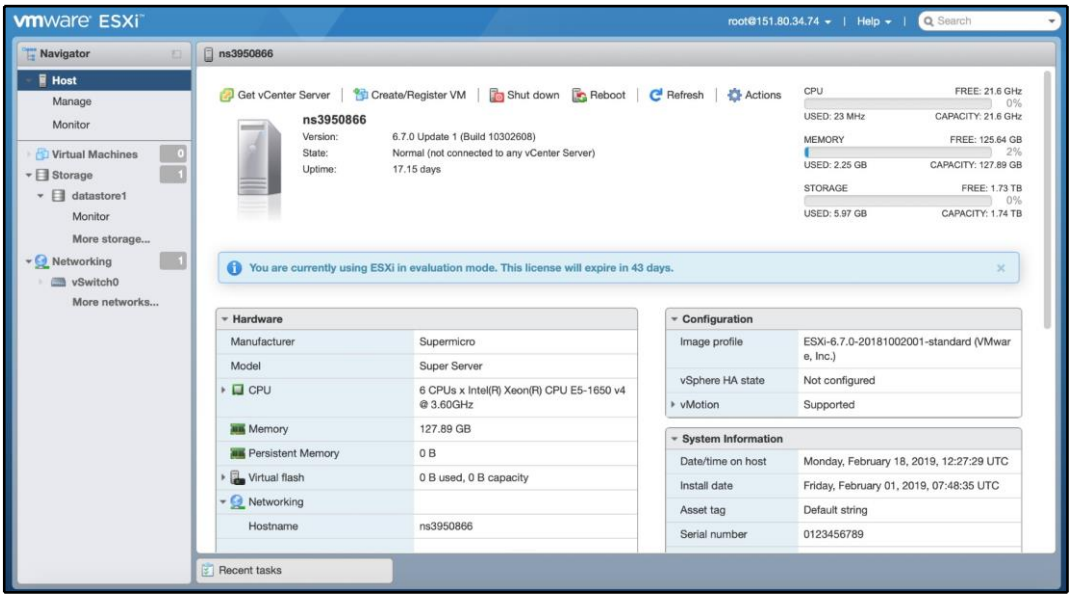
Production network

This is the simulation of the production network. Different VLANs are used in the production network to test network connectivity:

Production network 1 - VLAN 10		
	VMs	10.0.10.1-253
	GW	10.0.10.254
Production network 2 - VLAN 20		
	VMs	10.0.20.1-253
	GW	10.0.20.254
Production network 3 - VLAN 30		
	VMs	10.0.30.1-253
	GW	10.0.30.254

A detailed implementation guide

In this example, I am using a dedicated server provided by `OVH.com`. As mentioned previously, however, you can use any server you like. At this stage, I have a bare metal server and a fresh copy of ESXi 6.7 U1 installed on the top of the server:

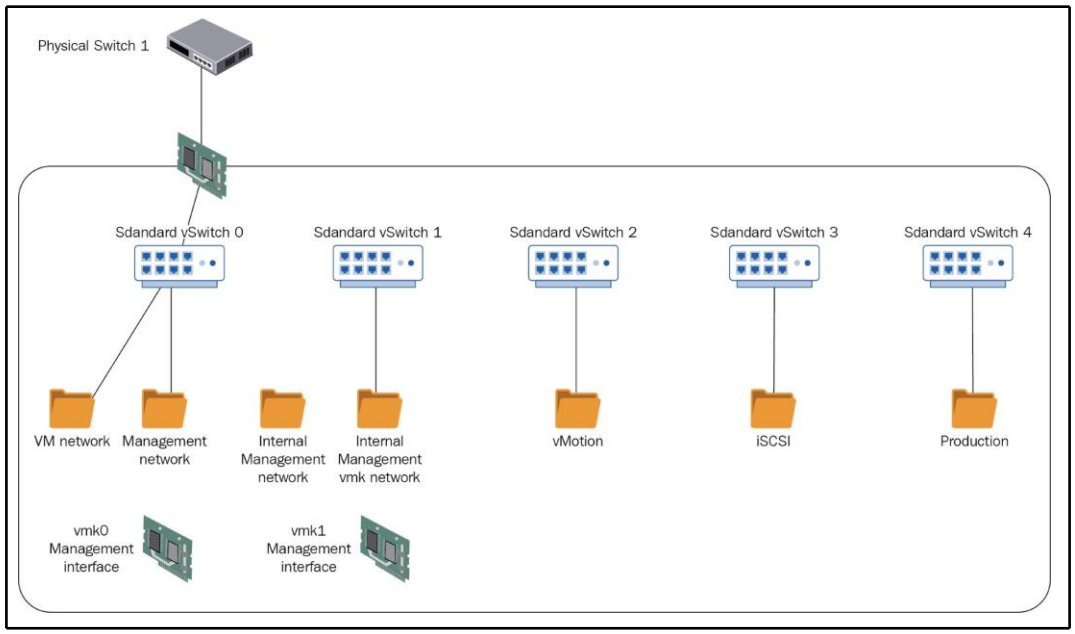


Master ESXi server configuration

First, we need to configure our Master ESXi server so that it will be possible to host our virtual machines. Before we do that, we need to start with the network configuration of the Master ESXi server.

Network configuration

Let's have a look at the network topology of the Master ESXi hypervisor first:

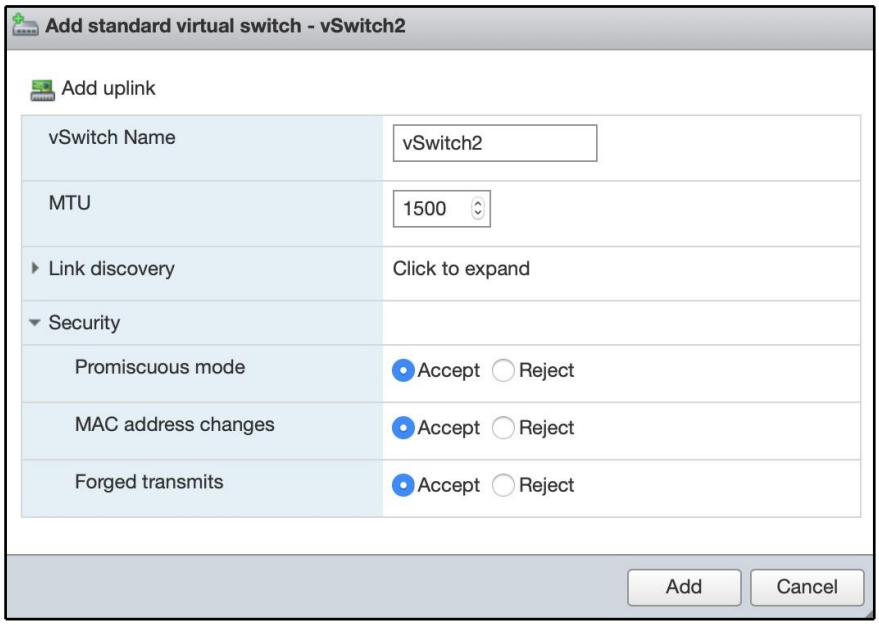


Virtual switches

As a first step, we need to define our virtual switches. The following vSwitches will be created:

vSwitch name	Uplinks	MTU	Description
`vSwitch1`	No	1500	Used for management of the network
`vSwitch2`	No	1500	Used for vMotion
`vSwitch3`	No	9000	Used for iSCSI
`vSwitch4`	No	1500	Used for the production network

Because we will be running nested virtualization, all security policies must be set to **Accept**:



You can check the following article, which explains why the security policy must be changed: `https://www.virtuallyghetto.com/2013/11/why-is-promiscuous-mode-forged.html`.

Port groups

Next, we need to define port groups that will be used to connect our virtual machines:

Name	vSwitch	VLAN ID	Notes
Internal management	vSwitch1	0	
Internal management—vmk	vSwitch1	0	For vmk1
vMotion	vSwitch2	0	
iSCSI	vSwitch3	0	
Production	vSwitch4	4095	ESG trunking must be enabled with VLAN 4095

On the Master ESXi level, we need one additional VMkernel port that will be used for management from our management station:

Name	Port group	IP address	Services
vmk1	Internal management—vmk	172.16.1.253	Management

Virtual machines

Now, when we have configured the networking of the Master ESXi server, let's create several virtual machines that will be used for our lab. In my case, I am running all VMs with a thin disc. Of course, your resource configuration might be different, so take this as an example only:

VM name	CPU	RAM	HDD	OS	Notes
DC01.learnvmware.local	1	2	20	Windows	Domain controller 1
DC02.learnvmware.local	1	2	20	Windows	Domain controller 1
Mgmt.learnvmware.local	2	4	40	Windows	Management station
iSCSI	4	4	20	Windows	iSCSI storage array
Esxi-prod-1.learnvmware.local	2	12	10	ESXi	vESXi1
Esxi-prod-2.learnvmware.local	2	12	10	ESXi	vESXi2
Esxi-prod-3.learnvmware.local	2	12	10	ESXi	vESXi3
Esxi-prod-4.learnvmware.local	2	12	10	ESXi	vESXi4

Let's take a look at the network configuration of the virtual machines:

Virtual machine	Interface type	Port group
DC01	E1001	Internal management
DC02	E1001	Internal management
Mgmt	E1001	Internal management
iSCSI	E1001	Internal management
iSCSI	VMXNET3	iSCSI
iSCSI	VMXNET3	iSCSI
ESXi	VMXNET3	Internal management
ESXi	VMXNET3	Internal management
ESXi	VMXNET3	vMotion
ESXi	VMXNET3	vMotion
ESXi	VMXNET3	iSCSI
ESXi	VMXNET3	iSCSI
ESXi	VMXNET3	Production
ESXi	VMXNET3	Production

For all Windows-based virtual machines, install Windows Server 2012 R2 as a base OS, configure the computer name and the IP address, enable remote desktop services, and change the firewall settings if necessary. Also, install VMtools to the guestOS.

For ESXi-based virtual machines, install vSphere 6.7U1.

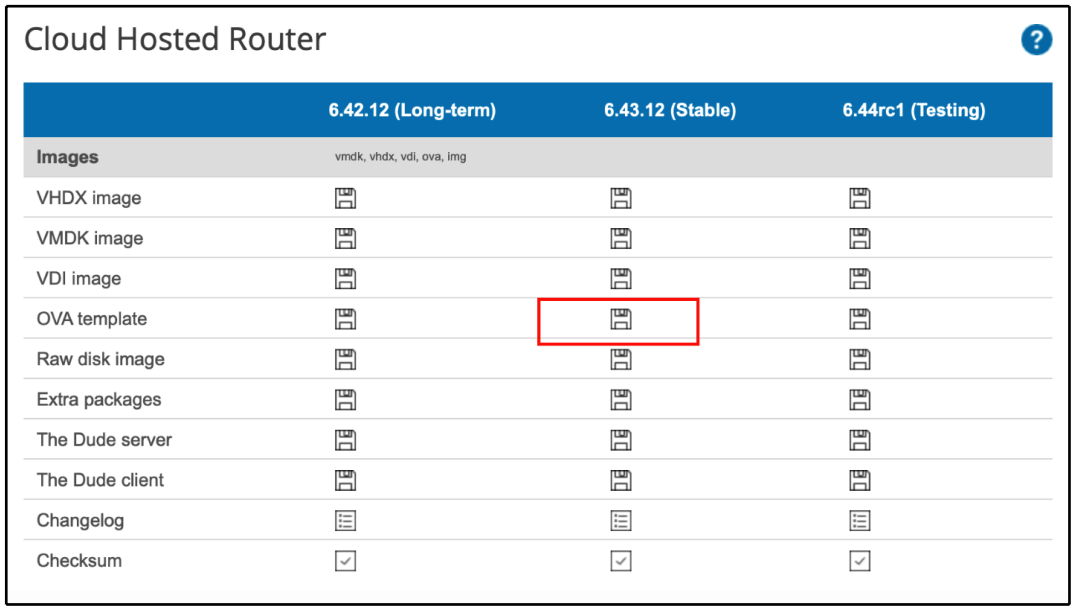
Note that at this stage, the network connectivity is still unavailable to the external network. Only connections inside the internal management network will work.

Virtual router

As a virtual router, I am using **RouterOS** from MikroTik, but feel free to install any vRouter with which you have hands-on experience.

RouterOS can be downloaded from `https://mikrotik.com/download`. Once you register your account on `Mikrotik.com`, you can obtain a free 60-day evaluation license.

Let's deploy the OVA package from MikroTik. There is a specialized appliance type called **Cloud Hosted Router** that is built to be run in virtualized environments:



Once you have deployed the virtual router, all you need to do is to assign more virtual network adapters:

vNIC ID	Port group
Network adapter 1	VM network
Network adapter 2	Internal management
Network adapter 3	Production

Virtual router configuration

Before we can communicate between multiple IP subnets, we need to configure our virtual router correctly:

Interface name	IP address	Port group
Eth1	`X.X.X.X`	VM network (public IP address from the service provider)
Eth2	`172.16.1.254/24`	Internal management

At this stage, we can use any GUI to configure our router. We do this using CLI. In my example, you can see the configuration of the RouterOS as follows:

```
ip address add address=X.X.X.X/X interface=ether1
ip address add address=172.16.1.254/24 interface=ether2
```

For the default gateway, you can use the following command:

```
ip route add address=0.0.0.0/0 dst-address=Y.Y.Y.Y
```

In these commands, `X.X.X.X` is the public IP address from the service provider, and `Y.Y.Y.Y` is the gateway that you have been assigned.

If you have configured everything correctly, you should now be able to connect to the GUI configuration interface of the RouterOS using the public IP address:

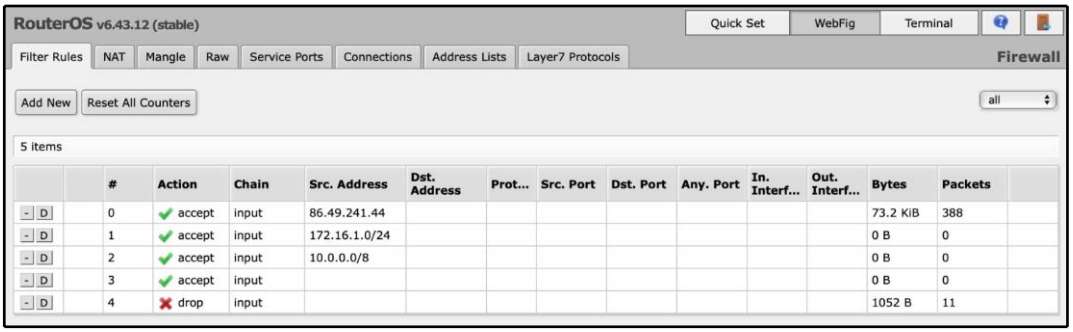
Firewalls and access to the virtual router

RouterOS, by default, does not use a password for the admin user. The first thing you should do is to change the blank password to something else. From the GUI menu, select **System** | **Users** and select the default admin user and change its password.

Then, it is recommended to limit the connection to the management interface of the RouterOS using firewall rules. Switch to **IP** | **Firewall** from the menu and define the following firewall rules:

ID	Chain	Source address	Action	Notes
0	Input	`z.z.z.z`	Permit	Your home IP address, so you can connect to the virtual router
1	Input	`172.16.1.0/24`	Permit	Allows connectivity from the management network
2	Input	`10.0.0.0/8`	Permit	Allows connectivity from the production network
3	Input		Permit	Check only the **related** and **established** options in the **connection state**
4	Input		Drop	Drop anything that is not permitted

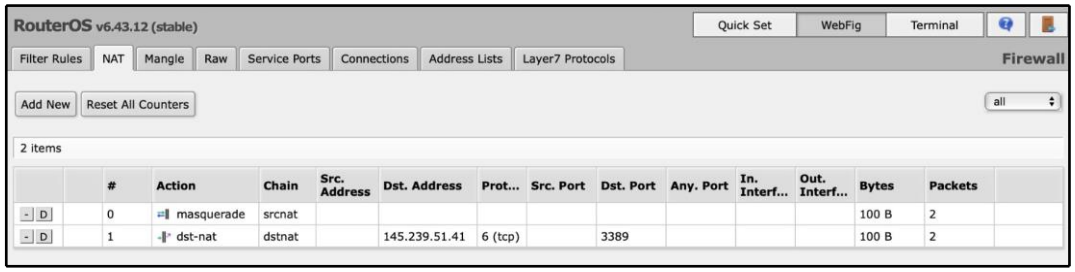
These rules are shown in the following screenshot:



Next, we need to configure NAT so that our virtual machines will be able to connect to the internet and so that we can connect to our management station. Two rules should be defined in the **NAT** tab:

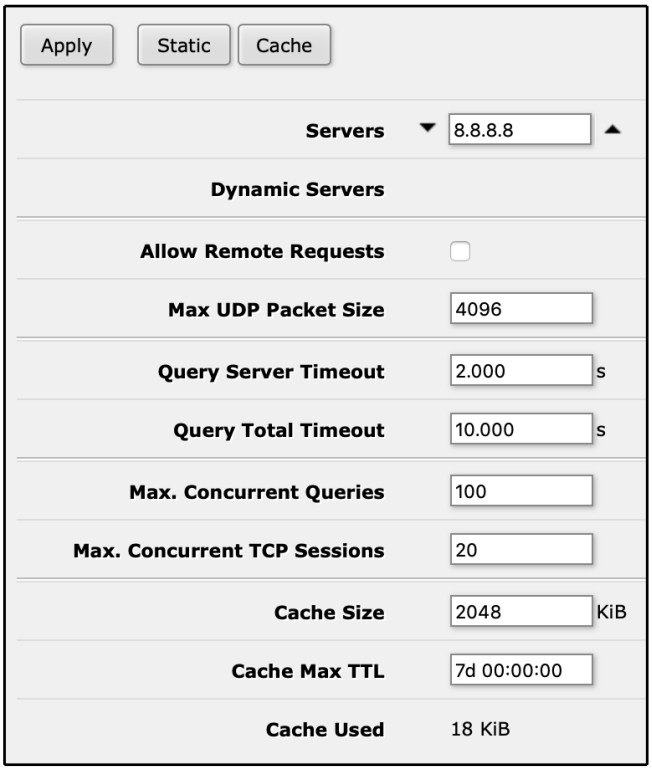
ID	Chain	Action type	Notes
0	`srcnat`	`masquerade`	
1	`dstnat`	`dst-nat`	Fill in the destination address (the public IP of the virtual router) and port `3389`. In **Action**, select `dst-nat`. The **To Address** value will be `172.16.1.250` and the **To Port** value should be `3389`.

The following screenshot contains the NAT rules:



DNS configuration

Our virtual router must be able to translate DNS records, so we need to configure the DNS servers. To do that, select **IP** | **DNS** and fill in your favorite DNS servers in the **Servers** field:

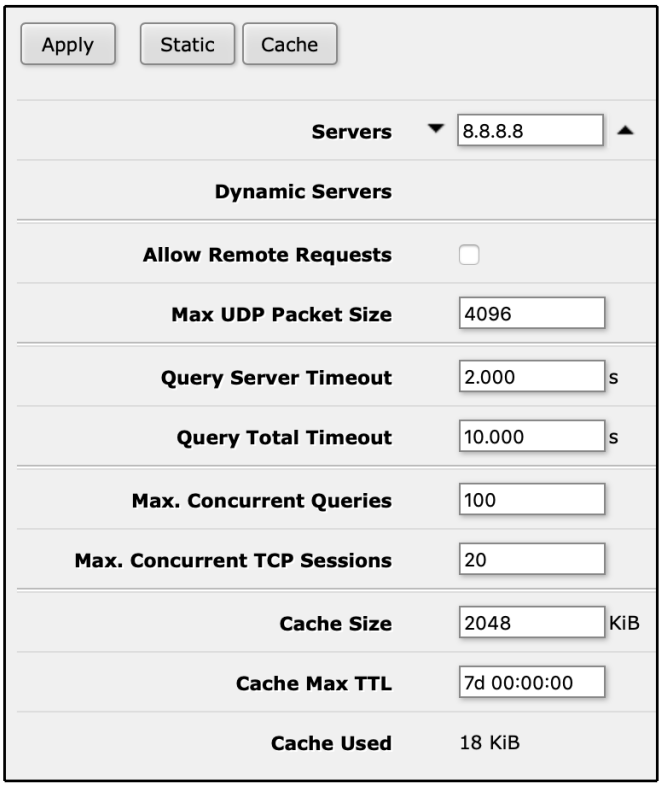


License configuration

By default, RouterOS runs with an evaluation version that is capped at 1 Mbps. You can obtain a 1 Gbps license for free; all you need to do is specify the username and password that you created during registration at `Mikrotik.com`.

Switch to **System** | **License** and click **Renew License**. You need to fill in your username and password and the desired license type. **P1** is a 1 Gbps license, **P10** is a 10 Gbps license, and **P unlimited** has no cap at all.

Once you have configured the username and password, you should see that you have successfully obtained a trial license:



VLAN configuration

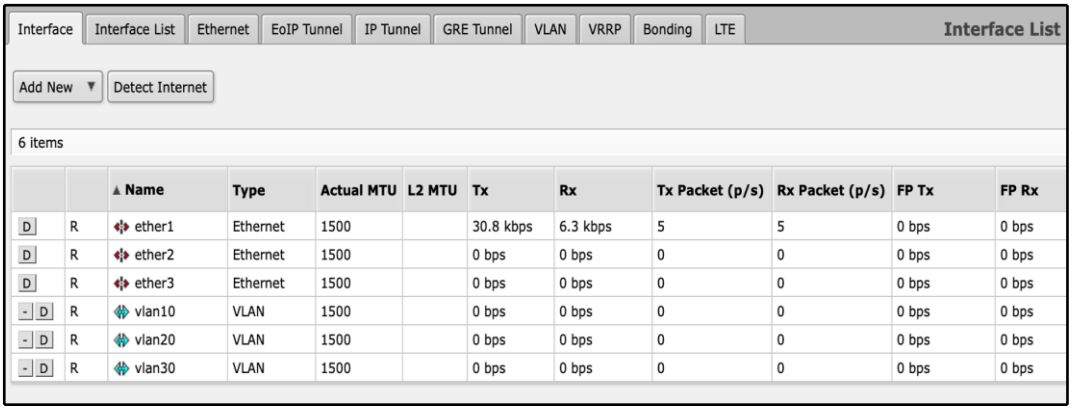
We then need to configure our virtual router to support VLANs. Switch to **Interface** from the main menu and select **New VLAN interface.** Three VLANs need to be created with the following properties:

- The VLAN IDs are `10` `20`, and `30`
- The interface is `ether2`
- The names are `vlan10`, `vlan20`, and `vlan30`

You can also use the following CLI command:

```
interface vlan add interface=ether3 vlan-id=10 name=VLAN10
```

The interface configuration should appear as follows:



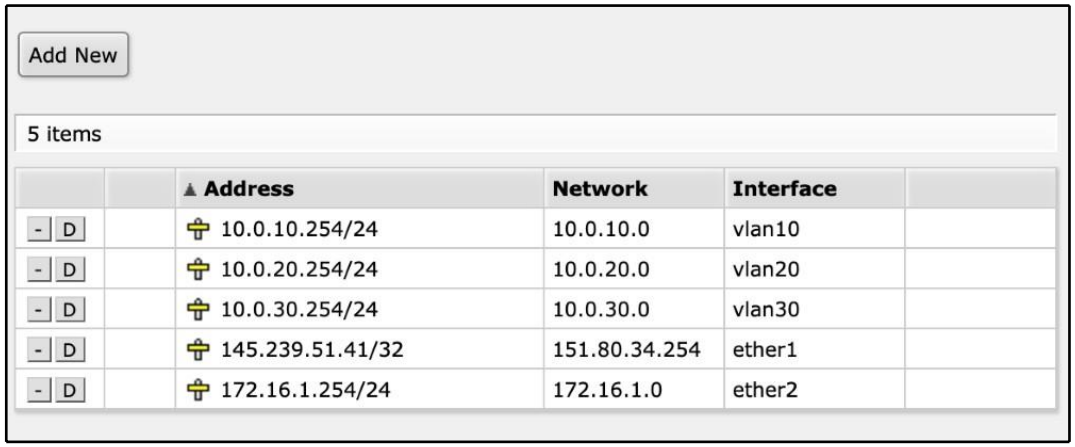
		Name	Type	Actual MTU	L2 MTU	Tx	Rx	Tx Packet (p/s)	Rx Packet (p/s)	FP Tx	FP Rx
D	R	ether1	Ethernet	1500		30.8 kbps	6.3 kbps	5	5	0 bps	0 bps
D	R	ether2	Ethernet	1500		0 bps	0 bps	0	0	0 bps	0 bps
D	R	ether3	Ethernet	1500		0 bps	0 bps	0	0	0 bps	0 bps
- D	R	vlan10	VLAN	1500		0 bps	0 bps	0	0	0 bps	0 bps
- D	R	vlan20	VLAN	1500		0 bps	0 bps	0	0	0 bps	0 bps
- D	R	vlan30	VLAN	1500		0 bps	0 bps	0	0	0 bps	0 bps

We now need to configure an L3 interface for the new VLANs so that the virtual machines that are connected to them will be able to reach the IP interface of the virtual router.

To do this, switch to **IP** | **Addresses** from the main menu and add the following three IP addresses:

IP address	Interface
10.0.10.254/24	vlan10
10.0.20.254/24	vlan20
10.0.30.254/24	vlan30

The IP configuration should appear as follows:



Windows infrastructure

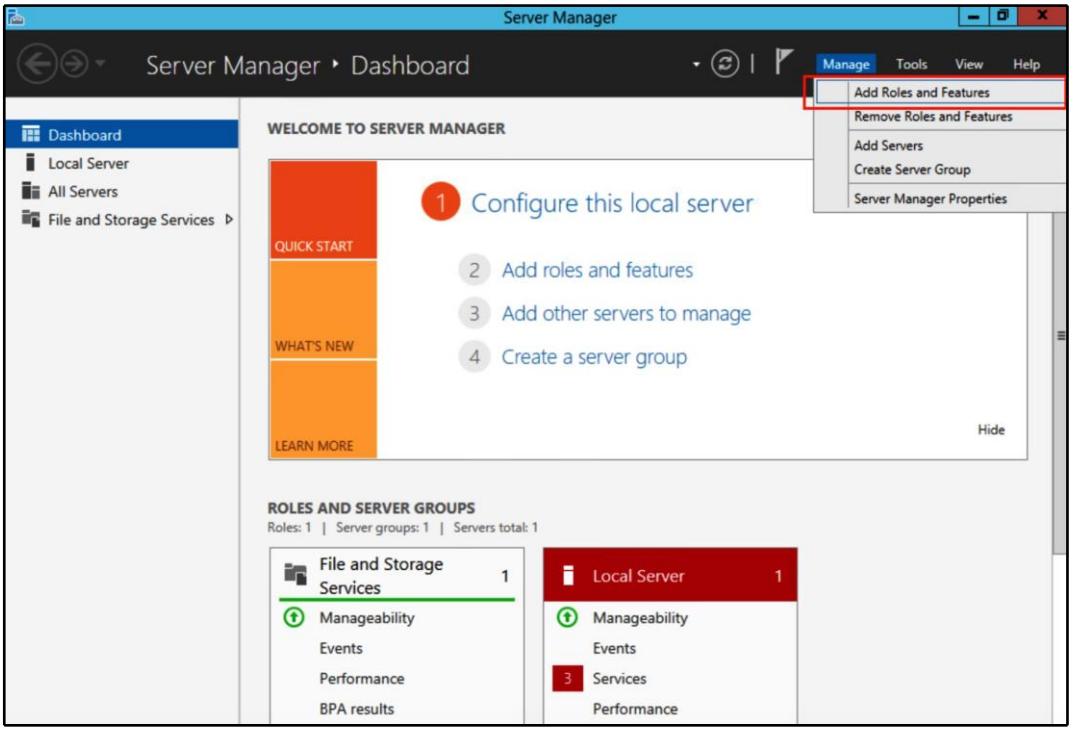
We have two new virtual machines that will be used as domain controllers, so let's have a look at the configuration of the Windows environment.

Once you have installed the guestOS and VMtools, configured the IP address, and changed the computer name, proceed with the AD installation.

DC01.learnvmware.local

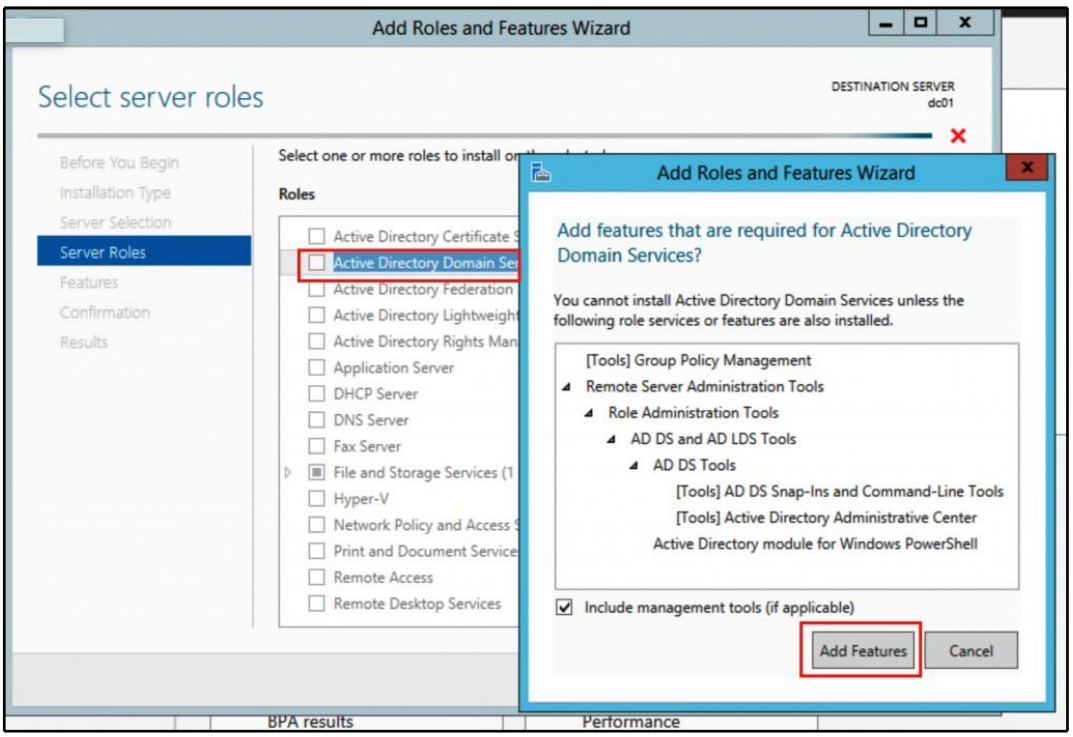
This will be our first AD DC. The installation of AD is quite a straightforward process. All you need to do is install the AD server role on the server.

From **Server Manager**, click **Manage,** and then select **Add Roles and Features**:



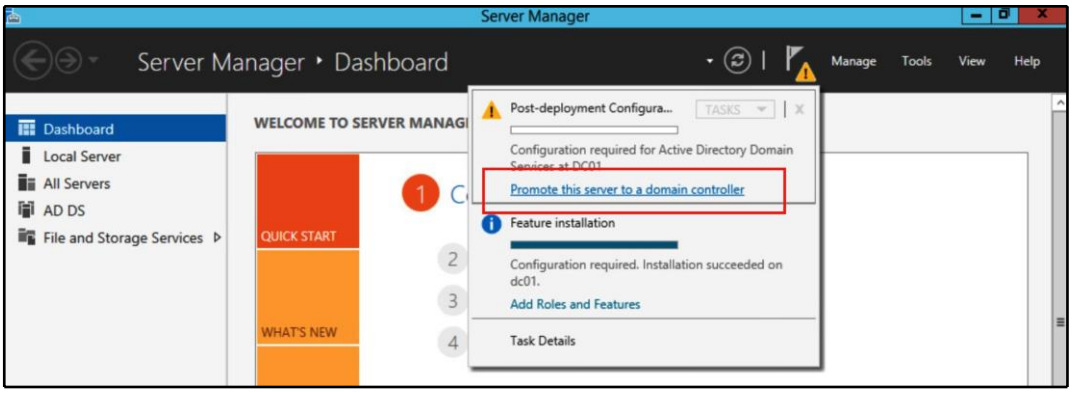
In the installation type, select role-based installation:

1. Select your server (`dc01`).
2. Select **Active Directory Domain Services Role** and add the required features:

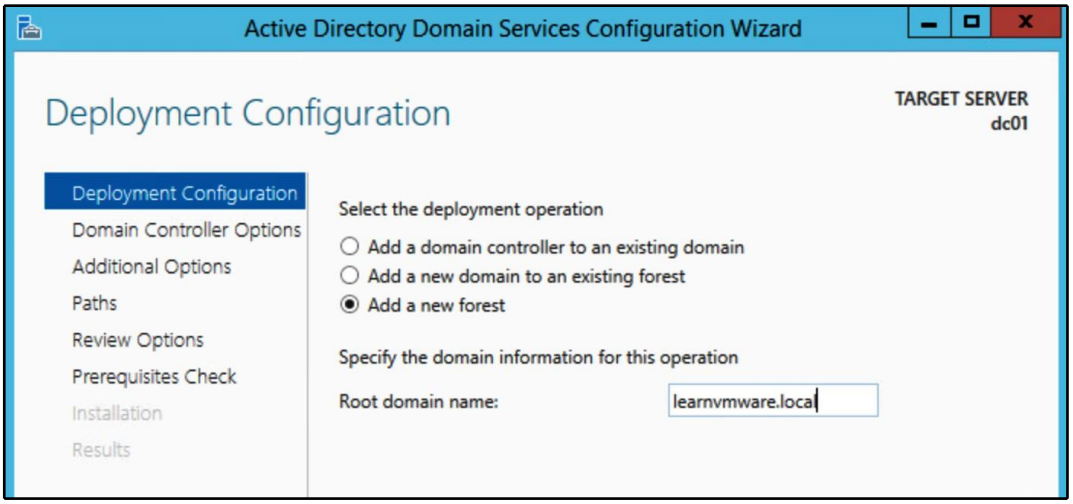


There is no need to select any additional features, so let's finish the wizard. It will take some time to install the required files, and once everything is installed, you can proceed with the AD configuration.

To do that, all you need to do is to click on **Promote this server to a domain controller**:



In **Deployment Configuration**, select **Add a new forest** and specify your local active directory domain. In this example, this is `learnvmware.local`:



In the next step, do not change anything except the **Directory Services Recovery Mode** (**DSRM**) password, which is a required field.

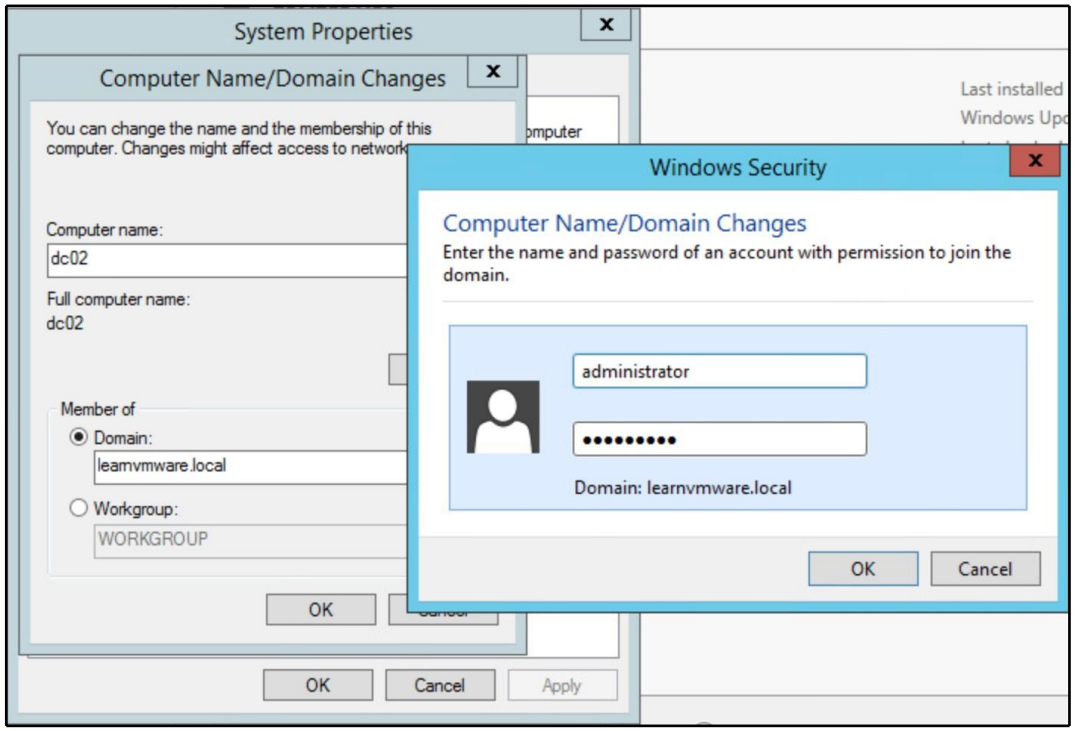
It is beyond the scope of this guide to cover AD itself, so let's proceed with the installation and keep all the defaults as the wizard suggests.

Once the installation is over, the server will automatically reboot to finish the **Active Directory Federation Services** (**ADFS**) installation.

DC02.learnvmware.local

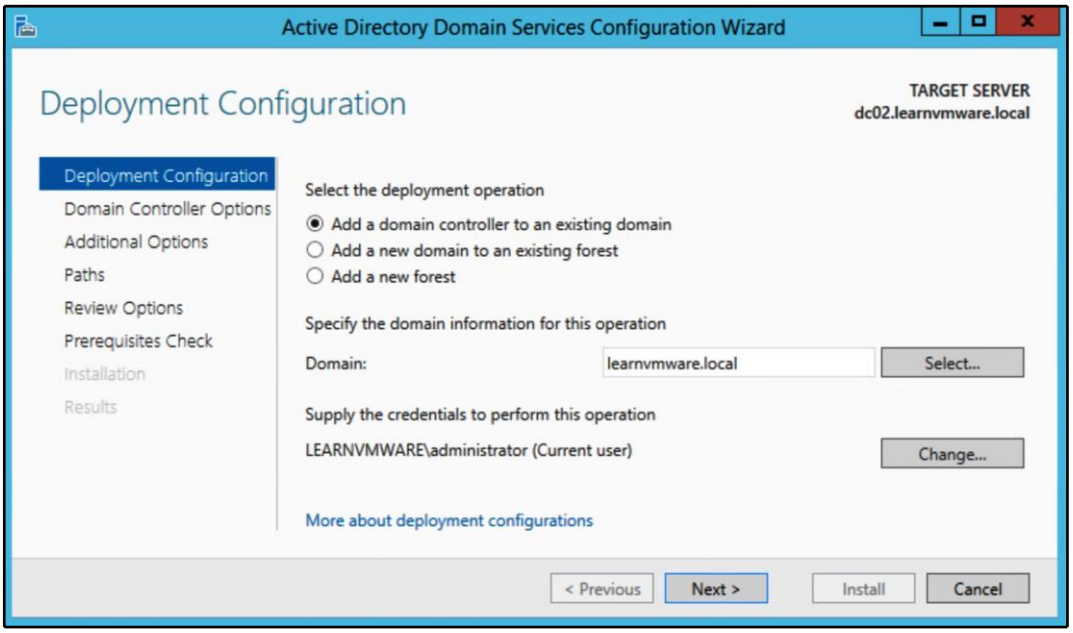
With our secondary domain controller, the process will be slightly different. As a first step, we will join this computer to our new AD domain. To do this, click on Configure this local server and then click on the **Workgroup** link. Then, follow the instructions as follows:

1. Click the **Change** button.
2. Select the domain and fill in your domain name as configured. You will need to provide an AD username and password. This would be the password you configured when you were installing Windows Server on `dc01` in the first place:



3. Reboot the server. After the server is rebooted, do not forget to log in as a domain administrator: `DOMAINAME\administrator`.

4. Now, you need to install the ADFS role as described in the *DC01.learnvmware.local* section. The only difference would be that during the ADFS configuration, you will select **Add a domain controller to an existing domain**:



Finish the installation wizard and reboot the server.

Congratulations ! You have successfully deployed your AD infrastructure! It's now time to move on to other servers.

Mgmt.learnvmware.local

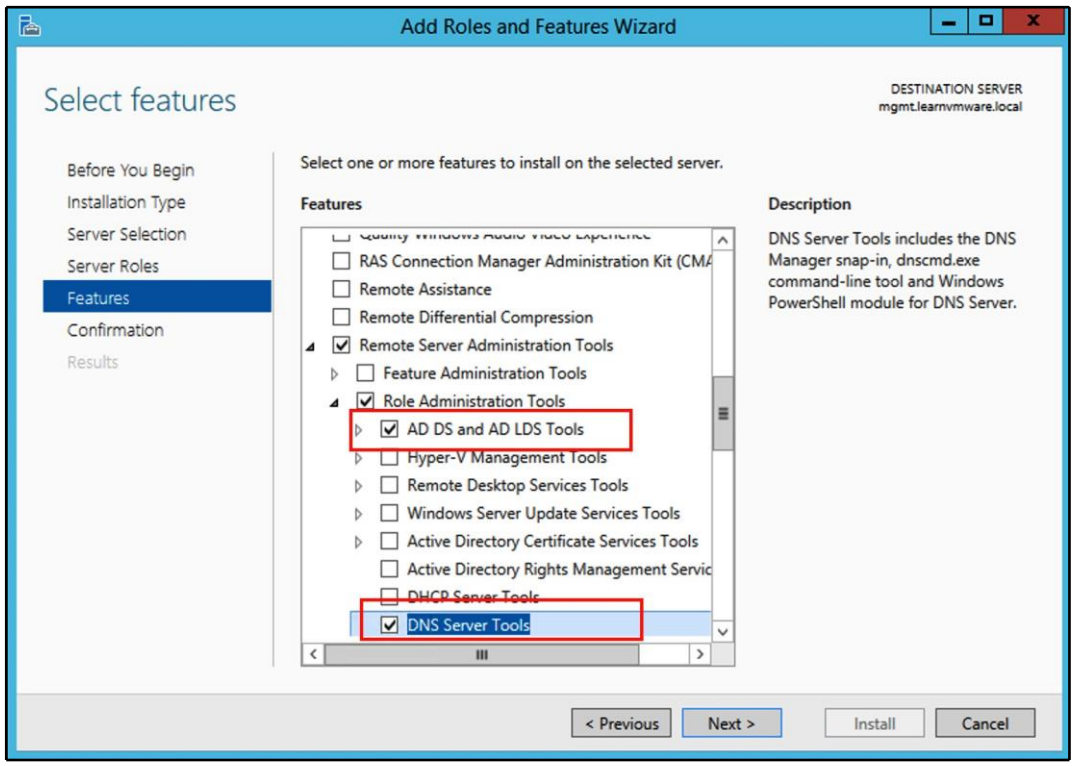
On this server, all we need to do is to join this computer to the AD domain as described in the *DC02.learnvmware.local* section.

This server will be used as a jump-host to the lab. Because we have already configured the NAT, you should be able to connect to the server using the remote desktop protocol.

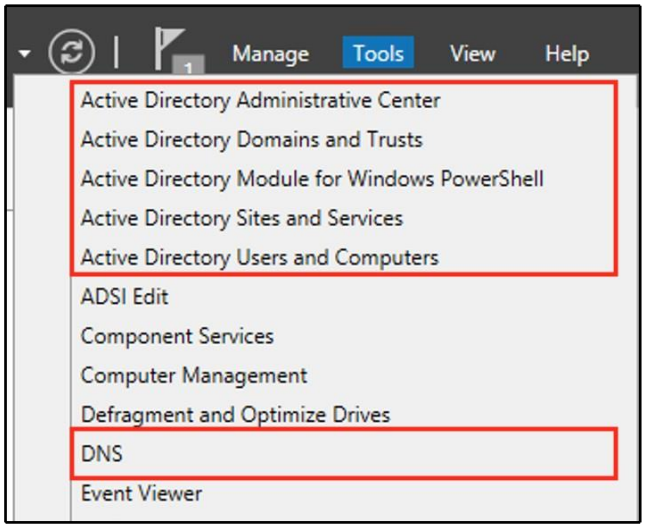
We would now like to install several remote management tools on the server so that we can connect to the AD directly from our management server:

1. Again, click on **Manage** and select Add or Remove features.
2. Do not select any role to install.

As the feature to install, select the **AD DS and AD LDS Tools** and **DNS Server Tools** under **Remote Server Administration Tools**:



Once installed, you should see the new management snap-ins in the **Tools** menu:



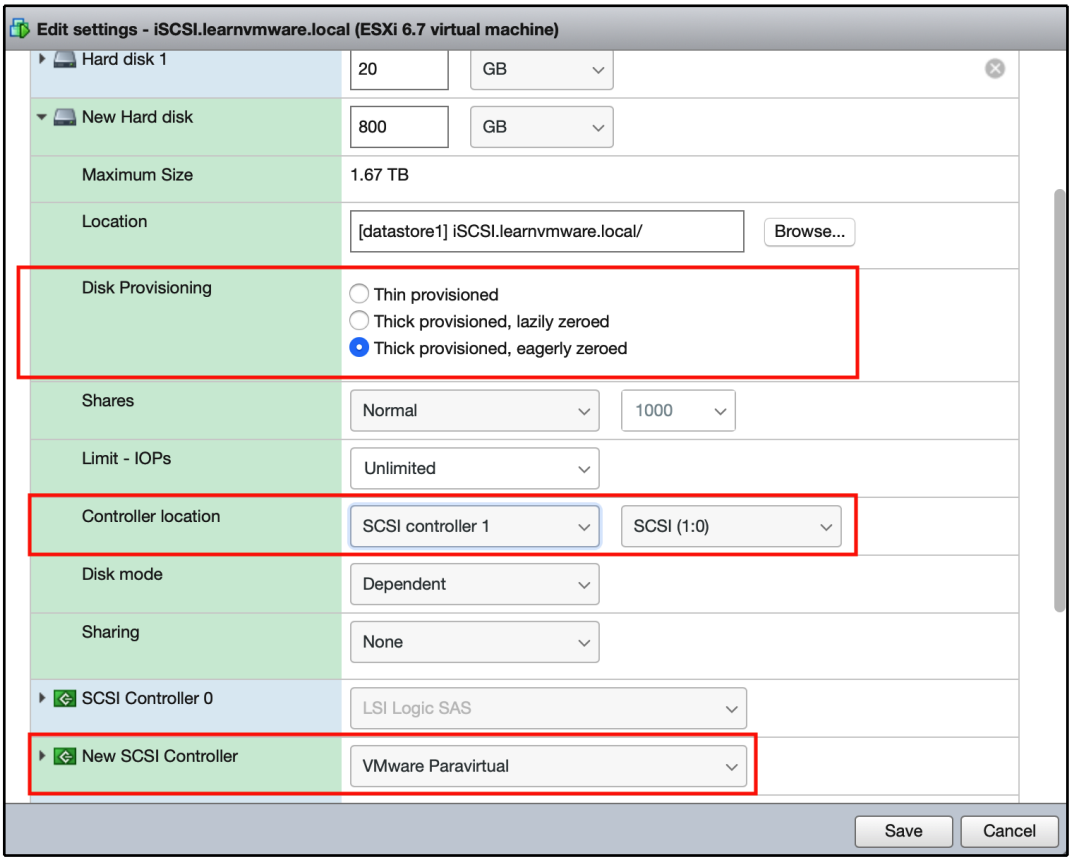
iscsi.learnvmware.local

Again, this server needs to be joined to the AD first. Once the server is part of the AD domain, we will install the iSCSI target service that will be used for our shared storage. Before we do that, we need to perform several reconfigurations on the Master ESXi level.

Storage design

Right now, our iSCSI virtual machine has only a single small disk, but we need to provide a much larger space for our vSphere infrastructure. We need to assign a new virtual disk to the iSCSI virtual machine.

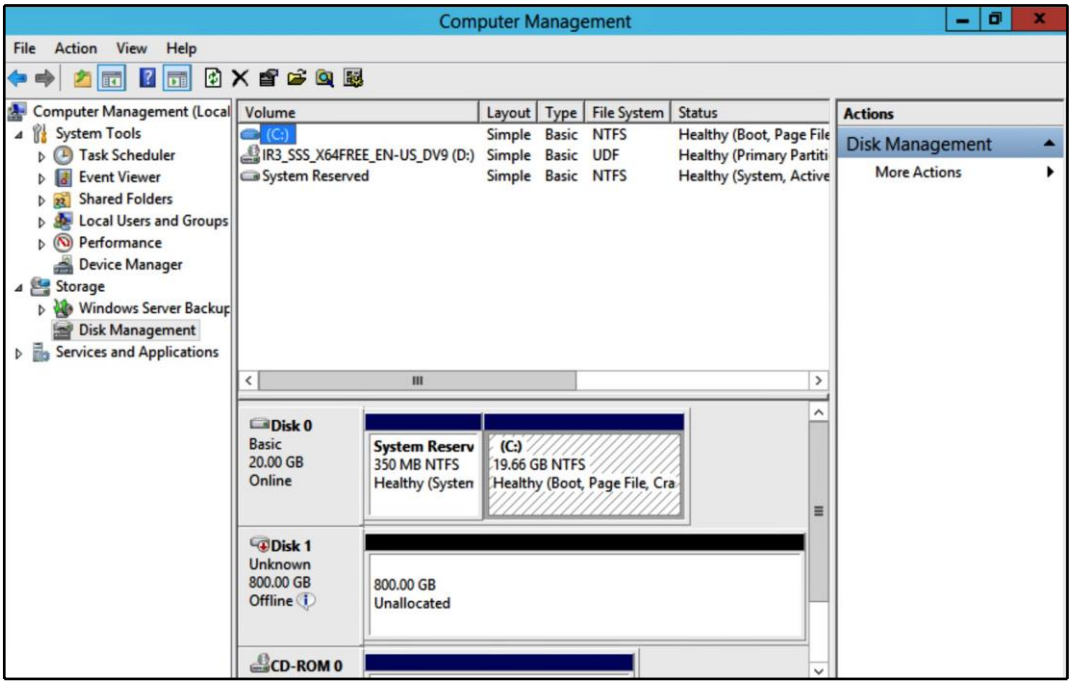
Bear in mind that for optimal performance, you should attach this new virtual disk to the PVSCSI controller and the disk type should be **Thick provisioned, eagerly zeroed**:



iSCSI target configuration

Now that we have attached our new virtual disk to the VM, we need to bring it online and format it. Follow these steps:

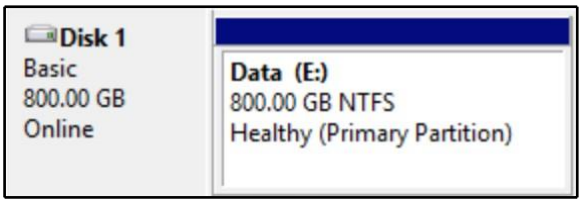
1. From **Tools**, select **Computer Management** and **Disk Management** under the **Storage** menu. As you can see, our guest OS can correctly see the new device:



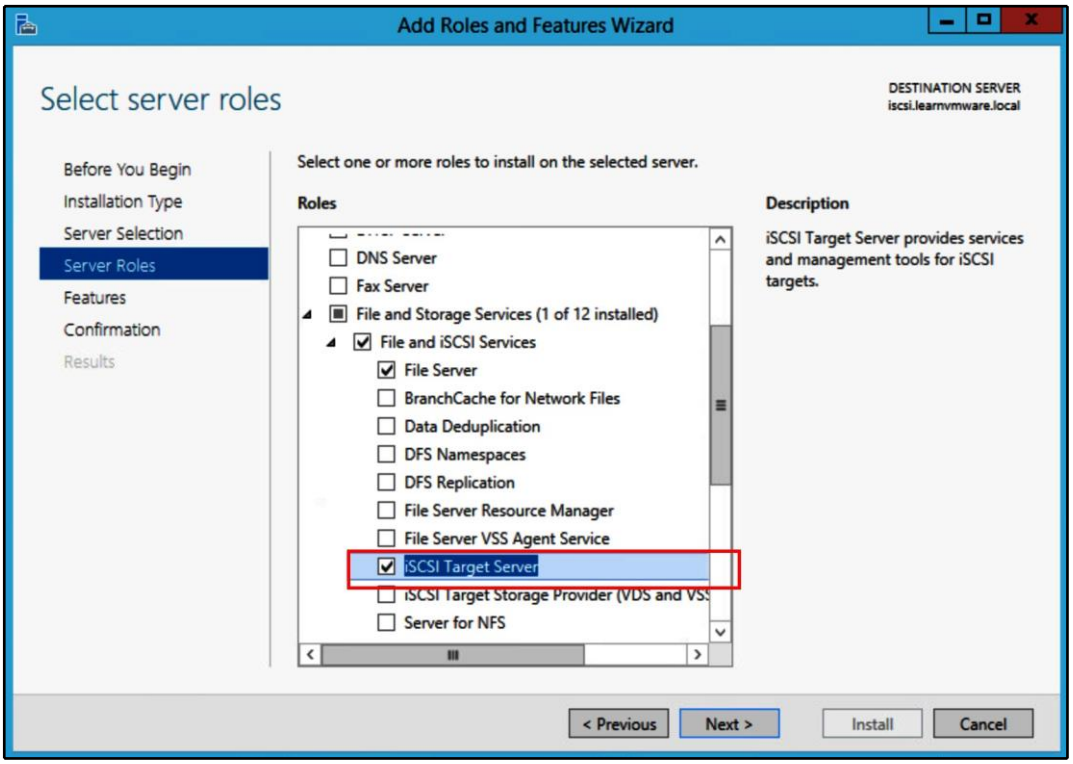
2. Right click on **Disk 1** and select **Online**.
3. Once the disk is online, right-click it again and select **Initialize**.

All that is left to do now is to create a new partition that will be used to store our iSCSI disks:

1. Right-click on the **Unallocated** space.
2. Select **New Simple Volume...**.
3. Assign a new drive letter, format the volume with NTFS, and provide a name for the volume if required. You should then be able to see that the disk is online and the new partition is created:



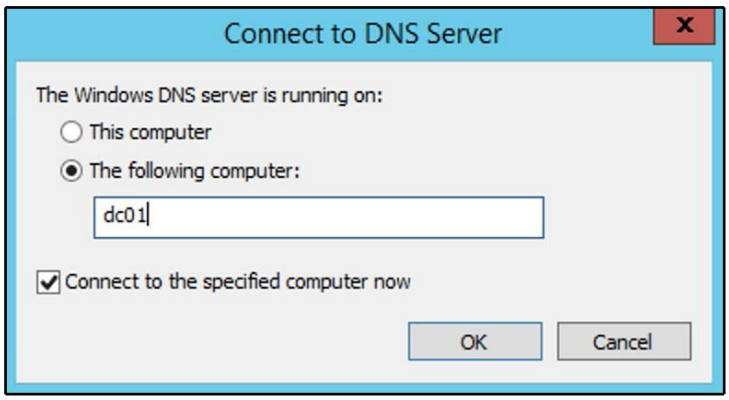
Once the disk is online, you can install the iSCSI target role. To do that, merely launch the add or remove roles and features wizard and select the **iSCSI Target Server** role:



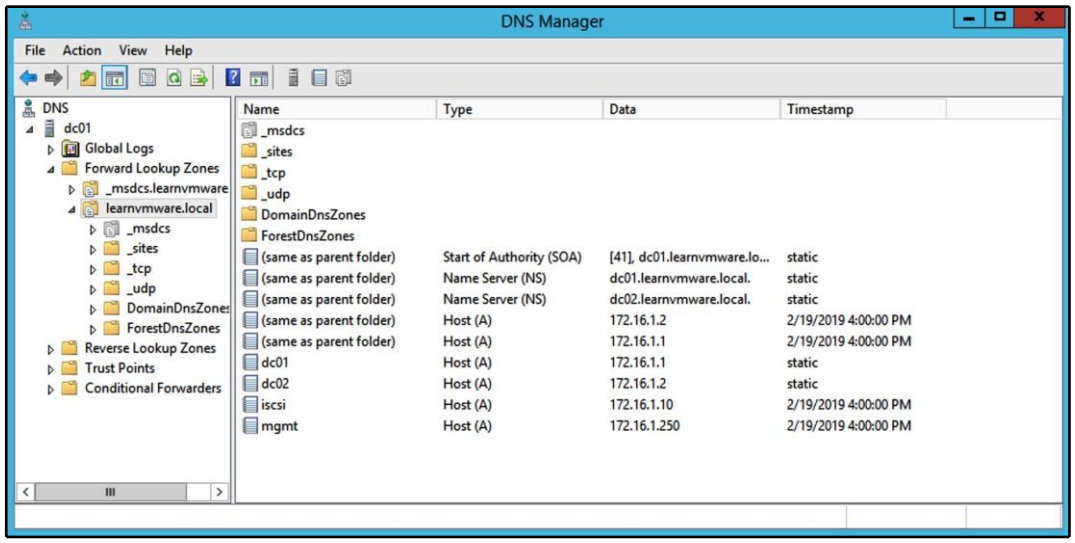
DNS configuration

For our vSphere infrastructure, we will need valid A DNS records. Let's add these to the DNS:

1. From the management server, click on **Tools** and select **DNS**.
2. Since the DNS service is not installed on our management station, we need to select one of our domain controllers. For example, let's choose `dc01`:



3. Once you are connected, navigate to your AD domain from the menu on the left:



4. Now, we need to add several **A** records. Right-click on the right-hand side and select **New Host (A or AAA)...** and fill in the IP address and name.

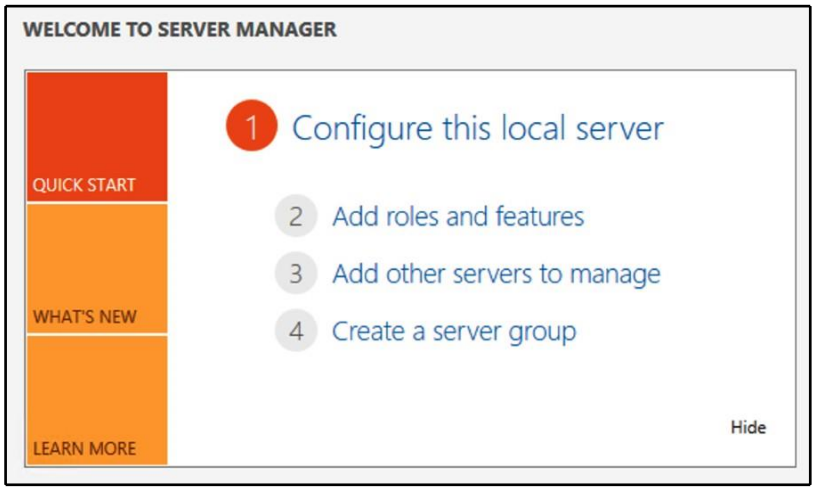
The following DNS records will be required:

IP address	DNS name
172.16.1.100	vcsa.learnvmware.local
172.16.1.11	esxi-prod-1.learnvmware.local
172.16.1.12	esxi-prod-2.learnvmware.local
172.16.1.13	esxi-prod-3.learnvmware.local
172.16.1.14	esxi-prod-4.learnvmware.local

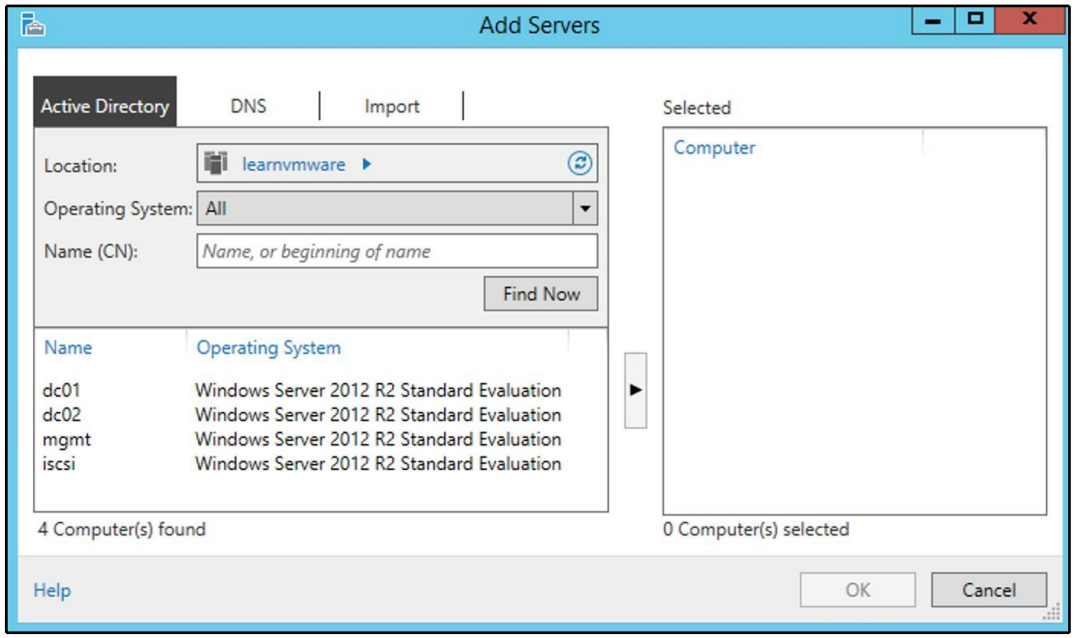
Centralized management

Once we have deployed our AD, we can benefit from centralized management. We will use our management server for all tasks within our AD Domain. Before we can do that, however, we need to add the other computers that will be managed from the management server:

1. From **Server Manager**, select **Add other servers to manage**, as shown in the following screenshot:



2. In the wizard, you can click the **Find Now** button to display all computer accounts from the AD. Select them and move them to the right:



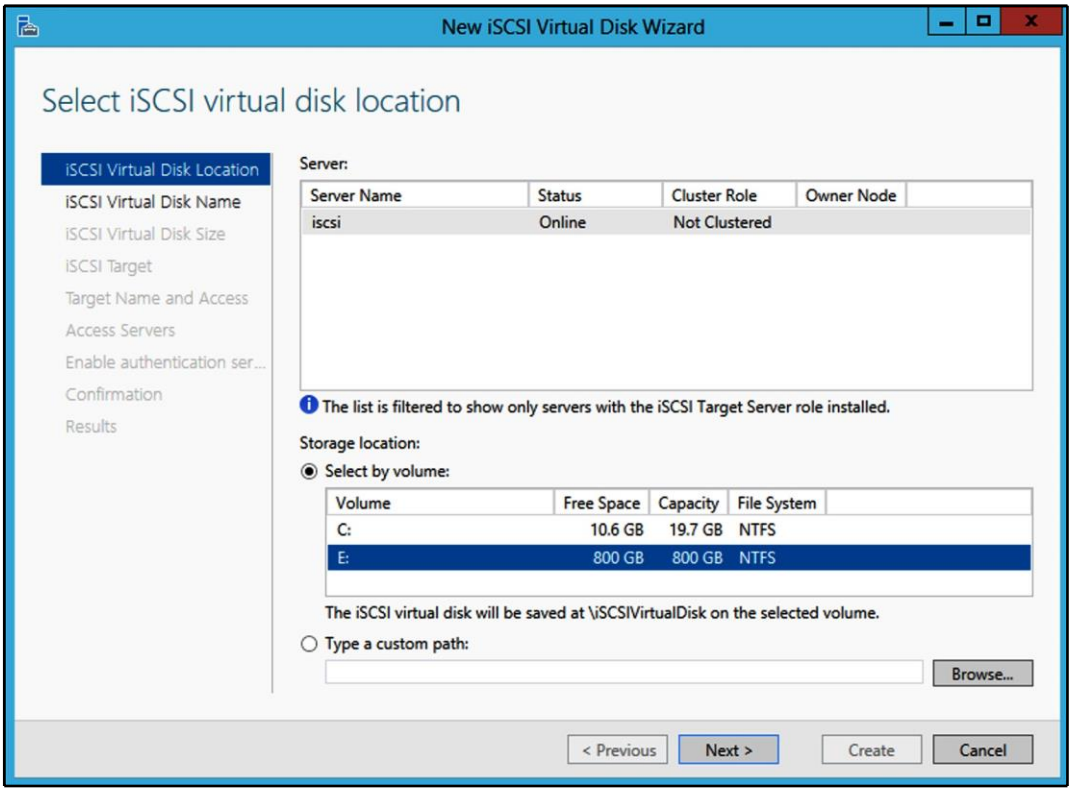
3. Once the servers are added, we can, for example, configure our iSCSI target service directly from our management station.

iSCSI target configuration

If you are logged in to either the management server or the iSCSI server, you can quickly provision a new disk that will be exported as an iSCSI volume:

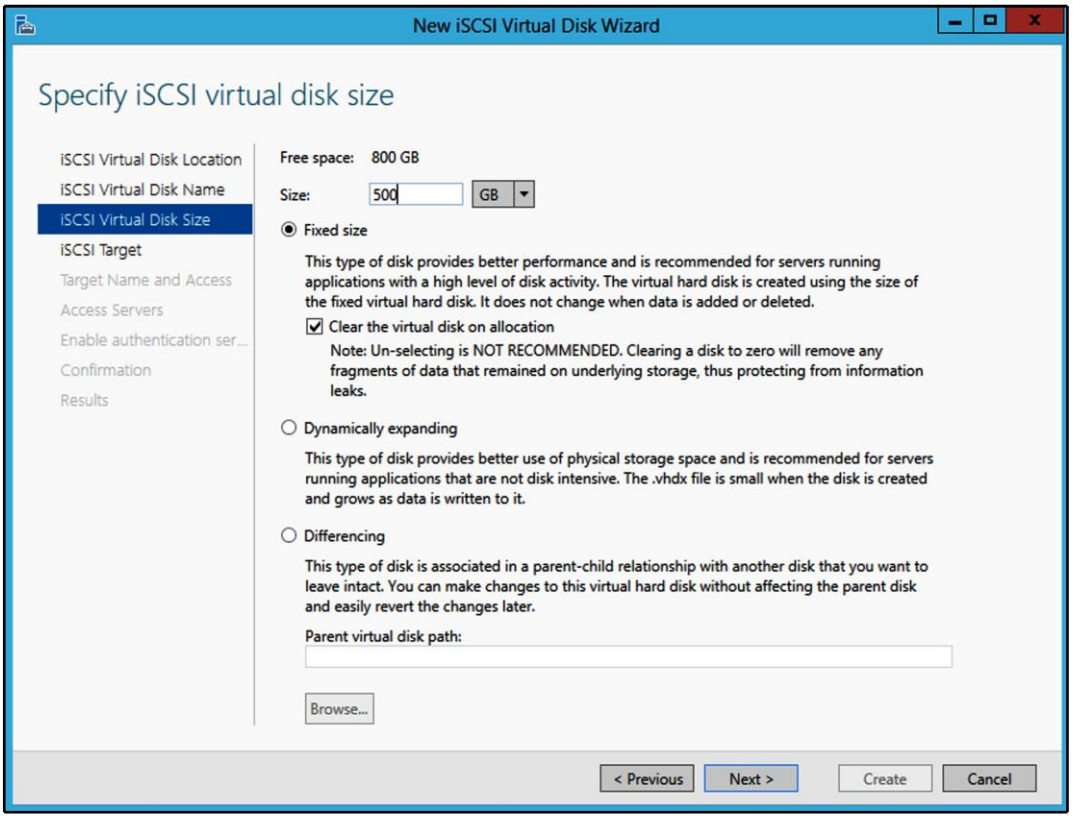
1. Open **Server Manager** and, under **File and Storage Services**, select **iSCSI**:

2. First, we need to configure our iSCSI device. Click on **Tools** and select a new iSCSI device.
3. Select the server on which the iSCSI device will be configured and the volume to store the content of the device:



4. In the iSCSI virtual disk, provide a name and description for the new iSCSI disk.

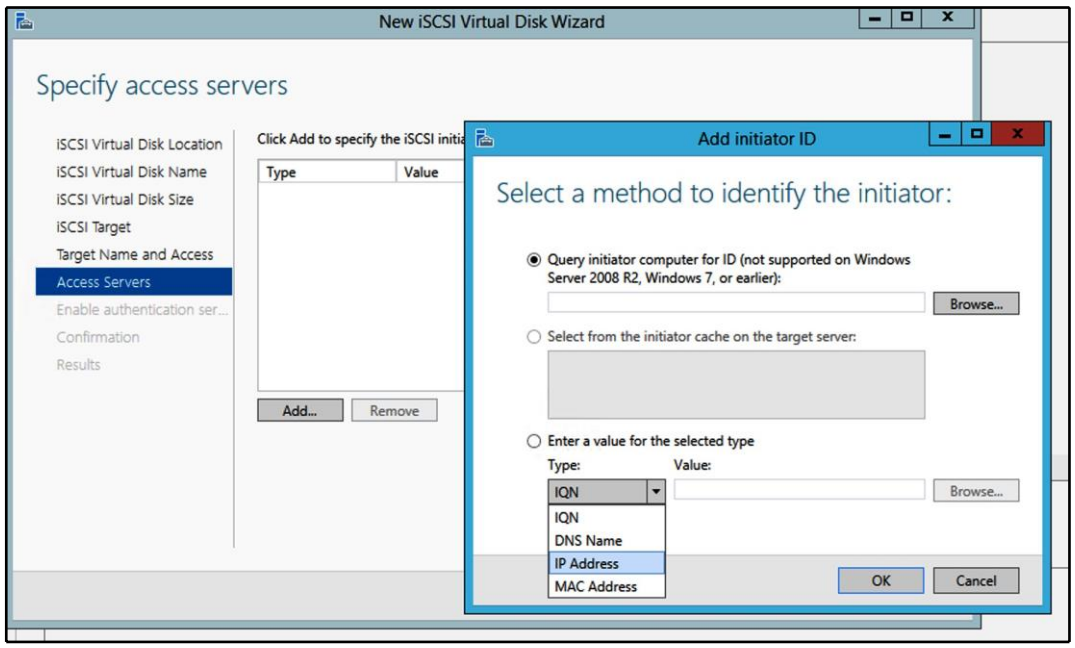
Now, you have the option to configure the size of the disk and its type. I have decided to create a 500 GB volume, as shown in the following screenshot:



Since we have not defined any iSCSI target, we need to define it now. The first step is to provide the name of the iSCSI target.

In **Access Servers**, you can define which iSCSI initiators will be able to connect to our storage. There are multiple options for how to identify the iSCSI initiator, but the most commonly used methods are using the IQN or the IP address.

Let's stick with a good old-fashioned IP addresses and provide the IP addresses of the VMkernel interfaces that will be used to connect to our iSCSI target (you can find the IP addresses in the IP address plan):



If you need to, you can also enable **CHAP authentication** for your iSCSI target. Then, confirm the creation of the new iSCSI device and the iSCSI target. You should be able to connect to the iSCSI server through iSCSI. You'll then be able to access the 500 GB volume.

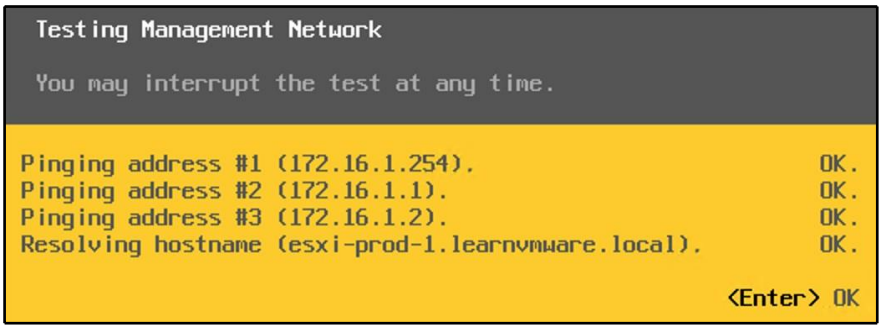
ESXi servers

Your virtual ESXi servers should be installed at this stage, so let's start with the configuration.

The first task will be to configure the management network of our virtual ESXi hypervisors. To do that, you need to access the **DCUI** console from the Master ESXi servers:

As you already know, to configure the IP address, you need to access the **Configure Management Network** option in the DCUI. In the interfaces, you have to select the physical NICs that are connected to the management network. In my case, these are `vmnic0` and `vmnic1`. No VLANs are used in the environment, so there is no need to configure them. All that remains is the IP configuration and the DNS settings.

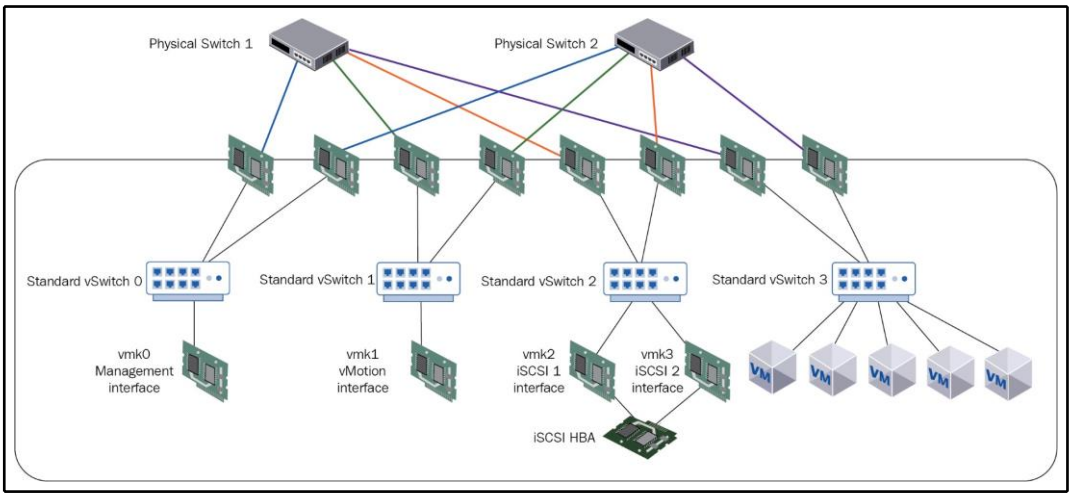
Once you have successfully configured the IP and DNS settings, do not forget to test the configuration using the **Testing Management Network** option:



Now, when the management connectivity is configured, we can switch to the web client of the ESXi servers and continue from there.

Network configuration

Let's take a look at the network configuration of our virtual ESXi servers:



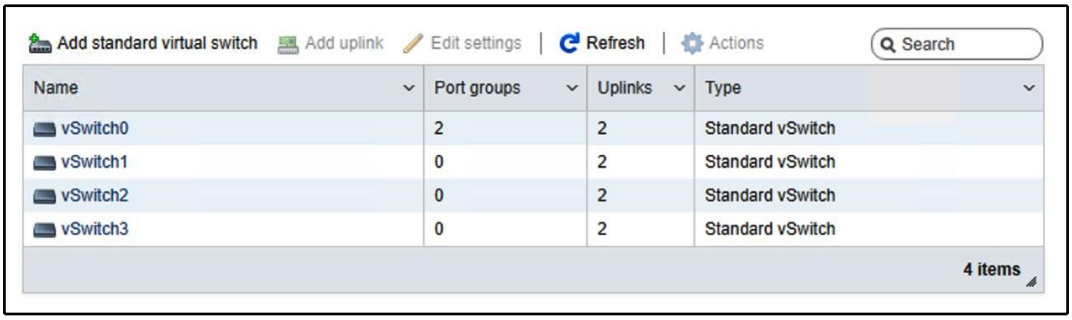
It looks a bit complicated, but you will soon get familiar with the design. I won't cover all the ESXi servers; I'll only cover the first one, since the configuration is the same for each apart from the IP addresses.

vSwitches

The following vSwitches need to be created:

vSwitch name	MTU	Physical adapters	Notes
`vSwitch1`	1500	vmnic2, vmnic3	vMotion
`vSwitch2`	9000	vmnic4, vmnic5	iSCSI
`vSwitch3`	1500	vmnic6, vmnic7	Production traffic

The configuration should end up looking as follows:

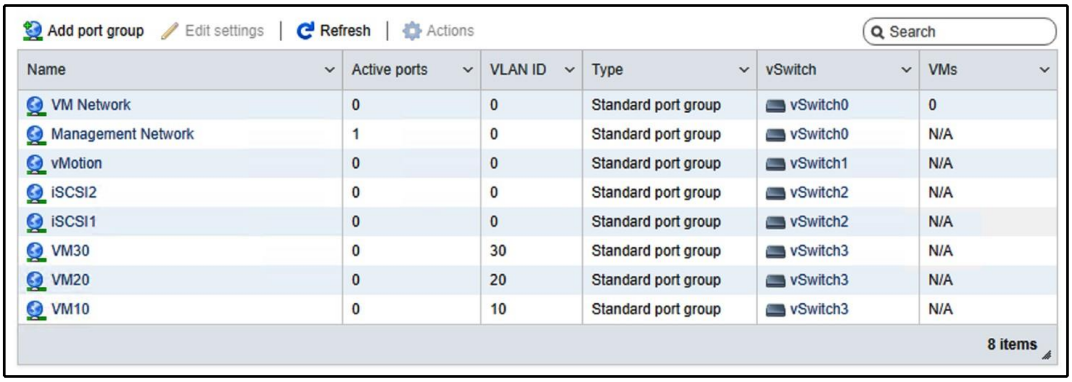


Port groups

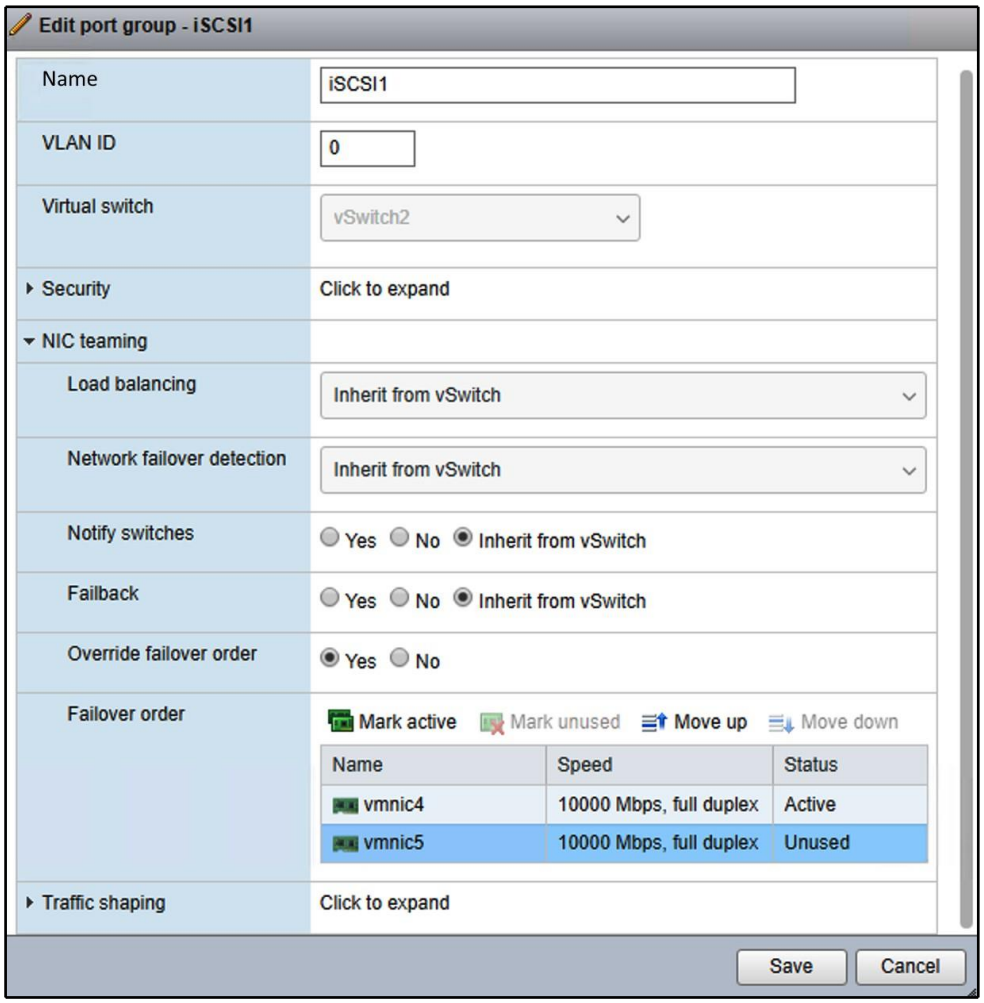
The following port groups need to be created:

Port group name	vSwitch	VLAN tag	Notes
`vMotion`	vSwitch1	0	vMotion
`iSCSI1`	vSwitch2	0	iSCSI PG 1
`iSCSI2`	vSwitch2	0	iSCSI PG2
`VM10`	vSwitch3	10	VM test PG 1
`VM20`	vSwitch3	20	VM test PG 2
`VM30`	vSwitch3	30	VM test PG 3

In the following screenshot, you can see the port groups that need to be configured:



There is one more task we need to do related to our iSCSI port groups. As you already know, if you need to work with storage multipathing, you must ensure that the VMkernel ports that are used to bind to the iSCSI initiator are not balanced over multiple **vmnics**. To do this, open the configuration of the first iSCSI port group and override the failover order. For the iSCSI1 port group, we will only use **vmnic4**. For the iSCSI2 port group, only **vmnic5** will be used. This is shown in the following screenshot:

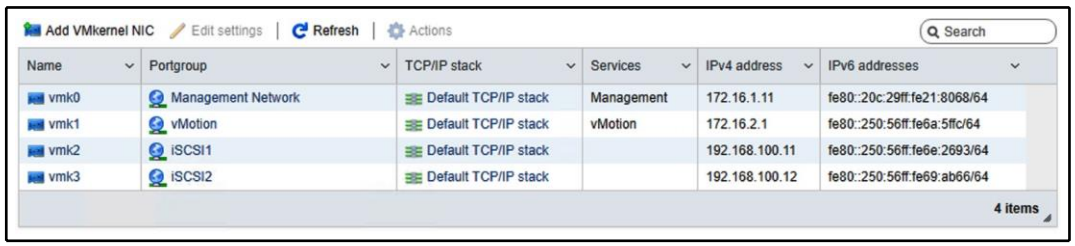


VMkernel ports

To enable our ESXi server to communicate over the network, we have to configure the following VMkernel ports:

VMkernel port name	IP address	Port group
Vmk1	`172.16.2.1`	vMotion
Vmk2	`192.168.100.11`	iSCSI 1
Vmk3	`192.168.100.12`	iSCSI 2

These can be seen in the following screenshot:



Network verification

Once you have configured your network, you can verify the connectivity from the CLI of the ESXi server. To do that, connect to your ESXi server using **SSH** (don't forget to start the SSH service if you have not done so already) and issue the `vmkping` command.

The following IP addresses should be accessible:

```
[root@esxi-prod-1:~] vmkping 192.168.10.1
 PING 192.168.10.1 (192.168.10.1): 56 data bytes
 64 bytes from 192.168.10.1: icmp_seq=0 ttl=128 time=0.285 ms
 64 bytes from 192.168.10.1: icmp_seq=1 ttl=128 time=0.281 ms

[root@esxi-prod-1:~] vmkping 192.168.10.2
 PING 192.168.10.2 (192.168.10.2): 56 data bytes
 64 bytes from 192.168.10.2: icmp_seq=0 ttl=128 time=0.302 ms
 64 bytes from 192.168.10.2: icmp_seq=1 ttl=128 time=0.200 ms
```

If any of the servers are not responding, make sure that the Windows Firewall is not blocking the communication.

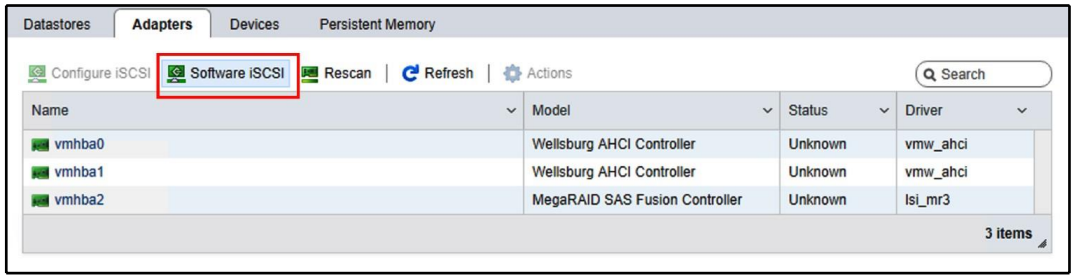
You can also check the ARP table of the ESXi server by using the following command:

```
[root@esxi-prod-1:~] esxcli network ip neighbor list
 Neighbor Mac Address Vmknic Expiry State Type
 ----------- ----------------- ------ ------- ----- ------
 172.16.1.250 00:0c:29:ec:1c:12 vmk0 858 sec Unknown
 172.16.1.1 00:0c:29:3f:6f:3d vmk0 1099 sec Unknown
 172.16.1.2 00:0c:29:ad:b5:d4 vmk0 1102 sec Unknown
 192.168.10.2 00:0c:29:95:16:36 vmk2 1096 sec Unknown
 192.168.10.1 00:0c:29:95:16:40 vmk2 1088 sec Unknown
```

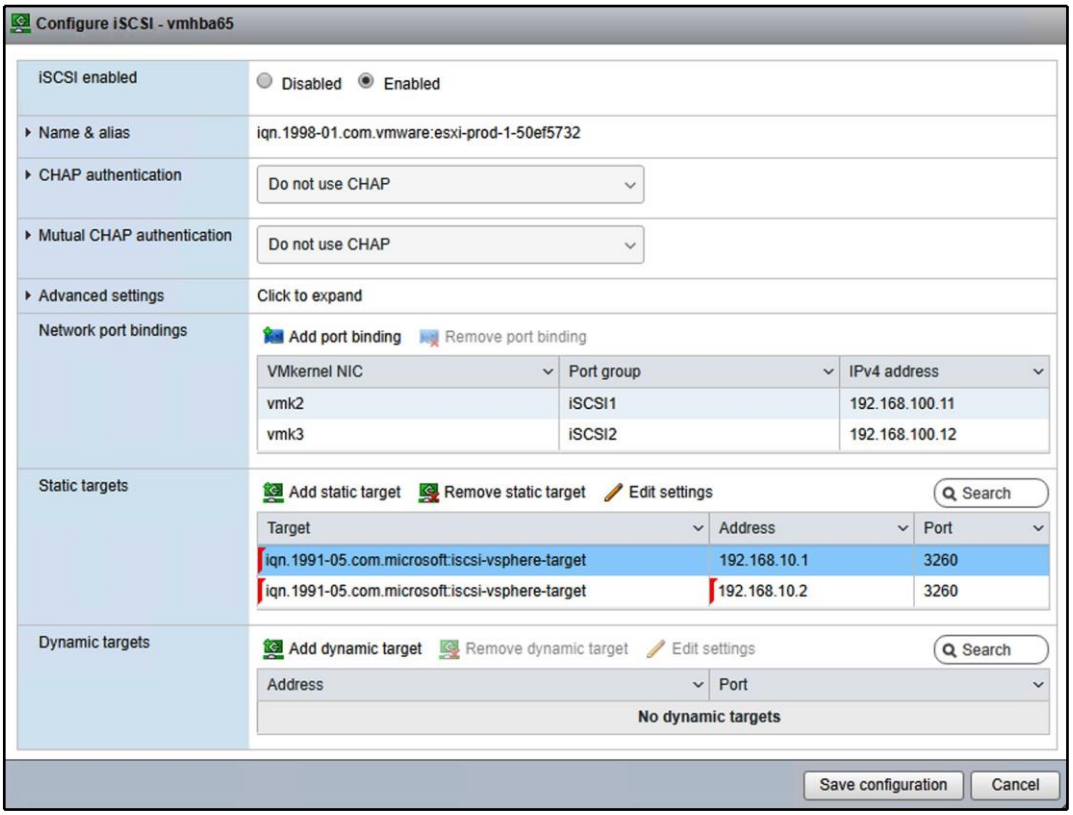
At this stage, you should be able to access all the network resources of our lab. It's time to take a look at the storage configuration.

Storage configuration

As you know, we are working with the iSCSI-based storage array. We will be using a software-based iSCSI initiator on the ESXi server. To configure the software-based iSCSI initiator, switch to the **Storage** view and the **Adapters** tab and select **Configure iSCSI**:

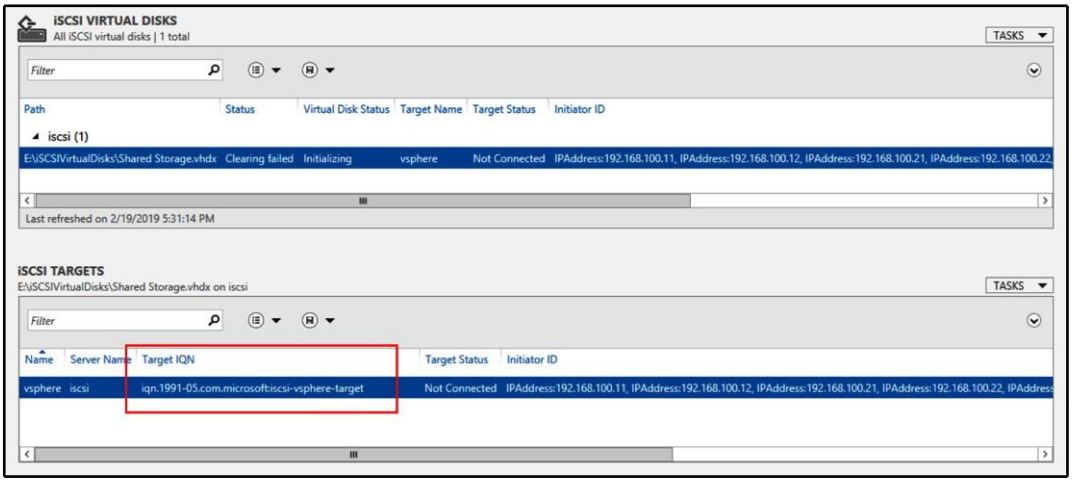


The new configuration wizard will be shown, as demonstrated in the following screenshot:

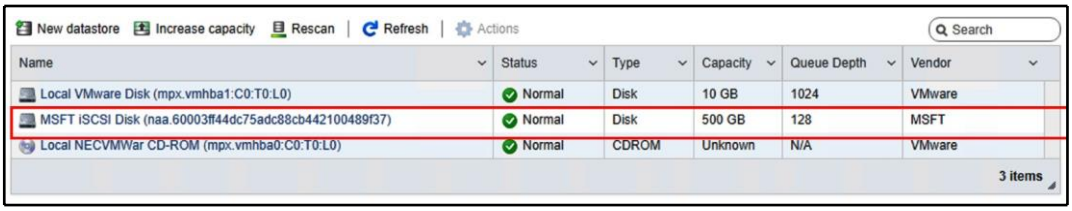


All you need to do is to enable the iSCSI initiator, bind your two new VMkernel adapters to the iSCSI initiator (**vmk2** and **vmk3**), and configure the static iSCSI target.

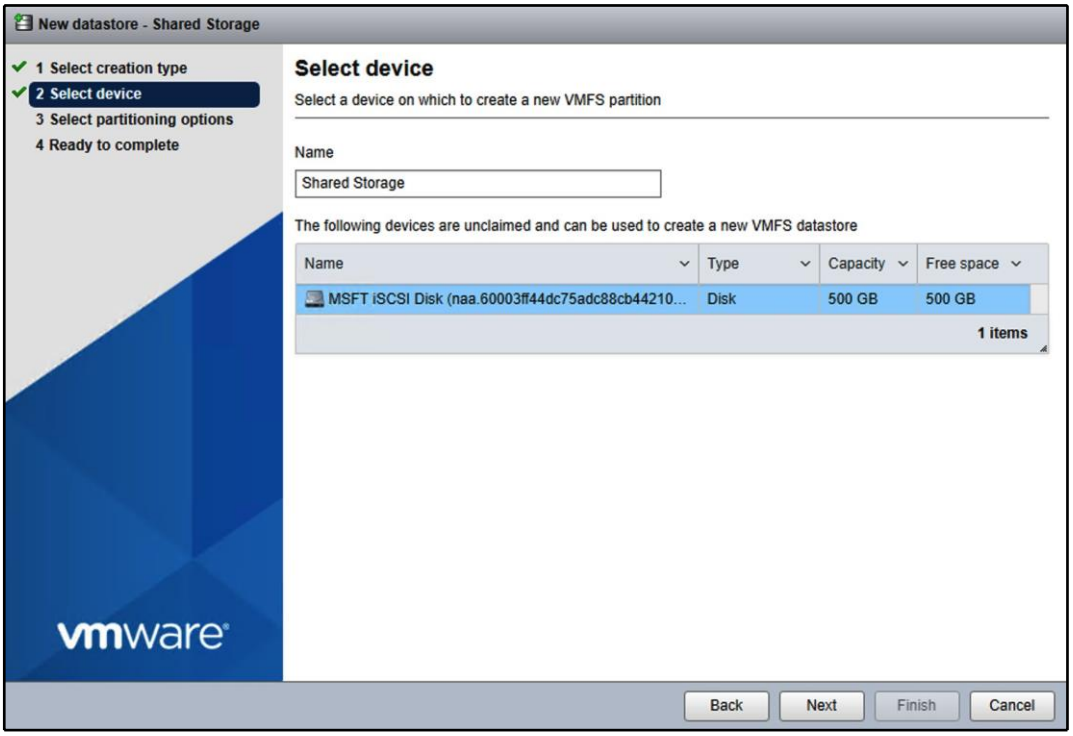
To obtain the name of your iSCSI target, open the **File and Storage Services** view from **Server Manager** and select the **iSCSI** option. In the following screenshot, you can see the iSCSI target name:



Once the iSCSI initiator is configured, the adapters are automatically rescanned by the ESXi server. If you have done the configuration correctly, you should be able to see your iSCSI device from the **Device** view, as shown in the following screenshot:



The final step that needs to be undertaken is to create a new datastore. To do this, switch to the **Datastore** view, click on **New datastore**, and select a new iSCSI disk, as shown in the following screenshot:

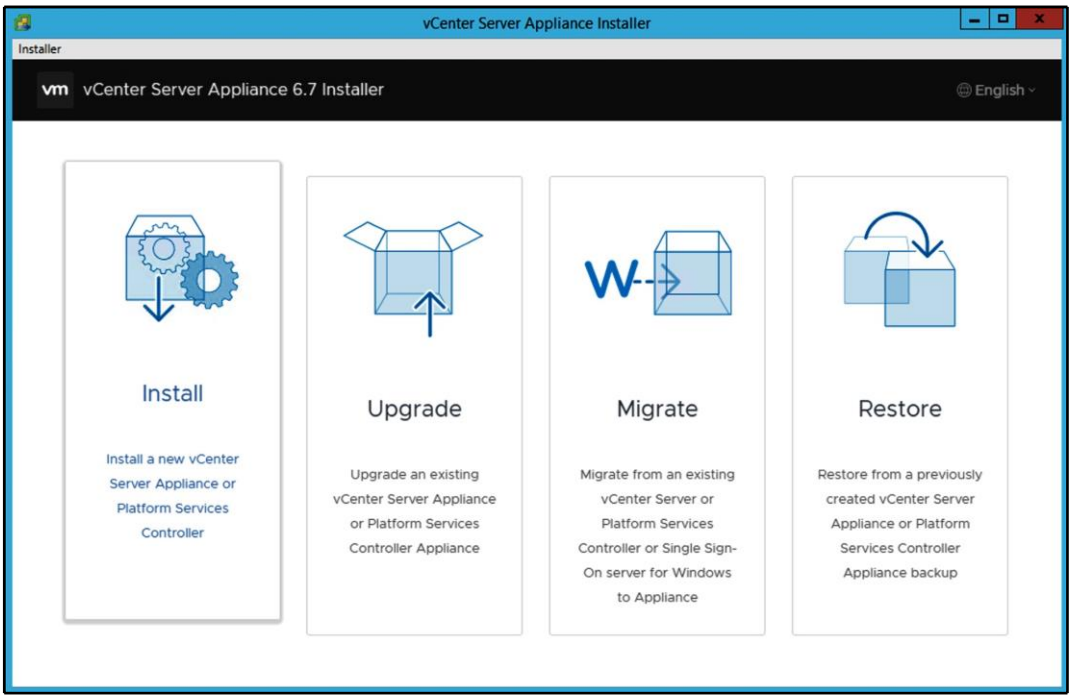


If you would like to test features such as Storage DRS, you will need multiple datastores and datastore clusters configured. You can provision more iSCSI disks on the iSCSI target server. After you rescan the iSCSI adapter within ESXi, you will see the additional devices.

You have successfully configured your first ESXi server. All you need to do now is to configure the remaining three ESXi servers so you end up with four fully configured ESXi hypervisors.

The vCenter Server

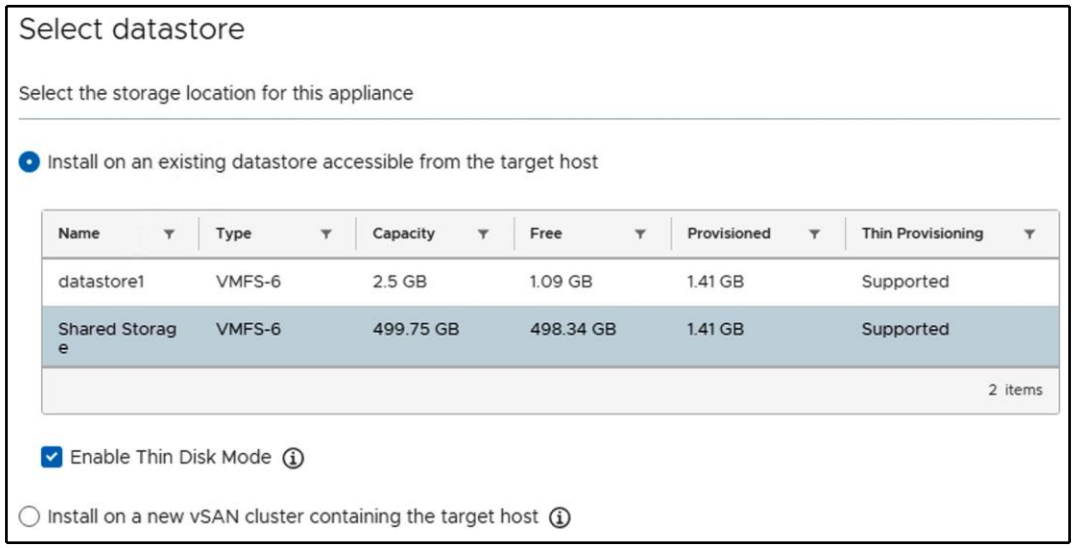
Now, when we have our ESXi servers installed, it is time to install the vCenter Server. To do that, just plug in the ISO image of the vCSA downloaded from the VMware website and follow the installation wizard:



In our case, we will be working with the embedded deployment mode, in which both the vCenter Server and PSC are deployed on the same appliance.

As a destination location for the installation, use our first ESXi server, `esxi-prod-1.learnvmware.local`. You need to provide the root password before you can proceed:

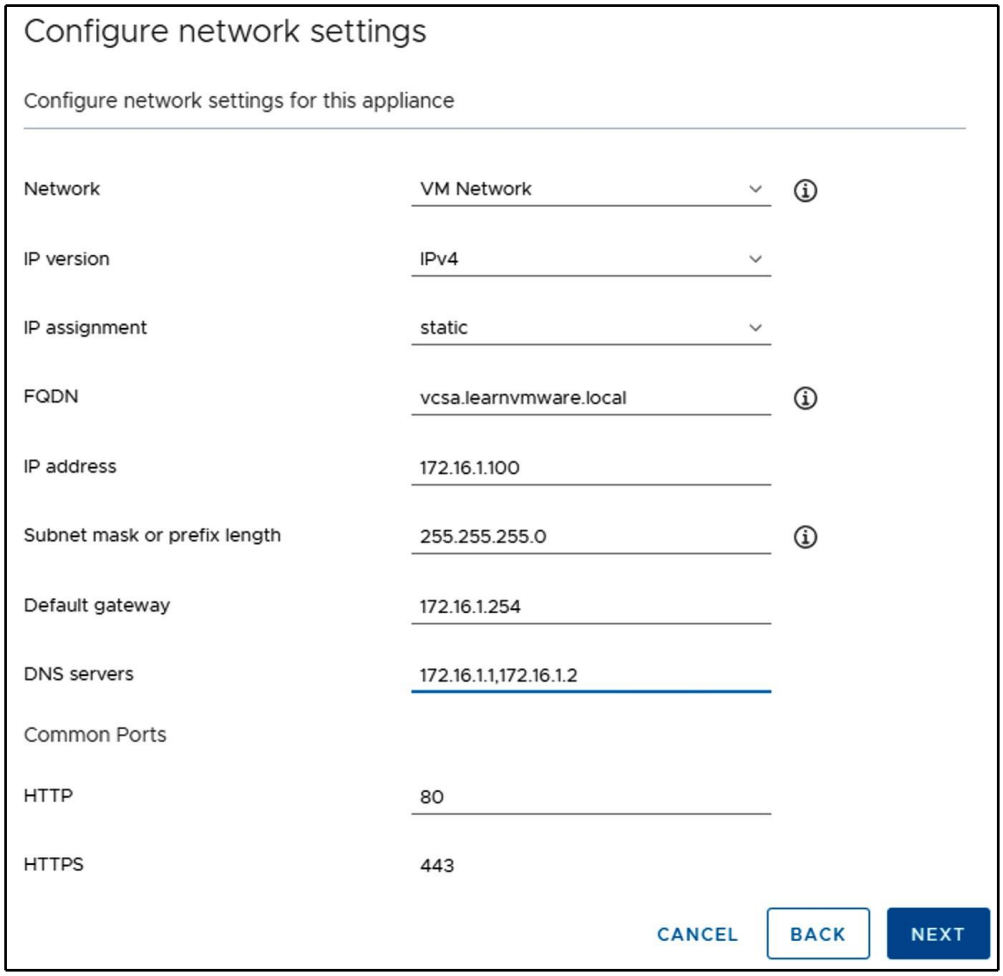
1. In the **Deployment size** option, select your desired infrastructure size. For most lab environments, the **Tiny** deployment size is sufficient.
2. Now, we need to select which datastore we will deploy the vCenter Server Appliance on. Make sure you select your iSCSI shared storage. If you want to save some storage space, select **Enable Thin Disk Mode** for the virtual disks:



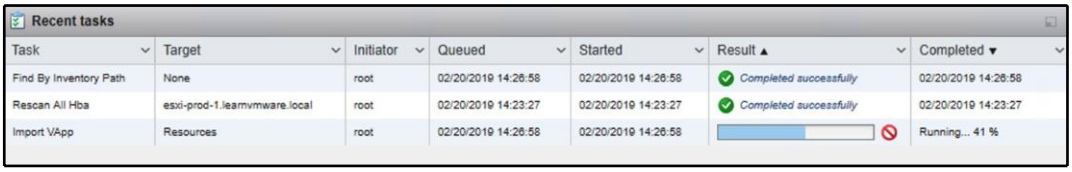
In the next stage, provide the network configuration for your vCSA. According to our IP address plan, you should fill in the following values:

- **Port group**: **VM Network**
- **FQDN**: `vcsa.learnvmware.local`
- **IP address**: `172.16.1.100`
- **Netmask**: `255.255.255.0`
- **Gateway**: `172.16.1.254`
- **DNS**: `172.16.1.1,172.16.1.2`

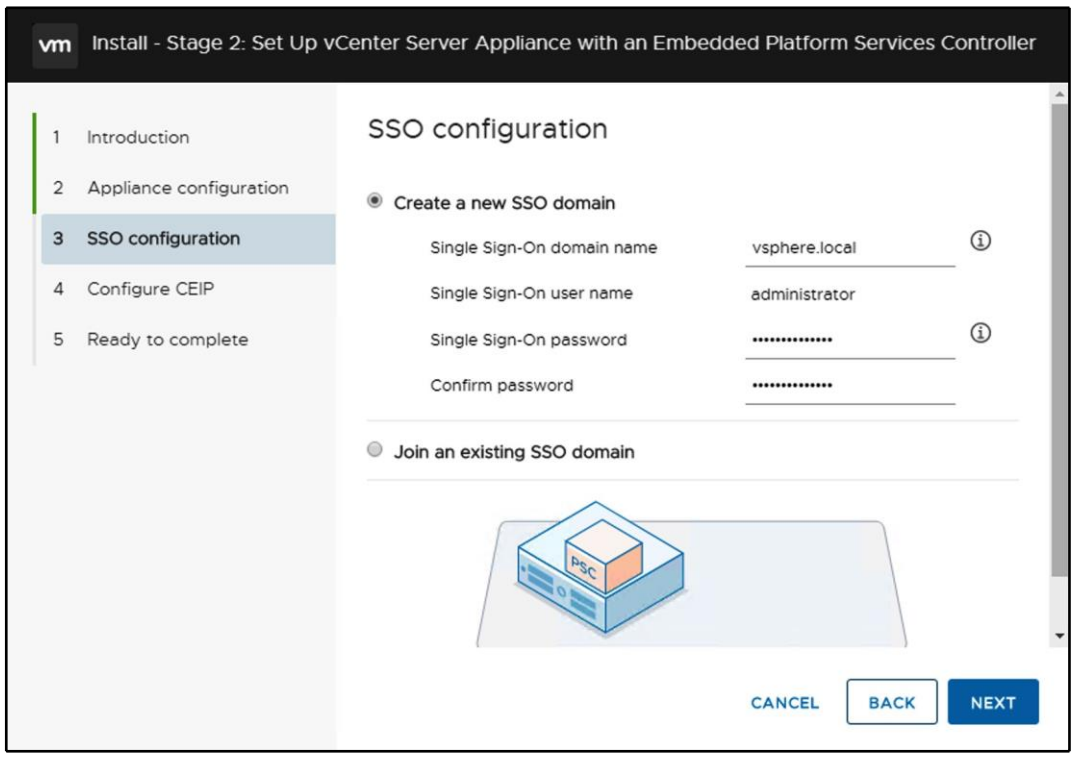
The following screenshot shows the network configuration:



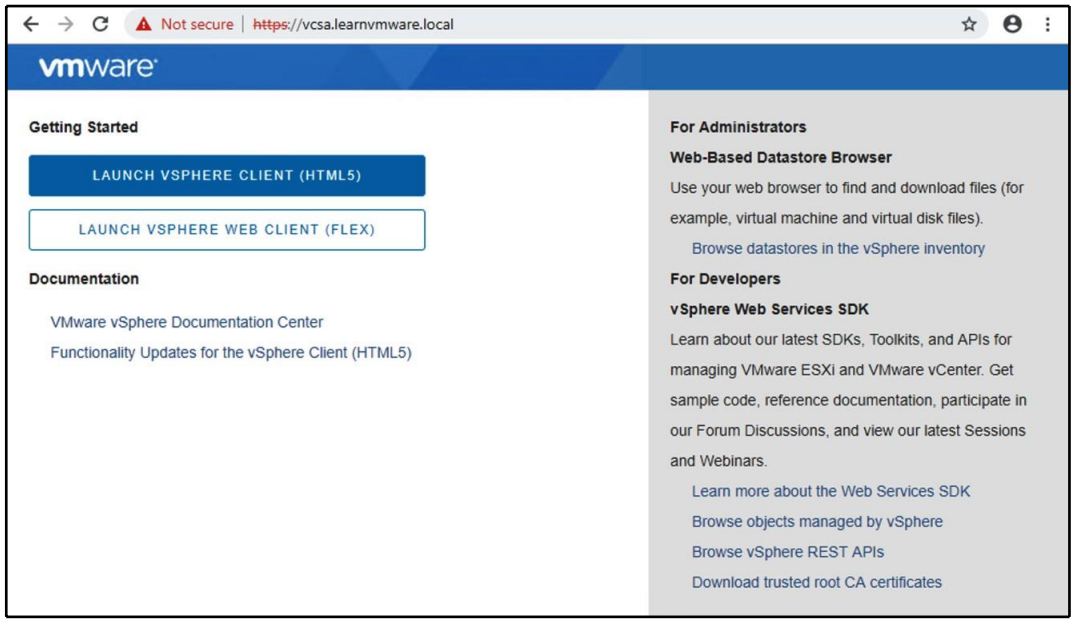
Review the network settings and proceed with the installation. If you check your first ESXi host, `esxi-prod-1.learnvmware local`, you should see that the installation wizard has initiated the deployment task:



Once this stage is completed, we can proceed with the configuration of the PSC. All we need to do is provide the default **Single-Sign-On domain name** and the **administrator** password, as shown in the following screenshot:



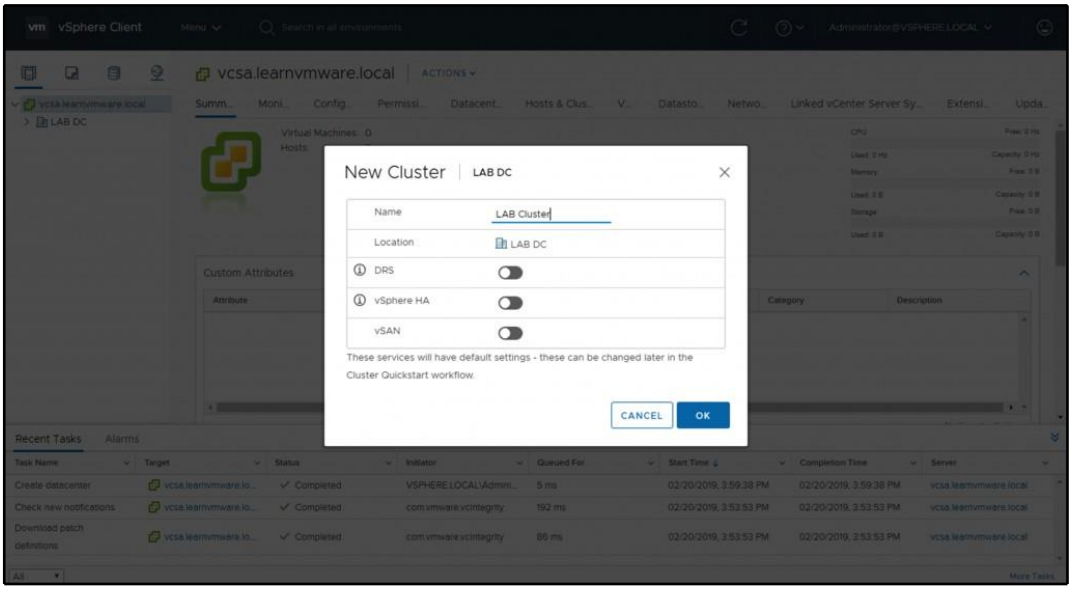
That's it. You have now successfully deployed your vCenter Server Appliance and you can try to log in to the system:



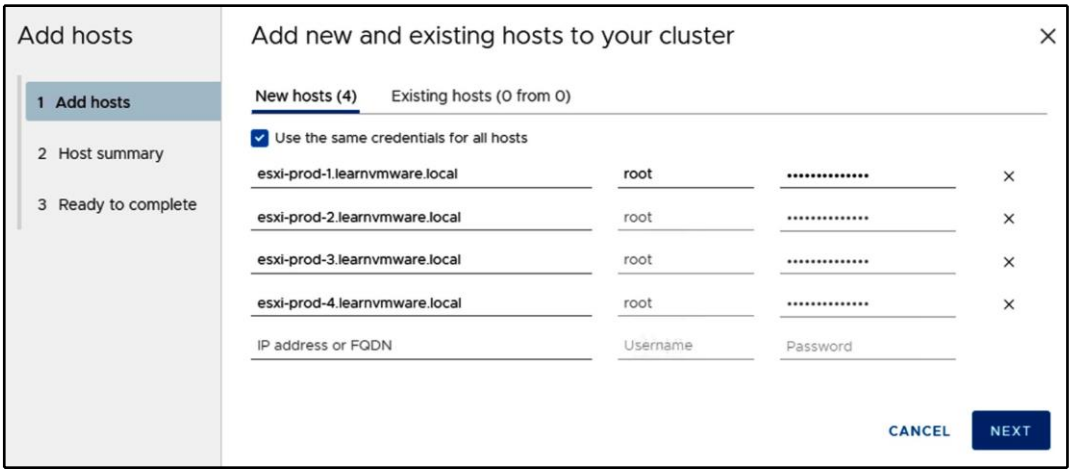
vSphere configuration

Now, when all your components are deployed and configured, you can start working on the vSphere configuration. I am not going to cover everything here; after all, it is your lab, so feel free to test anything we have covered or any other feature or technology you are interested in.

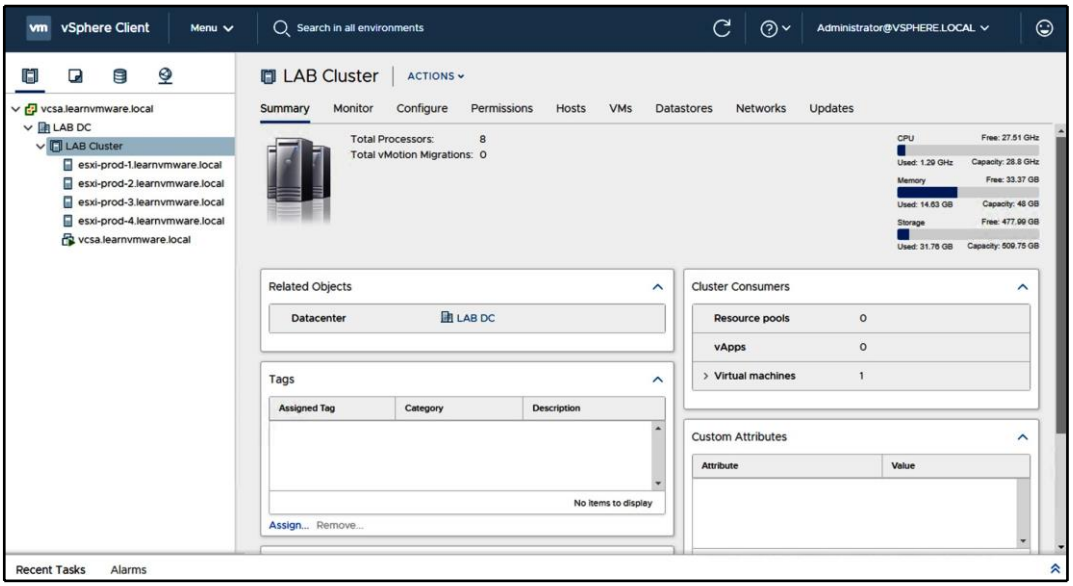
What we will do at this stage is configure our first data center object and vSphere cluster. As you already know, every vSphere object is related to the data center, so we need to create our first data center object. Now, when we have a data center, we can create our first vSphere cluster. If you would like to enable some cluster services at this stage, feel free to do so. In this example, however, I'm just going to create a default cluster without any additional services:



As a final step, let's add our four ESXi hypervisors to the cluster, as shown in the following screenshot:



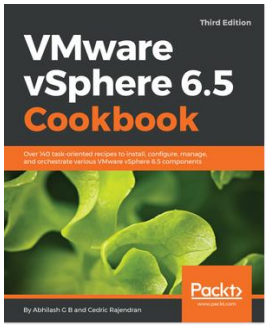
Finally, you should see your four ESXi hypervisors residing in your new vSphere cluster:



That's it! Congratulations ! You now have a lab environment in which you can test anything you need to.

Other Books You May Enjoy

If you enjoyed this book, you may be interested in these other books by Packt:



VMware vSphere 6.5 Cookbook - Third Edition
Abhilash G B, Cedric Rajendran

ISBN: 978-1-78712-741-8

- Upgrade your existing vSphere environment or perform a fresh deployment
- Automate the deployment and management of large sets of ESXi hosts in your vSphere Environment
- Configure and manage FC, iSCSI, and NAS storage, and get more control over how storage resources are allocated and managed
- Configure vSphere networking by deploying host-wide and data center-wide switches in your vSphere environment
- Configure high availability on a host cluster and learn how to enable the fair distribution and utilization of compute resources
- Patch and upgrade the vSphere environment
- Handle certificate request generation and renew component certificates
- Monitor performance of a vSphere environment

VMware NSX Cookbook
Bayu Wibowo, Tony Sangha

ISBN: 978-1-78217-425-7

- Understand, install, and configure VMware NSX for vSphere solutions
- Configure logical switching, routing, and Edge Services Gateway in VMware NSX for vSphere
- Learn how to plan and upgrade VMware NSX for vSphere
- Learn how to use built-in monitoring tools such as Flow Monitoring, Traceflow, Application Rule Manager, and Endpoint Monitoring
- Learn how to leverage the NSX REST API for management and automation using various tools from Python to VMware vRealize Orchestrator

Leave a review - let other readers know what you think

Please share your thoughts on this book with others by leaving a review on the site that you bought it from. If you purchased the book from Amazon, please leave us an honest review on this book's Amazon page. This is vital so that other potential readers can see and use your unbiased opinion to make purchasing decisions, we can understand what our customers think about our products, and our authors can see your feedback on the title that they have worked with Packt to create. It will only take a few minutes of your time, but is valuable to other potential customers, our authors, and Packt. Thank you!

Index

D

E

F

G

H

I

J

K

L

M

N

O

S

T

U

V

W