Appendix: VCP-DVC vSphere 6.7 Exam

Introduction

This document is a *very* rough draft of an appendix that we intend to be used with the *VCP6-DCV: Official Cert Guide* (VMware Press) when preparing for the Professional vSphere 6.7 Exam.

To prepare to take the vSphere 6.7 Exam, you can use this Appendix as your guide. It covers each 6.7 exam objective in order, providing various levels of detail for each objective. Some objectives are covered in detail here. Other objectives require you to follow the references to the *VCP6-DCV: Official Cert Guide* (VMware Press) and to official VMware documentation.

NOTE: Exam objectives 3.x and 6.x do not exit.

Objective 1.1

vSphere Overview and Components

VMware vSphere is a suite of products that you can use to virtualize enterprise datacenters and build private clouds.

vSphere Components

Table 1-1 describes the installable VMware products are the core components in a vSphere environment.

Table 1-1 Installable Core vSphere Components

Component	**Description**
vCenter Server	The major management component in the vSphere environment. Its services include vCenter Server, vSphere Web Client, vSphere Auto Deploy, and vSphere ESXi Dump Collector. vCenter Server for Windows also contains the VMware vSphere Syslog Collector.
Platform Services Controller (PSC)	The vSphere component that handles the infrastructure security functions such as vCenter Single Sign-On (SSO), licensing,

	certificate management and server reservation. Its services include vCenter Single Sign-On, License Service, Lookup Service, and VMware Certificate Authority.
ESXi Server	The host (hypervisor) on which virtual machines run.
VMware Update Manager (VUM)	A service that runs in the vCenter Server Appliance that facilitates upgrading and updating ESXi hosts and virtual machines.
VMware Tools	Software that is installed in the virtual machine guest OS that allows integration with the ESXi host.

Some optional, vSphere features require the deployment of additional components and specific vSphere licensing. Table 1-2 describes two of these optional components, which require deploying additional virtual appliances.

Table 1-2 Optional vSphere Components

Optional Component	**Description**
vSphere Replication	An extension to VMware vCenter Server that provides hypervisor-based virtual machine replication and recovery.
vCenter Server High Availability	Provides protection for the vCenter Server Appliance (VCSA) against host, hardware, and application failures. Provides automated active / passive failover with minimal downtime. It can also be used to significantly reduce downtime when you patch VCSA.

Many vSphere features, such as those described in Table 1-3, require specific vSphere licensing and configuration, but do not require the installation or deployment of additional software or virtual appliances.

Table 1-3 Available vSphere Features

Available vSphere Features	**Description**
vCenter Appliance File-Based Backup and Restore	A new feature in vSphere 6.7 that enables you to backup and restore

	the vCenter Server and PSC appliances
vMotion	Provides live virtual machine migrations with negligible disruption from a source ESXi host to a target ESXi host.
vSphere HA	Automated failover protection for VMs against host, hardware, network, and guest OS issues. In case of host system failure, cold migrates and restarts failed VMs on surviving hosts.
Distributed Resource Scheduler (DRS)	Balances VM workload in a cluster based on compute usage. Includes live (vMotion) migrations of VMs when necessary.
Storage vMotion	Live migrations with negligible disruption of VMs from a source datastore to a target datastore.
Fault Tolerance (FT)	Automated, live failover protection for VMs against host, hardware, network, and guest OS issues.
Distributed Power Management (DPM)	Optimizes power consumption in an ESXi cluster.
Proactive HA	Minimizes VM downtime by proactively detecting hardware failures and placing the host Quarantined Mode or Maintenance Mode.
Content Library	Centralized repository used manage and distribute templates, ISO files, scripts, vApps, and other files associated with VMs.
Host Profiles	Provides a means to apply a standard configuration to a set of ESXi hosts.

The add-on products in Table 1-4 are commonly used in a vSphere environment and are discussed in this guide. These products are typically sold separately from vSphere.

Table 1-4 Add-on Products

Product	Description
VSAN	A product that provides a SAN experience to your vSphere environment leveraging local storage in the ESXi hosts. It tightly integrates with vSphere and is the leading Hyper-Converged Infrastructure (HCI) solution to provide a flash-optimized, secure, and simple to use SAN.
NSX	A product that adds software based virtualized networking and security to a vSphere environment.
vRealize Suite	A suite of products that add operations (vRealize Operations Manager), automation (vRealize Automation), and orchestration (vRealize Orchestrator) to a vSphere environment.

NOTE: Although it is an add-on product, VSAN is covered in the VCP-DCV exam and in this guide.

The vSphere Host client is a web-based interface provided by each ESXi host. It is available immediately following the installation of a host. Its primary purpose is to provide a GUI for configuration, management, and troubleshooting purposes when vCenter Server is not available. For example, during the implementation of a new vSphere environment, you could use the vSphere Host Client to create virtual machines for running DNS, Active Directory, and vCenter Server database prior to deploying vCenter Server. For another example, you could use the vSphere Host Client to power-down, troubleshoot, reconfigure, and restart the vCenter Server virtual machine.

The vSphere Client is the preferred web-based GUI for managing vSphere. It is provided by services running in the vCenter Server. The vSphere Web Client is the legacy GUI, which is typically only used when necessary. It is also a service provided by the vCenter Server

Editions and Licenses

VMware vSphere comes in many editions, where each edition is intended to address specific use cases by providing specific features. When planning for a vSphere environment, you should prepare to procure at least three line items, a vCenter Server license, a vSphere license, and support for the environment. The vCenter license line item should identify the desired edition and quantity.

Table 1-5 provides a summary of the features that are provided with each edition of vCenter Server 6.7.

Table 1-5 vCenter Server Editions

Feature	Essentials	Essentials Plus	Foundation	Standard
Number of ESXi hosts	3 (2 CPU Max)	3 (2 CPU Max)	4	2000
vCenter License	Packaged with vSphere license in Essentials	Packaged with vSphere license in Essentials Plus	Sold separately from vSphere license	Sold separately from vSphere license
Basic level vCenter features, like single pane of glass management, Update Manager, and VMware Converter	Supported	Supported	Supported	Supported
Common vCenter features like vMotion and vSphere HA, vSphere Replication	Not supported	Supported	Supported	Supported
Advanced Features like vCenter Server High Availability (VCHA), vCenter Server Backup and Restore	Not supported	Not supported	Not supported	Supported

Table 1-6 provides *some* of the features that are provided with each edition of vSphere 6.7.

Table 1-6 vCenter Server Editions

Feature	Standard	Enterprise Plus	Platinum
vSphere HA, vSphere Replication, Storage vMotion, Quick Boot, vCenter Backup and Restore, VVOLs	Supported	Supported	Supported

Distributed Switch, Proactive HA, NIOC, SIOC, Storage DRS, DRS, DPM, VM Encryption, Cross-vCenter vMotion, Long Distance vMotion	Not supported	Supported	Supported
Fault Tolerance	Supported up to 2 vCPUs	Supported up to 8 vCPUs	Supported up to 8 vCPUs
Automated Discovery of Application Assets, contextual intelligence of application state, orchestrated response to security threats (AppDefense)	Not supported	Not supported	Supported

vCenter Server / Platform Services Controller Architecture

This section describes the architecture for the vCenter Server and Platform Services Controller (PSC).

vCenter Server Database

vCenter Server requires a database to store information about the objects in its inventory, such as virtual machines, clusters, hosts and data stores. The database will also store performance metrics that vCenter Server collects for all its objects. Each vCenter Server instance must have its own database. Each vCenter Server 6.7 Appliance uses an embedded PostgresSQL database. For environments with up to 20 hosts and 200 virtual machines, you can use the bundled PostgreSQL database for vCenter Server 6.7 for Windows. Larger environments require a supported external (Oracle or Microsoft SQL Server) when using vCenter Server 6.7 for Windows.

vCenter Server instances cannot share the same database schema. Multiple vCenter Server databases can reside on the same database server, or they can be separated across multiple database servers. You cannot install vCenter Server and point to an older external vCenter Server database.

If you choose to use an external database, you should deploy the database and the associated ODBC DSN entry prior to deploying vCenter Server.

SSO

vCenter Single Sign-On (SSO) is an authentication broker and security token exchange infrastructure. When users authenticate to vCenter Single Sign-On, they receive a SAML token that they use going forward to authenticate to vCenter services. In vSphere 6.x, vCenter Single Sign-On is provided by the PSC. The PSC provides other services that support vCenter Server and vCenter Server components, such as VMware Certificate Authority and License Service.

Each PSC is associated with a SSO domain. The domain name defaults to `vsphere.local`, but in vSphere 6.x you can change it during installation of the first PSC. To prevent authentication conflicts, use a name that is not used by OpenLDAP, Microsoft Active Directory, or other directory services. The domain determines the local authentication space. You can split a domain into multiple sites and assign each PSC and vCenter Server instance to a site. Sites are logical constructs, but usually correspond to geographic location.

> **NOTE**: You cannot change the domain to which a Platform Services Controller or vCenter Server instance belongs.

SSO is composed of many components, including the Security Token Service (STS), the Administration Server, the VMware Directory Service (vmdir) and Identity Management Service.

STS issues Security Assertion Markup Language (SAML) tokens, which represent the identity of users from its identity sources.   The tokens allow users, who successfully authenticate to SSO to use any vCenter service that SSO supports without needing to authenticate again to each service. SSO signs each token with a signing certificate, which it stores on disk.

The Administration server allows users with administrator privileges in SSO to use the vSphere Web Client to configure SSO and to configure SSO users and groups. Initially, only the administrator account in the SSO domain can configure SSO. In vSphere 5.5, this user was `administrator@vsphere.local`, but in vSphere 6.0, you can change the SSO domain during the vCenter Server installation.

The vmdir service is associated with the domain that was created during the vCenter Server installation. It is included in each embedded deployment and on each Platform Services Controller (PSC). It is a multi-tenanted, multi-mastered directory service that makes an LDAP directory available on port 389. It also uses port 11711 for backward compatibility with vSphere 5.5 and earlier systems. For environments with multiple PSC instances, the vmdir content of one instance is replicated to the other. Starting in vSphere 6.0, the vmdir service stores not only SSO information, but also certificate information.

The Identity Management Service handles the identity sources and STS authentication requests.

vCenter Server and PSC Deployment Types

Concerning the vCenter Server and PSC architecture, you have many options, such as vCenter Server for Windows vs vCenter Server Appliance (VCSA), embedded PSC vs external PSC, and Enhanced Link Mode vs Embedded Linked Mode.

During the vCenter Server and PSC installation, you can choose from three deployment types as described in Table 1-7.

Table 1-7 vCenter Server and PSC Deployment Types

Deployment Type	Description
vCenter Server with an embedded Platform Services Controller	Deploy all vCenter Server and PSC services on the same virtual machine or physical server. See Figure 1-1.
Platform Services Controller	Deploy just the PSC services on the virtual machine or physical server. See the top node in Figure 1-2
vCenter Server with an external Platform Services Controller	Deploy just the vCenter Server services on the virtual machine or physical server. (Requires a

	previously deployed PSC). See the bottom nodes in Figure 1-2

Figure 1-1 vCenter Server with Embedded PSC

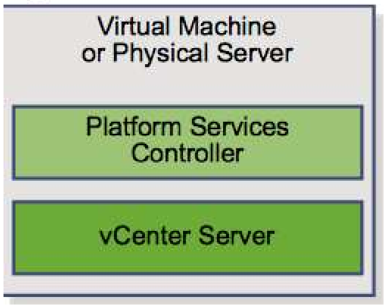
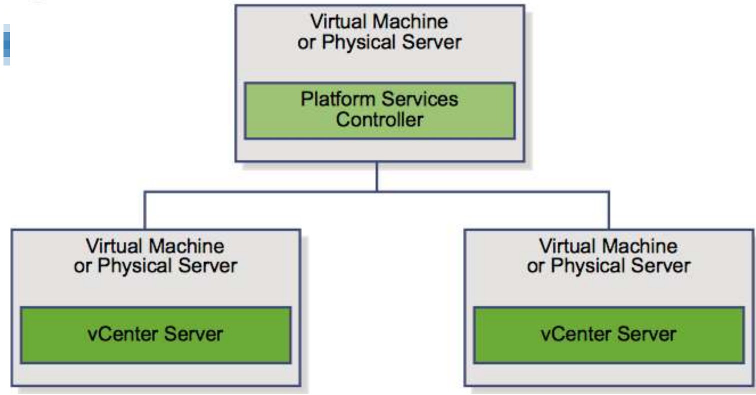

Virtual Machine or Physical Server
Platform Services Controller
vCenter Server

Figure 1-2 Two vCenter Servers with External PSC


Virtual Machine or Physical Server
Platform Services Controller
Virtual Machine or Physical Server
vCenter Server
Virtual Machine or Physical Server
vCenter Server

The advantages of deploying vCenter Server with an embedded PSC are:

- Communication between the vCenter Server and PSC is directly within the server and is not over the network. It is not prone to outages due to network connectivity or name resolution issues.
- When deployed on Windows Severs, fewer Windows licenses are required.
- Fewer servers will need to be managed.

Note: After you deploy or install vCenter Server with an embedded Platform Services Controller, you can reconfigure the deployment type and switch to vCenter Server with an external Platform Services Controller

The disadvantages of deploying vCenter Server with an embedded PSC are:

- An instance of PSC is required for each vCenter Server, which may consume more resources.
- The model may not be suitable for large environments.

The advantages of deploying vCenter Server with an external PSC are:

- Fewer resources are consumed by the combined services in PSC.
- Supports multiple vCenter Server instances

The disadvantages of deploying vCenter Server with an external PSC are:

- The connection between the vCenter Server and PSC are susceptible to network issues.
- When deployed on Windows Servers, additional Windows licenses may be required
- Additional servers must be managed

When using vCenter with an external PSC, the PSC can serve multiple vCenter Server instances, which may be a mixture of Windows based vCenter Servers and vCenter Server appliances. In this case, the PSC itself could be deployed in a Windows Sever or in a PSC appliance.

You can deploy multiple instances of PSC that replicate data to provide high availability. When installing vCenter Server with an embedded PSC or when installing an external PSC, you can choose to create a new domain or choose to join an existing domain. When creating a new domain, you must specify a new SSO site name. When joining an existing domain, you may choose to create a new SSO site or join an existing SSO site. Specifying additional sites is key to installing SSO in a multisite deployment. The vCenter Single Sign-On multisite configuration is designed for deployments with multiple physical locations

A typical multisite deployment involves a vCenter Server with an external PSC implemented at each site, where the PSCs share a common SSO domain. The vCenter Server at each site is aware of the site topology and uses the local PSC in normal circumstances. This topology allows for the use of Enhanced Linked Mode. Optionally, a second instance of PSC could be deployed to each site to provide high availability. At each site, a load balancer is used for connecting to the PSC nodes. Optionally, Windows based vCenter Server nodes could be clustered with Windows Server Failover Cluster (WSFC) at each site.

Enhanced Linked Mode uses one or more PSC instances to link multiple vCenter Servers, allowing users to view and search across the linked vCenter Servers. Enhanced Linked Mode provides the following features:

- Works with Windows based vCenter Servers as well as vCenter Server appliances.
- Users can logon into all linked vCenter Instances simultaneously with a single user name and password.
- Users can view and search the inventories of multiple linked vCenter Server instances
- Replicates roles, permissions, licenses, tags and policies.

To join vCenter Server instances in Enhanced Linked Mode, connect them to the same PSC or to PSC instances that share the same SSO domain. Enhanced Linked Mode requires vCenter Server Standard licensing. It is not supported with vCenter Server Foundation or vCenter Server Essentials.

NOTE: You can create a vCenter Enhanced Linked Mode group only during the deployment of vCenter Server Appliance or installation of vCenter Server. You cannot create a vCenter Enhanced Linked Mode group after you install vCenter Server or after you deploy the vCenter Server Appliance.

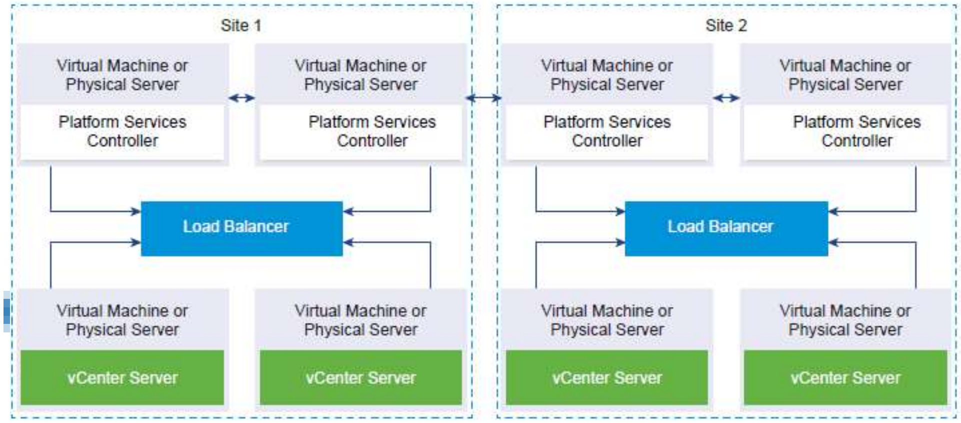
In vSphere 5.5 and earlier, Linked Mode relied on Microsoft ADAM to provide replication. Starting in vSphere 6.0, the PSC provides the replication for Enhanced Linked Mode and ADAM is no longer required. Because of this change, you must isolate vCenter Server 5.5 instances from any Linked Mode groups prior to upgrade.

In vSphere 6.5 and later, sites are important. In the event of a PCS failover, the vCenter Server instances are repointed to a different PSC in the same site.

To ensure PSC high availability in external deployments, you must install or deploy at least two joined PSC instances in your vCenter Single Sign-On domain. You can include a load balancer to provide automatic failover, but this requires the PSC instances be of the same operating system type. Using PSC instances with mixed operating systems behind a load balancer is unsupported.

Figure 1-3 shows an example of two sites with load balanced PSC and vCenter Server pairs.

Figure 1-3 Two Sites with Load Balanced Pairs


Site 1
Site 2
Virtual Machine or Physical Server
Platform Services Controller
Load Balancer
vCenter Server

Alternatively, you can join two or more PSC instances in the same site with no load balancer to provide high availability with manual failover.

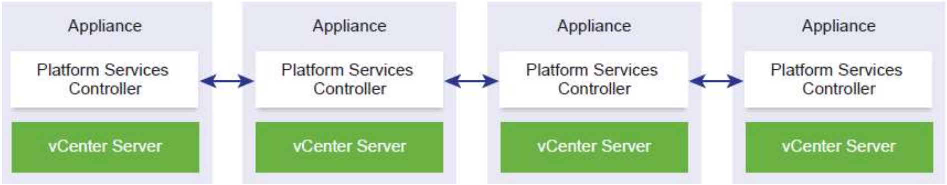
If you deploy a SSO domain that includes three or more PSC instances, you can manually create a ring topology, which ensures reliability when one of the instances fails. To create a ring topology, run the following command against the first and last PSC instance that you deployed.

```
/usr/lib/vmware- vmdir/bin/vdcrepadmin -f createagreement
```

vCenter Enhanced Linked Mode allows you to manage the inventories of a group of vCenter Servers and VCSAs by logging into just any member of the group. You can join up to ten VCSAs and eight vCenter Server systems with vCenter Enhanced Linked Mode. You can create a vCenter Enhanced Linked Mode group only during the deployment of VCSA or installation of vCenter Server. You cannot create a vCenter Enhanced Linked Mode group after you install vCenter Server or after you deploy VCSA.

vCenter Embedded Linked Mode is effectively Enhanced Linked Mode using VCSA with embedded PSC as shown in Figure 1-4.

Figure 1-4 Embedded Linked Mode


Appliance
Platform Services Controller
vCenter Server
Appliance
Platform Services Controller
vCenter Server
Appliance
Platform Services Controller
vCenter Server
Appliance
Platform Services Controller
vCenter Server

NOTE: Embedded linked mode is not supported for Windows vCenter Server installations. vCenter Embedded Linked Mode is supported starting with vSphere 6.5 Update 2 and suitable for most deployments

NOTE: During Stage 2 of the deploying a new VCSA with an embedded PSC, if you choose to create a new SSO domain and you plan to use embedded link mode, then use `Default-First-Site` as the site name.

Embedded Linked Mode Features

- Does not require an external PSC, which provides a simpler infrastructure than Enhanced Linked Mode.
- Enables the use of a simplified backup and restore process based on vCenter File Based Backup and Restore.
- Enables a simplified HA process, removing the need for load balancers.
- Allows up to 15 vCenter Server Appliances can be linked.
- For a vCenter High Availability (vCenter HA) cluster, three nodes are considered one logical vCenter Server node. A single vCenter Server standard license is needed for one vCenter HA cluster.

NOTE: When vCenter High Availability is enabled for vCenter Servers using embedded linked mode and a vCenter HA failover event occurs to the passive node, if the passive node is unable to communicate the other vCenter Server node, the replica on the vCenter HA node enters read-only mode.

You can join a vCenter Server with an embedded PSC to another embedded node during deployment of the vCenter Server Appliance. For example, to deploy vCenter Embedded Linked Mode using CLI, you can use these steps.

- Configure the JSON configuration template `embedded_vCSA_on_VC.json` (or `embedded_vCSA_on_ESXi.json`) for Appliance 1 as an instance on ESXi Host 1.
- Deploy Appliance 1 by running the `vcsa-cli-installer` command.
- Configure the JSON configuration template `embedded_vCSA_replication_on_VC.json` (or `embedded_vCSA_replication_on_ESXi.json`) for Appliance 2 as an instance on ESXi Host 1
- Deploy Appliance 2 by running the `vcsa-cli-installer` command using the `embedded_vCSA_replication_on_VC.json` (or `embedded_vCSA_replication_on_ESXi.json`) file.

Infrastructure Requirements

This section describes some of the main infrastructure requirements that you should address prior to implementing vSphere.

Compute Requirements

When preparing to implement a vSphere environment you should prepare a sufficient amount of supported compute (CPU and memory) resources as described in this section.

vCenter and PSC

You can install vCenter Server 6.7 in a Windows virtual or physical server . Alternatively, you can deploy the VCSA 6.7 on a ESXi hosts 6.0 or later, which can be managed by vCenter Server 6.0 or later.

To prepare for a Windows based installation of vCenter Server or PSC, you should plan to address the compute specifications in Table 1-8

Table 1-8 Compute Specs for Windows based vCenter Server

Component	Number of CPUs	Memory
PSC	2	4
vCenter Server for a Tiny Environment Up 10 hosts or 100 virtual machines	2	10
vCenter Server for a Small Environment Up 100 hosts or 1000 virtual machines	4	16
vCenter Server for a Medium Environment Up 400 hosts or 4000 virtual machines	8	24
vCenter Server for a Large Environment Up 1000 hosts or 10,000 virtual machines	16	32
vCenter Server for a X-Large Environment Up 2000 hosts or 35,000 virtual machines	24	48

To prepare for the deployment of a VCSA you should plan to address the address the compute specifications in Table 1-9.

Table 1-9 Deployment Size Options for vCenter Server Appliance

Deployment Size Option	Details
Tiny	2 CPUs, 10 GB memory. Up 10 hosts or 100 virtual machines
Small	4 CPUs, 16 GB memory. Up 100 hosts or 1000 virtual machines
Medium	8 CPUs, 24 GB memory. Up 400 hosts or 4000 virtual machines
Large	16 CPUs, 32 GB memory. Up 1000 hosts or 10,000 virtual machines
X-Large	24 CPUs, 48 GB memory. Up 2000 hosts or 35,000 virtual machines

To prepare for the deployment of a PSC appliance, you should plan to provide it with 2 vCPUs and 4 GB memory.

NOTE: If you want to an ESXi host with more than 512 LUNs and 2048 paths then you should deploy a vCenter Server Appliance for a large or x-large environment.

ESXi

To install ESXi 6.7 ensure the hardware system meets the following requirements

- Supported system platform: For a list of supported platforms, see the *VMware Compatibility Guide* at http://www.vmware.com/resources/compatibility
- Two or more CPU cores.
- 64-bit x86 processors released after September 2006.
- The CPU's NX/XD bit must be enabled in the BIOS.
- 4 GB or more of physical RAM. (VMware recommends 8GB or more for production environments)
- To support 64-bit virtual machines, hardware virtualization (Intel VT-x or AMD RVI) must be enabled on the CPUs
- One or more supported Ethernet controllers, Gigabit or faster. For a list of supported network adapters, see the *VMware Compatibility Guide*.
- SCSI disk or a local, non-network, RAID LUN with unpartitioned space for the virtual machines.
- For Serial ATA (SATA), a disk connected through supported SAS controllers or supported on-board
- SATA controllers.

NOTE: SATA disks are considered remote, not local. These disks are not used as a scratch partition by default because they are considered remote. You cannot connect a SATA CD-ROM device to a virtual machine on an ESXi 6.7 host. To use the SATA CD-ROM device, you must use IDE emulation mode

When you use the *VMware Compatibility Guide* to identify supported system models, be sure to select the vSphere version and required features. For example, the *VMware Compatibility Guide* identifies over forty models when you select Cisco blades and ESXi 6.7 U2, but it only identifies 10 models when you add the UEFI Secure Boot feature.

For vSphere 6.7, you should ensure you meet the ESXi booting requirements.

- With Unified Extensible Firmware Interface (UEFI), you can boot systems from hard drives, CD-ROM drives, or USB media.
- VMware Auto Deploy supports network booting and provisioning of ESXi hostswith UEFI.
- Boot systems from disks larger than 2 TB if the system firmware add-in card firmware supports it per vendor documentation.

NOTE: Changing the host boot type between legacy BIOS and UEFI is not supported after you install ESXi 6.7.

Storage requirements

When preparing to implement a vSphere environment you should prepare a sufficient amount of supported storage resources as described in this section.

vCenter Server / PSC

To prepare for the deployment of vCenter Server and PSC you should plan to address their storage requirements. Table 1-10 contains the storage requirements for a VCSA with an Embedded or External Platform Services Controller. It allows for VMware Update Manager which runs as a service in VCSA.

Table 1-10 Storage Sizes for the vCenter Server Appliance

Deployment Size	Default Storage Size	Large Storage Size	X-Large Storage Size
Tiny	300 GB	825 GB	1700 GB
Small	340 GB	870 GB	1750 GB
Medium	525 GB	1025 GB	1905 GB
Large	740 GB	1090 GB	1970 GB
X-Large	1180 GB	1970 GB	2110 GB

The PSC appliance requires 60 GB.

ESXi

Installing ESXi 6.7 requires a boot device that is a minimum of 1 GB. When booting from a local disk, SAN or iSCSI LUN, a 5.2-GB disk is required to allow for the creation of the VMFS volume and a 4-GB scratch partition on the boot device. If a smaller disk or LUN is used, the installer attempts to allocate a scratch region on a separate local disk, otherwise it places `/scratch` on the ESXi host's ramdisk and links it to `/tmp/scratch`. You can reconfigure `/scratch` to use a separate disk or LUN. For best performance and memory optimization, do not leave `/scratch` on the ESXi host ramdisk. Likewise, when installing ESXi 6.7 on USB and SD devices, the installer attempts to allocate a scratch region on a local disk, otherwise it places `/scratch` on the ramdisk.

Network requirements

Networking Concepts

In order to prepare for network virtualization in vSphere, you should understand some the following concepts.

.

- Physical Network: A network of physical machines that are connected so that they can send data to and receive data from each other.
- Virtual Network: A network of virtual machines running on a physical machine that are connected logically to each other so that they can send data to and receive data from each other.
- Opaque Network: An opaque network is a network created and managed by a separate entity outside of vSphere. For example, logical networks that are created and managed by VMware NSX appear in vCenter Server as opaque networks of the type `nsx.LogicalSwitch`. You can choose an opaque network as the backing for a VM network adapter. To manage an opaque network, use the management tools associated with the opaque network, such as VMware NSX Manager or the VMware NSX API management tools
- vSphere Standard Switch: Works much like a physical Ethernet switch. It detects which virtual machines are logically connected to each of its virtual ports and uses that information to forward traffic to the correct virtual machines. A vSphere standard switch can be connected to physical switches by using physical Ethernet adapters, also referred to as uplink adapters,
- VMkernel TCP/IP Networking Layer: Provides connectivity to hosts and handles the standard infrastructure traffic of vSphere vMotion, IP storage, Fault Tolerance, and vSAN.

VMware recommends using network segmentation in vSphere environments for separating each type of VMkernel traffic and virtual machine traffic. You can implement network segments using unique VLANs and IP subnets. Here is a set of commonly used network segments in vSphere.

- Management
- vMotion
- vSphere Replication

- vSphrere High Availability heartbeat
- Fault Tolerance
- IP Storage
- Virtual Machine (typically segregated further by application or by other factors, such as test and production)

vCenter Server and PSC Network Requirements

Table 1-11 provides details for *some* of the required network connectivity involving vCenter Server and PSC. For each applicable connection, you should ensure that your network and firewall allow the described connectivity.

Table 1-11. Required Ports for vCenter Sever and PSC

Protocol / Port	Description	Required for
TCP 22	System port for SSHD.	Appliance based vCenter Server and PSC deployments
TCP 80	vCenter Server requires port 80 for direct HTTP connections. Port 80 redirects requests to HTTPS port 443.	Windows -based and appliance-based deployments of vCenter Server and PSC
TCP 88	This port must be open to join Active Directory.	Windows -based and appliance-based deployments of PSC
TCP / UDP 389	This is the LDAP port number for the Directory Services for the vCenter Server group.	Windows-based and appliance-based deployments of PSC.
TCP 443	The default port used by vCenter Server to listen for connectionsfrom the vSphere Web Client and SDK clients	Windows-based and appliance-based deployments of vCenter Server and PSC.
TCP / UDP 514	vSphere Syslog Collector port for vCenter Server on Windows and vSphere Syslog Service port for vCenter Server Appliance	Windows-based and appliance-based deployments of vCenter Server and PSC.
TCP / UDP 902	The default port that the vCenter Server system uses to send data to managed hosts.	Windows based and appliance-based deployments of vCenter Server
TCP 1514	vSphere Syslog Collector TLS port for vCenter Server on Windows and vSphere Syslog Service TLS port for vCenter Server Appliance	Windows-based and appliance-based deployments of vCenter Server and PSC.
TCP 2015	DNS Management	Windows-based and appliance-based deployments of PSC.
TCP / UDP 2020	Authentication framework management	Windows-based and appliance-based deployments of vCenter Server and PSC.
TCP 5480	Appliance Management Interface (VAMI)	Appliance-based deployments of vCenter Server and PSC.
TCP / UDP 6500	ESXi Dump Collector port.	Windows based and appliance-based deployments of vCenter Server

TCP 8084	vSphere Update Manager SOAP port used by vSphere Update Manager client plug-in.	Appliance-based deployments of vCenter Server
TCP 9084	vSphere Update Manager Web Server Port used by ESXi hosts to access host patch files from vSphere Update Manager server.	Appliance-based deployments of vCenter Server
TCP 9443	vSphere Web Client HTTPS	Windows based and appliance-based deployments of vCenter Server

ESXi Network Requirements

Table 1-12 provides details for *some* of the required network connectivity involving ESXi. For each applicable connection, you should ensure that your network and firewall allow the described connectivity.

Table 1-12. Required Ports for ESXi

Protocol / Port	**Service**	**Direction**	**Description**
TCP 5988	CIM Server	Inbound	Server for Common Information Model (CIM)
TCP 5989	CIM Secure Server	Inbound	Secure Server for CIM
UDP 8301, 8302	DVSSync	Inbound, Outbound	Used for synchronizing states of distributed virtual ports between hosts that have VMware FT record/replay enabled
TCP 902	NFC	Inbound, Outbound	ESXi uses Network File Copy (NFC) for operations such as copying and moving data between datastores.
UDP 12345, 23451	vSAN Clustering Service	Inbound, Outbound	Used by VSAN nodes for multicast to establish cluster members and distribute vSAN metadata..
UDP 68	DHCP	Inbound, Outbound	DHCP client for IPv4.
UDP 53	DNS	Inbound	DNS Client
TCP / UDP 53	DNS	Outbound	DNS Client
TCP / UDP 8200, 8100, 8300	Fault Tolerance	Inbound	Traffic between hosts for vSphere Fault Tolerance (FT).
TCP / UDP 80, 8200, 8100, 8300	Fault Tolerance	Outbound	Supports vSphere Fault Tolerance (FT).
TCP 2233	VSAN Tranport	Inbound	vSAN reliable datagram transport for vSAN storage IO.
TCP 22	SSH	Inbound	SSH Server
TCP 902, 443	vSphere Web Client	Inbound	Allows user connections from vSphere Web client
TCP / UDP 547	DHCPv6	Outbound	DHCP client for IPv6.
UDP 9	WOL	Outbound	Wake on LAN
TCP 3260	iSCSI	Outbound	Supports software iSCSI
TCP 8000	vMotion	Outbound	Supports vMotion
UDP 902	vCenter Agent	Outbound	Used by the vCenter Agent

Infrastructure Services

In addition to providing the required compute, storage, and network infrastructure, you should provide supporting infrastructure services, such as Active Directory (AD), Domain Name Services (DNS), and Network Time Protocol (NTP).

AD

In many vSphere environments, vCenter Single Sign-On (SSO) is integrated with directory services, such as Microsoft Active Directory (AD). SSO can authenticate users from its own internal users and groups, and it can connect to trusted external directory services such as AD. If you plan to leverage AD for an SSO identity source, you should ensure the proper network connectivity, service account credentials, and AD services are available and ready for use.

If you plan to install vCenter Server for Windows and use AD identity sources, you should ensure the Windows server is a member of the AD domain but is not a domain controller.

> **NOTE**: If the system that you use for your vCenter Server installation belongs to a workgroup rather than a domain, then vCenter Server is not able to discover all domains and systems available on the network when using some features.

DNS

You may wish to assign static IP addresses and resolvable fully qualified domain names (FQDNs) to your vSphere components, such as vCenter Server, PSC, and ESXi hosts. Before installing these components, you should ensure that the proper IP addresses and FQDNs entries are registered in your Domain Name System (DNS). You should configure forward and reverse DNS records.

For example, prior to installing vCenter Server for Windows, you should assign a static IP address and host name to the Windows server. The IP address must have a valid (internal) domain name system (DNS) registration. During the vCenter Server installation, you must provide the FQDN or the static IP of the Windows (host) machine. VMware recommends using the FQDN. You should ensure that DNS reverse lookup returns the appropriate FQDN when queried with the IP address of the Windows (host) machine. Otherwise, the installation of the Web Server component that supports the vSphere Web client fails.

When you deploy the vCenter Server Appliance, the installation of the Web server component that supports the vSphere Web Client fails if the installer cannot look up the fully qualified domain name (FQDN) for the appliance from its IP address. Reverse lookup is implemented using PTR records. If you plan to use an FQDN for the appliance system name, you must verify that the FQDN is resolvable by a DNS server.

Starting with vSphere 6.5, vCenter Server supports mixed IPv4 and IPv6 environment. If you want to set up the vCenter Server Appliance to use an IPv6 address version, use the fully qualified domain name (FQDN) or host name of the appliance.

Ensure that each vSphere Web Client instance and each ESXi host instance can successfully resolve the vCenter Server FQDN.

Ensure that the ESXi host management interface has a valid DNS resolution from the vCenter Server and all vSphere Web Client instances. Ensure that the vCenter Server has a valid DNS resolution from all ESXi hosts and all vSphere Web Clients.

NTP

For topologies with external PSC instances, ensure that you provide time synchronization between the nodes. All vCenter Server instances, PSC instances, and third-party load balancers in the vCenter Single Sign-On domain must be time synchronized. ESXi hosts must be time synchronized to support features such as vSphere HA. In most environments, you should plan to use NTP sever for time synchronization. Prior to implementing vSphere, verify that the NTP servers are running and available.

Be prepared to provide the names or IP addresses for the NTP servers when installing vSphere components, like vCenter Server, ESXi, and PSC. For example, during Stage 2 of the deployment of a vCenter Server Appliance with embedded PSC, you can choose to **synchronize time with NTP servers** and provide a list of NTP server name or IP addresses separated by commas. Alternatively, you choose to allow the appliance to **synchronize time with the ESXi host**.

> **NOTE**: If a vCenter Server Appliance is set for NTP time synchronization, it ignores its `time_tools-sync` Boolean parameter. Otherwise, if the parameter is TRUE, VMware Tools synchronizes the time in the appliance's guest OS with the ESXi host.

Other Requirements

This section describes a few additional requirements for a few of the optional components (see Table 1-3), and add-on products (see Table 1-4).

Additional Requirements

Here are a few requirements for some specific, commonly used vSphere features.

vSphere Web Client

The vSphere Web Client 6.7 requires Adobe Flash Player version 16 to version 23. For best performance and the most recent security updates, use Adobe Flash Player 23. For Windows users, VMware supports Microsoft Internet Explorer 10.0.19 and later, Mozilla Firefox 39 and later, and Google Chrome 34 and later. For Mac users, VMware supports Mozilla Firefox 39 and later, and Google Chrome 34 and later. For best performance, VMware recommends Google Chrome.

vCenter Server File-Based Backup and Restore

If you plan to schedule file-based backups using the VAMI, you must prepare a FTP, FTPS, HTTP, HTTPS, or SCP server with sufficient disk space to store the backups.

GUI Installer

You can use the GUI installer to interactively install a vCenter Server Appliance or PSC. To do so, you must run the GUI deployment from a Windows, Linux, or Mac machine that is in the network on which you want to deploy the appliance.

Distributed Power Management (DPM)

DPM requires the ability to wake a host from standby mode, which means it needs the ability to send a network command to the host to power on. For this feature, DPM requires iLO, IPMI, or a Wake On LAN network adapter to be present in each participating host in the cluster. DPM must be supplied with the proper credentials to access the interface and power on the host.

vSphere Replication Requirements

In order to use vSphere Replication 8.1, you must deploy a vSphere Replication Management Service (VRMS) appliance. Optionally, you can add nine additional vSphere Replication Service (VRS) appliances. You should plan for the compute, storage, and network needs of these appliances.

The VRMS appliance requires 2 vCPUs and 8 GB memory. Optionally, you can configure it for 4 vCPUs. Each VRS appliance requires 2 vCPUs and 716 MB memory. The amount of CPU and memory resources consumed by the vSphere Replication Agent on each host is negligible.

Each VRMS and VRS appliance contains two virtual disks whose sizes are 13 BG and 9 GB. To thick provision these virtual disks, you must provide 22 GB storage. If you do not reserve the memory, you should provide storage for the VRMS (8 GB) and VRS (716 MB each) swap files.

Each appliance has at least one network interface and requires at least one IP address. Optionally, you can use separate network connections to allow each appliance to separate it management and replication traffic.

The main storage requirement for vSphere Replication is to support the target datastore to where the VMs will be replicated. At a minimum in the replication target datastore, you should provide enough storage to replicate each virtual disk, to support each replicated VM's swap file, and to store each VMs multiple point in time captures (snapshots).

SDDC Requirements

To build a Software Defined Data Center (SDDC), you may plan to implement additional VMware products, such as VSAN, NSX, and vRealize Suite. Here are some of the requirements you should address.

VSAN

When preparing to implement VSAN, verify that the ESXi hosts meet the vSAN hardware requirements. All capacity devices, drivers, and firmware versions in your vSAN configuration must be certified and listed in the vSAN section of the *VMware Compatibility Guide*.

Table 1-13 provides the storage device requirements for VSAN hosts.

Table 1-13. Storage Device Requirements for VSAN Hosts

Component	Requirements
Cache	One SAS or SATA solid-state disk (SSD) or PCIe flash device.
Virtual Machine Data Storage	For hybrid group configuration, ensure that at least one SAS or NL-SAS magnetic disk is available.

	For all-flash group configuration, ensure that at least one SAS, or SATA solid-state disk (SSD), or PCIe flash device is available.
Storage Controllers	One SAS or SATA host bus adapter (HBA), or a RAID controller that is in passthrough mode or RAID 0 mode.

Prepare a network for VSAN traffic. This is the network in which you will connect a VMkernel network adapter for each ESXi host. For non-stretched VSAN clusters the network should provide a maximum Round Trip Time (RTT) of 1 ms.

NSX

When preparing to implement NSX, ensure that you address the hardware and network latency requirements.

A typical NSX implementation involves deploying an NSX Manager, three NSX Controllers, and one or more NSX Edges. Table 1-14 provides the hardware requirements for these devices.

Table 1-14 Hardware Requirements for NSX Appliances

Appliance	**Memory**	**vCPUs**	**Disk Space**
NSX Manager	16 GB	4 or 8	60 GB
NSX Controller	4 GB	4	28 GB
NSX Edge	Compact: 512 MB Large: 1 GB Quad Large: 2 GB X-Large: 8 GB	Compact: 1 Large: 2 Quad Large: 4 X-Large: 6	X-Large: 2.75 GB Other: 1 GB

You should ensure that the network latency is no higher than 150 ms RTT for NSX Manager connections with NSX Controllers, vCenter Server, and ESXi hosts.

vRealize Suite

The requirements for vRealize Automation, vRealize Operations, and vRealize Orchestrator are complex and beyond the scope of the VCP-DCV exam.

For example, the requirements for vRealize Operations (vROps) are greatly impacted by the planned vROps solution architecture. For a large vROps cluster you should carefully plan the number and type of nodes, the size of the nodes, and the size of remote collectors. The compute and storage requirements for a specific environment depend on many factors such as the number and type of objects in your environment, the collected data metrics, and vROps High Availability configuration.

The network connection between analytics cluster nodes must provide latency of 5 ms or less and bandwidth of at least 1 Gbs.

Objective 1.2

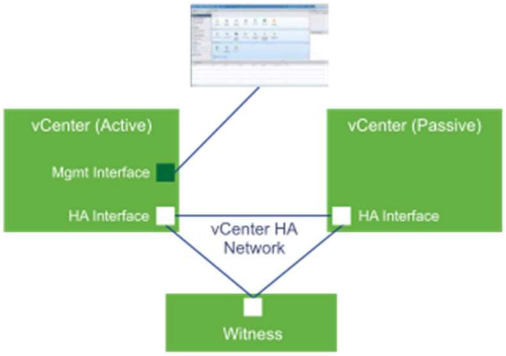
vCenter High Availability Requirements

vCenter HA is a native high availability solution for VCSA. It consists of active, passive, and witness nodes that are cloned from an existing vCenter Server instance. It includes a maintenance mode feature that prevents planned maintenance from triggering a failover. It uses a native PostgreSQL replication for the database and a separate, asynchronous file system replication for data outside of the database.

You can implement vCenter HA to protect the VCSA against host hardware, hypervisor, and application failures. In addition to an active VCSA, vCenter HA requires a passive node and a witness node. When you enable vCenter HA, the active node is cloned twice to create the passive and witness nodes. When preparing a vSphere environment for vCenter HA, you should ensure that you appropriately plan for the compute and storage resources to support all three nodes. You could estimate the total vCenter HA compute requirement by tripling the compute requirements that you initially planned for a single VCSA. You should plan for the required vCenter HA network connections.

Deploying each of the nodes on separate ESXi instances protects against hardware failure. Using ESXi hosts in a DRS cluster provides improved protection.

When vCenter HA configuration is complete, only the Active node has an active management interface (public IP). The three nodes communicate over a private network called vCenter HA network that is set up as part of configuration. The Active node is continuously replicating data to the Passive node.



The following list provides more detail on vCenter HA requirements

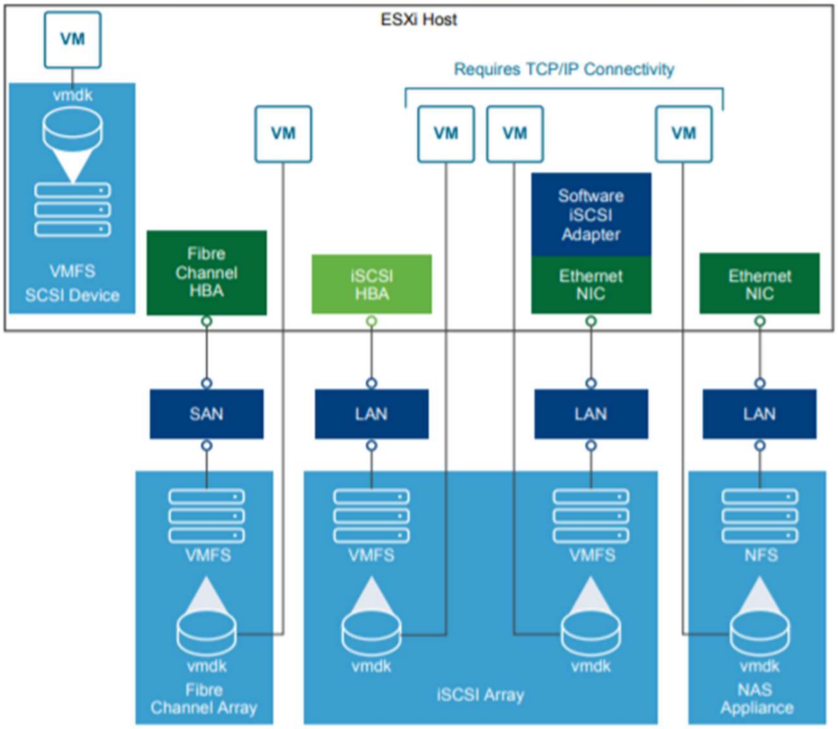
- ESXi 6.0 or later is required.
- Three hosts are strongly recommended to accommodate DRS automated VM to VM anti-affinity for the nodes.
- vCenter Server Appliance (VCSA) 6.5 is required.

- If you run the VCSA on ESXi hosts managed by a separate management vCenter Server, the management vCenter must be 5.5 or later.
- Deployment size Small (4 CPU and 16GB RAM) or bigger
- VMFS, NFS, and/or vSAN datastores.
- vCenter HA network latency between Active, Passive, and Witness nodes must be less than 10 ms
- The vCenter HA network must be on a different subnet than the management network.
- vCenter HA requires a single vCenter Server license.
- vCenter HA requires a Standard license.

Objective 1.3

How Virtual Machines Access Storage

A virtual machine communicates with its virtual disk stored on a datastore by issuing SCSI commands. The SCSI commands are encapsulated into other forms, depending on the protocol that the ESXi host uses to connect to a storage device on which the datastore resides.



Storage Virtualization – Traditional Model

Storage virtualization refers to a logical abstraction of physical storage resources and capacities from virtual machines. ESXi provides host-level storage virtualization. In vSphere environment, a traditional model is built around the following storage technologies and ESXi and vCenter Server virtualization functionalities

Storage device or LUN

In common ESXi vocabulary, the terms *device* and *LUN* are used interchangeably. The terms represent storage volumes that are presented to the host from a block storage system and is available to ESXi for formatting.

Virtual Disk

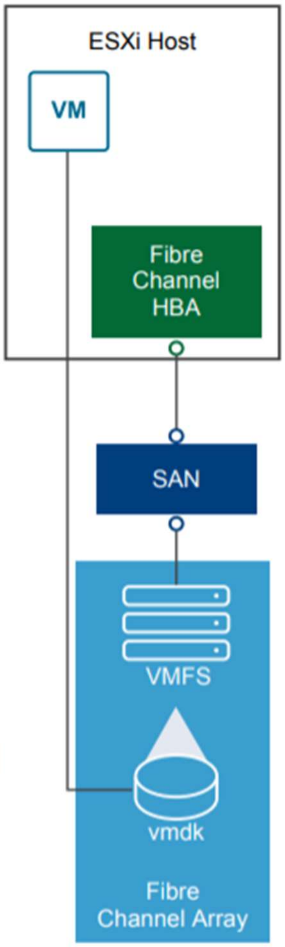
Virtual disks are sets of files that reside on a datastore that is deployed on physical storage. From the standpoint of the virtual machine, each virtual disk appears as if it were a SCSI drive connected to a SCSI controller. Whether the physical storage is transparent to the virtual machine guest operating system and applications.

Local Storage

Local storage can be internal hard disks located inside your ESXi host and external storage systems connected to the host directly through protocols such as SAS or SATA.   Local storage does not require a storage network to communicate with your host.

Fibre Channel

Fibre Channel (FC) is a storage protocol that a storage area network (SAN) uses to transfer data traffic from ESXi host servers to shared storage. It packages SCSI commands into FC frames. The ESXi host uses Fibre Channel host bus adapters (HBAs) to connect to the FC SAN. Unless you use Fibre Channel direct connect storage, you need Fibre Channel switches to route storage traffic. If your host contains FCoE (Fibre Channel over Ethernet) adapters, you can connect to your shared Fibre Channel devices by using an Ethernet network.

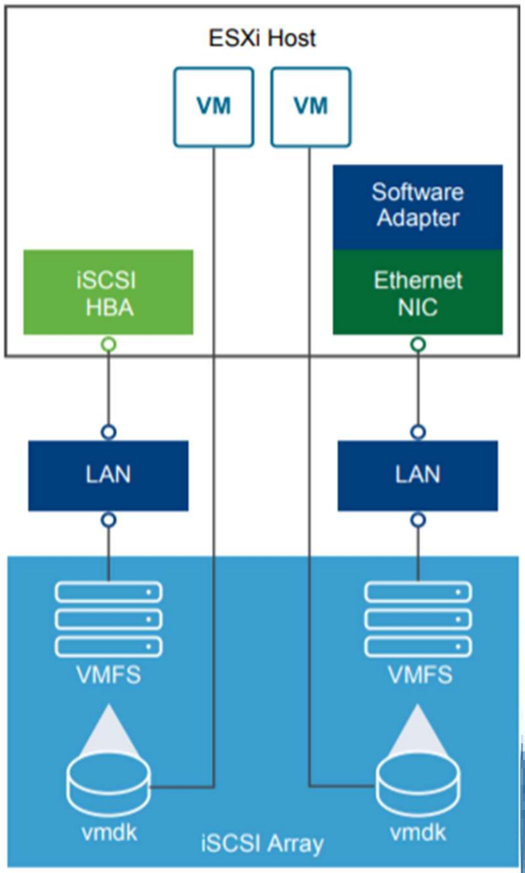


iSCSI

Internet SCSI (iSCSI) is a SAN transport that can use Ethernet connections between ESXi hosts and storage systems. To connect to the storage systems, your hosts use hardware iSCSI adapters or software iSCSI initiators with standard network adapters.

Hardware iSCSI (see the left side in the figure): The host connects to storage through a hardware adpapter that offloads the iSCSI and network processing. Hardware adapters can be dependent and independent.

Software iSCSI (see the right side in the figure): : The host uses a software-based iSCSI initiator in the VMkernel and a standard network adapter to connect to storage..

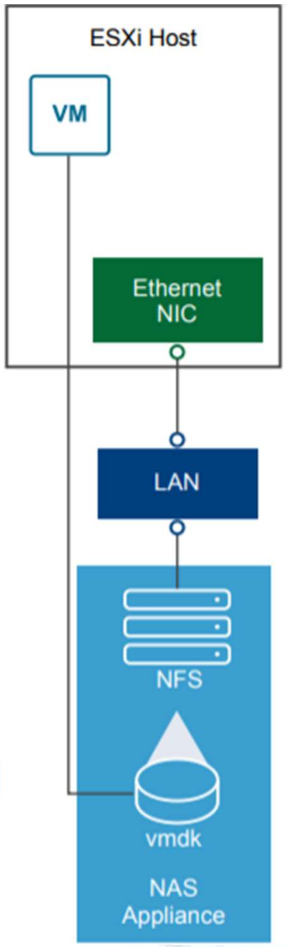


FCoE

If an ESXi host contains FCoE (Fibre Channel over Ethernet) adapters, it can connect to shared Fibre Channel devices by using an Ethernet network.

NAS / NFS

Stores virtual machine files on remote file servers accessed over a standard TCP/IP network. ESXi 6.7 uses Network File System (NFS) protocol version 3 and 4.1 to communicate with the NAS/NFS servers. You can use NFS datastores to store and manage virtual machines in the same way that you use the VMFS datastores

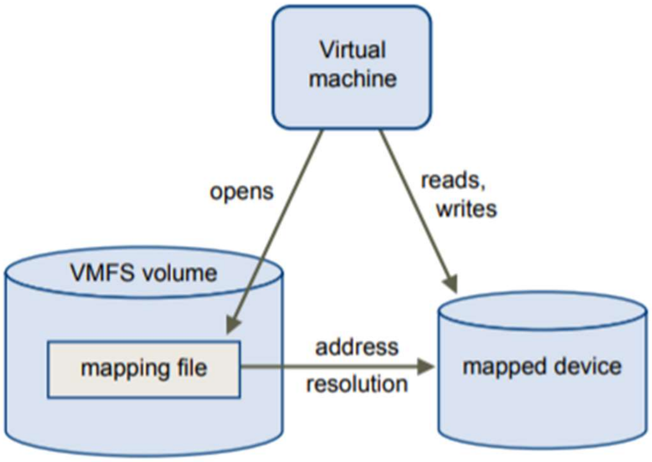


VMFS

The datastores that you deploy on block storage devices use the native vSphere Virtual Machine File System (VMFS) format. It is a special high-performance file system format that is optimized for storing virtual machines.

Raw Device Mappings (RDMs)

An RDM is a mapping file that resides in a VMFS datastore that acts as a proxy for a raw physical storage device. An RDM contains metatdata that a virtual machine uses to access the storage device directly. It gives you some of the advantages of direct access to a physical device and keeps some of the management advantages of VMFS based virtual disks.



Software Defined Storage Models

In addition to abstracting underlying storage capacities from VMs, as traditional storage models do, software-defined storage abstracts storage capabilities. With the software-defined storage model, a virtual machine becomes a unit of storage provisioning and can be managed through a flexible policy-based mechanism. The model involves the following vSphere technologies

VSAN

vSAN is a layer of distributed, based software that runs natively on each hypervisor in the cluster. It aggregates local or direct-attached capacity creates a single storage pool shared across all hosts in the vSAN cluster.

vVOLs

The Virtual Volumes functionality changes the storage management paradigm from managing space inside datastores to managing abstract storage objects handled by storage arrays. With Virtual Volumes, each virtual machine (rather than a datastore) is a unit of storage management. You can apply storage polices per virtual machine, rather than per LUN or datastore.

Storage Policy Based Management

Storage Policy Based Management (SPBM) is a framework that provides a single control panel across various data services and storage solutions, including vSAN and Virtual Volumes. Using storage policies, the framework aligns application demands of your virtual machines with capabilities provided by storage entities.

I/O Filters

I/O filters are software components that can be installed on ESXi hosts and can offer additional data services to virtual machines. Depending on implementation, the services might include replication, encryption, caching, and so on.

Objective 1.4

NIOC

VCP6-DCV Cert Guide: Chapter 3 – Objective 2.2

SIOC

VCP6-DCV Cert Guide: Chapter 5 – Objective 3.5

Objective 1.5

Provide ability to perform vSphere operations via vROps

VCP6-DCV Cert Guide: Chapter 11 – Objective 7.2 Monitor Networking and Storage Resources Using vROps

Manage logs via vRLI

Implement vCenter tags

Implement SPBM

RBAC (groups / permissions)

VCP6-DCV Cert Guide: Chapter 1 – Objective 1.1

Objective 1.6

HA

VCP6-DCV Cert Guide:

- Chapter 13 – Objective 7.5
- Chapter 15 – Objective 9.1

DRS

VCP6-DCV Cert Guide:

- Chapter 13 – Objective 7.5
- Chapter 15 – Objective 9.2

SDRS functionality

VCP6-DCV Cert Guide: Chapter 11 – Objective 7.2 – Monitor / Troubleshoot SDRS Issues

Proactive HA

Proactive HA integrates with select hardware partners to detect degraded components and evacuate VMs from affected vSphere hosts before an incident causes a service interruption.

Hardware partners offer a vCenter Server plug-in to provide the health status of the system memory, local storage, power supplies, cooling fans, and network adapters. As hardware components become degraded, Proactive HA determines which hosts are at risk and places them into a new state, called Quarantine Mode. While in Quarantine Mode, VMs are migrated to healthy hosts, as long as affinity or anti-affinity rules are not violated and there is no impact to VM performance. In addition, the affected hosts are avoided when new VMs are added to the cluster.

Predictive DRS

Predictive DRS is a new (version 6.5) feature that leverages the predictive analytics of vRealize Operations (vROps) Manager and vSphere DRS. Together, these two products can provide workload balancing prior to the occurrence of resource utilization spikes and resource contention. Nightly, vROps calculates dynamic thresholds, which are used to create forecasted metrics for the future utilization of VMs. The metrics are then passed to vSphere DRS to determine the best placement and balance of VMs before resource utilization spikes occur. Predictive DRS helps prevent resource contention on hosts that run VMs with predictable utilization patterns.

Prerequisites include the following:

- vCenter Server 6.5 or later.
- Predictive DRS must be configured and enabled in both vCenter Server and vROps.
- The vCenter Server and vROps clocks must be synchronized.

To configure Predictive DRS, use the vROps GUI to add a vCenter Server adapter instance, choose **Advanced Settings**, and select **True** in the **Provide data to vSphere Predictive DRS** drop-down menu

Objective 1.7

VCP6-DCV Cert Guide: Chapter 8 – Objective 5.1

Resource Pool Overview

How are resources applied by vSphere on resource pools and related objects?

Shares

Reservations

Limits

Use cases and examples

Objective 1.8

Physical and Virtual Network Infrastructure Comparison

Both physical switches and virtual switches are layer 2 networking devices. They can be configured with advanced functionality, such as VLAN segmentation, traffic shaping, load balancing, and more. Just like a physical switch, virtual switches have no VLAN segmentation by default. If an administrator desires, VLAN segmentation can be enabled, but this has to be specifically defined and must also match the physical switch configuration.

Virtual switches do not maintain MAC Address tables, the VMkernel keeps track of MAC Addresses, and the virtual switches forward MAC Addresses to the physical network by way of ARP.

Virtual switches can have port groups defined on them, but these are just network names, or labels by default, just like stickers on a physical switch are just labels. We could configure these port groups further, just like physical switch ports could be configured further.

The physical switch ports which ESXi hosts are attached to, should be configured as trunk ports, since ESXi hosts will be potentially sending multiple different subnets and multiple different VLANs worth of traffic through the same physical switch port.

Virtual Switch Concepts

NIC Teaming vSS and vDS

- IP Hash
- MAC based
- Virtual port based

vDS only

- Load Based Teaming

Considerations on Using Route Based on IP Hash:

- Advantages:

  - A more even distribution of the load compared to Route Based on Originating Virtual Port and Route Based on Source MAC Hash, as the virtual switch calculates the uplink for every packet.

  - A potentially higher throughput for virtual machines that communicate with multiple IP addresses.

- Disadvantages

  - Highest resource consumption compared to the other load balancing algorithms.

- The virtual switch is not aware of the actual load of the uplinks.
- Requires changes on the physical network.
- Complex to troubleshoot

Standard virtual switch (vSS)

A standard virtual switch (vSS) is a layer 2 virtual switch which each ESXi host manages and maintains. Standard virtual switches do not maintain MAC address tables, and as such are not susceptible to MAC address flooding attacks. MAC addresses are managed and maintained by the VMkernel, and forwarded to the physical switches via ARP. VMware standard virtual switches support features such as VLAN tagging (802.1q), traffic shaping, and have configurable load balancing and failover options.

However, even though these layer 2 switches have configurable features, they are not “managed switches” in that there isn’t an Operating System, and an administrator cannot connect to one remotely (except through the vSphere Client interfaces, vCLI, ESXi shell, or PowerCLI).

Every ESXi host has at least 1 virtual switch present, identified as vSwitch0. This virtual switch only has 1 physical network adapter port, or uplink, attached - the first network adapter port based on PCI bus ID. This network adapter is identified as vmnic0. No matter how many physical uplinks an ESXi host has, only 1 is attached to vSwitch0 by default, so there is no network redundancy by default. vSwitch0 is created during ESXi installation, and this virtual switch also has a VMkernel port (vmk0), named the Management Network, for connecting to the administrative agent, hostd. This network connection is how administrators manage and configure ESXi hosts after installation.

The virtual switch is also created with a second connection type on it, a virtual machine port group named VM Network. No virtual machine can attach directly to a virtual switch, and has to attach to a port group first. Since this port group is created during the installation of the ESXi host, and since there are no other virtual machine port groups by default, every VM that is created will automatically attach to this port group and have physical network access.

Distrubuted virtual switch (vDS)

A distributed virtual swicth (vDS) is a layer 2 virtual switch of which vCenter owns the configuration and then distributes this configuration to as many as 10,000 ESXi hosts.

Distributed virtual switches function in the same way as standard virtual switches, but support many more features.

Connect multiple ESXi hosts to each vDS.

Comparison of vSS / vDS

vSphere distributed switches (vDS) have numerous advantages over vSphere standard switches (vSS). They require Enterprise Plus licensing but yield many extra configuration options. Many of the Enterprise Plus features benefit large “plus-sized” enterprises, and when you look at the features and benefits that vDS has over vSS, this becomes evident.

Table 2-2 provides a side-by-side comparison of the features that are available in vSS and vDS:

Table 2-2 vSphere Standard Switches Versus vSphere Distributed Switches

Feature	vSS	vDS
Layer 2	x	x
VLAN tagging (802.1q)	x	x
IPv6	x	x
NIC Teaming	x	x
Outbound traffic shaping	x	x
Inbound traffic shaping		x
VM Network port block		x
Private VLANs		x
Load-based teaming		x
Datacenter level management		x
Network vMotion		x
Per-port policy settings		x
Port state monitoring		x
NetFlow		x

Next we explore the features specific to vDS in more detail:

- **Inbound traffic shaping:** Distributed virtual switches can do both inbound and outbound traffic shaping, whereas standard virtual switches handle just outbound traffic shaping. The settings that are available on inbound (average bandwidth in Kbps, peak bandwidth in Kbps, and burst size in KB) are available on vDS for both directions.
- **VM network port block:** On a vDS, network admins or whoever has access to the vDS can block individual vDS ports if needed. This may be useful if a virtual machine starts broadcasting a lot of traffic, as a result of a broken application, for example, and starts consuming a large portion of the network bandwidth. A network admin who has access to the vDS could block the individual switch port much as on a physical switch. One of the arguments against this is that if a virtual machine starts broadcasting traffic, why not disconnect the virtual network adapter from the port group? The problem with this argument is the assumption that the virtual network administrator (who may also be responsible for managing the physical network) has access to modifying virtual machine hardware. This might be something that is not desired, especially in large organizations with delegated control over different components within vSphere.
- **Private VLANs:** Private VLANs are an extension of the VLAN standard. They are not double encapsulation but do allow a VLAN to effectively be subdivided into other VLANs. This is useful, for example, for a hosting provider that has run out of VLANs or any environment where 4094 VLANs is not enough. vSphere supports private VLANs (and has since vSphere 4.0) because there are many organizations that employ private VLANs. vSphere administrators need to work with the networking teams to implement private VLANs because they affect traffic on the entire physical network as well.

  A VLAN that is to be subdivided becomes known as the primary private VLAN. This primary PVLAN is then carved up into one or multiple secondary PVLANs that exist only within the primary. When a virtual machine or

VMkernel port sends a packet, that packet is tagged at the dvPort group level on the vDS. Because this is not double encapsulation, packets only travel with one VLAN tag at a time. However, physical switches could be confused by seeing MAC addresses tagged with more than one VLAN tag, unless the physical switches are PVLAN aware and have their PVLAN tables configured appropriately. If the physical network is configured correctly, it identifies that the secondary PVLAN exists as part of the primary. This is defined in PVLAN tables on the physical networking switches themselves.

- **Load-based teaming:** In addition to the three load-balancing mechanisms that standard virtual switches offer (originating virtual port ID, source MAC hash, and source and destination IP hash), vDS offers a fourth load-balancing mechanism, load-based teaming, which does load balancing based on actual physical network adapter load. The VMkernel tracks the send and receive traffic for each physical uplink on the team and determines the load average in 30-second increments. Load-based teaming offers lower overhead than IP-based load balancing as well, and it doesn't require the physical switches to support 802.3ad (Etherchannel/link aggregation).
- **Datacenter-level management:** vDS makes it easy to both create virtual switches and update the configuration on existing distributed virtual switches. vDS can be pushed down to as many as 1000 ESXi servers, demonstrating the capability of updating and configuring networking for large environments.
- **Network vMotion:** A virtual machine can effectively maintain its switch connection when migrating to a different ESXi server host. The vDS port number is still the same, as are the policies and statistics that are tied to that switch port. As far as the operating system is concerned, the virtual machine hasn't moved. When you perform a migration of a virtual machine using standard virtual switches, the policies and statistics change in favor of the new switch that the VM gets connected to after the migration (and the VM gets attached to a new switch port on the second switch).
- **Per-port policy settings:** In addition to the ability to define security, traffic shaping, and teaming and failover policies at the virtual switch and port group level, with vDS you can define these policies for individual switch ports— effectively for individual virtual machines. As an administrator, you can allow or deny the ability to define per-port policies at the port group level on a vDS.
- **Port state monitoring:** In the vSphere Client or Web Client, administrators can see not only which vDS ports are in use but also some general information about the ports, such as the current state of the port (link up, link down, blocked, or information unavailable).
- **NetFlow:** NetFlow allows administrators to forward network flows—internal VM-to-VM traffic flows, or VM–to–physical device traffic flows and vice versa—to a centralized NetFlow collector. This collector can be a physical device on the network or a virtual machine. In either case, the collector is gathering the network traffic flows for the purpose of analyzing usage patterns for network monitoring and troubleshooting. Network flows can be sent at a sampled rate, and there is also the possibility of sending unsampled data as well.
- **Port mirroring:** Port mirroring allows administrators to duplicate everything that is happening on one virtual switch port to then be visible on a completely different virtual switch port. The vSphere Web Client allows you to create a mirroring session and determine the port to be mirrored and on which switch port to duplicate the traffic.
- **Network I/O Control:** NIOC allows prioritization of network traffic if there is contention on the network. This setting is enabled by default, and administrators can create network resource pools. There are several network resource pools configured by default: one for each different type of VMkernel network traffic, and one for all of the virtual machines. Administrators can create additional network resource pools for varying levels of priorities for virtual machines as needed. Network resource pools prioritize network traffic by utilizing a share mechanism (the same mechanism used for prioritization in the event of CPU, memory, or storage contention). They can also be used to define limits, if needed.

Advanced Policies and Features

Objective 1.9

vSphere cluster overview

DRS cluster

https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/performance/drs-enhancements-vsphere67-perf.pdf

HA cluster

https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-esxi-vcenter-server-67-availability-guide.pdf: Chapter 2 – Creating and Using vSphere HA Clusters

VSAN cluster

VCP6-DCV Cert Guide: Chapter 5 – Objective 3.2 – *VSAN topics*

Objective 1.10

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vm_admin.doc/GUID-CEFF6D89-8C19-4143-8C26-4B6D6734D2CB.html

Objective 1.11

VM migration overview

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-FE2B516E-7366-4978-B75C-64BF0AC676EB.html

Cold migrations

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-98C18721-A4B0-4BD2-96BF-1BBC29391B3E.html

Hot migrations

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-D19EA1CB-5222-49F9-A002-4F8692B92D63.html

Cross datastore migrations

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-561681D9-6511-44DF-B169-F20E6CA94944.html

Cross vCenter Server migrations

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-59C7D7FF-D17E-45BC-9145-06B2993880A2.html

vMotion overview

The state information includes the current memory content and all the information that defines and identifies the virtual machine. The memory content includes transaction data and the bits of the operating system and applications that are in the memory. The defining and identification information stored in the state includes all the data that maps to the virtual machine hardware elements. This information includes BIOS, devices, CPU, MAC addresses for the Ethernet cards, chipset states, registers, and so forth

vMotion prerequistes

Chapter 13: Trouble Issues with vMotion

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-3B41119A-1276-404B-8BFB-A32409052449.html

vMotion data flow details

Migration with vMotion occurs in three stages:

1. When the migration with vMotion is requested, vCenter Server verifies that the existing virtual machine is in a stable state with its current host.
2. The virtual machine state information (memory, registers, and network connections) is copied to the target host.
3. The virtual machine resumes its activities on the new host.

If errors occur during migration, the virtual machine reverts to its original state and location.

Storage vMotion overview

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-AB266895-BAA4-4BF3-894E-47F99DC7B77F.html

Storage vMotion prerequistes

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.vcenterhost.doc/GUID-A16BA123-403C-4D13-A581-DC4062E11165.html vLore

Extra – Differentiate between VMware cloud and virtualization technologies

Server Virtualization

VMware vSphere 6.7 is the industry leading virtualization and cloud platform. It provides virtualization (abstraction, pooling, and automation) of x86 based server hardware and related infrastructure, such as network switches. It provides live workload migrations, high availability, and efficient management at scale in a secured infrastructure.

VMware SDDC

A software defined data center (SDDC) is a data center that leverages logical infrastructure services that are abstracted from the underlying physical infrastructure. It allows any application to run on a logical platform that is backed by any x86, any storage, and any network infrastructure. Pioneered by VMware, the SDDC is the ideal architecture for private, public, and hybrid clouds. It extends virtualization concepts to all data center resources and services.

The SDDC includes compute virtualization (vSphere), network virtualization (NSX), and software defined storage (VSAN and VVOLs) to deliver abstraction, pooling and automation of the compute, network, and storage infrastructure services. It includes vRealize Automation and vRealize Operations to deliver policy based, automated management of the data center, services, and applications.

vCloud Suite and Private Clouds

VMware vCloud Suite is an enterprise-ready private cloud software suite that includes vSphere for data center virtualization and VMware vRealize Suite for cloud management platform.

VCF and Hybrid Clouds

Hybrid clouds are clouds that are a combination of private clouds, public clouds, and on-premises infrastructure. It is the result of combining any cloud solution with in-house IT infrastructure.

VMware Cloud Foundation (VCF) is the industry's most advanced hybrid cloud platform. It provides a complete set of software-defined services for compute, storage, networking, security, and cloud management to run enterprise apps in private or public environments. It delivers a simple path to the hybrid cloud by leveraging a common infrastructure and consistent operational model for on-premise and off-premise data centers.

VMC on AWS

VMware Cloud (VMC) on AWS is an integrated cloud offering jointly developed by AWS and VMware that provides a highly scalable, secure service that allows organizations to seamlessly migrate and extend their on-premises vSphere-based environments to the AWS Cloud. You can use it to deliver a seamless hybrid cloud by extending your on-premises vSphere environment to the AWS Cloud

VMware vCloud Director

VMware vCloud Director is a cloud service-delivery platform used by some cloud providers to operate and manage cloud based services. Service providers can use vCloud Director to deliver secure, efficient, and elastic cloud resources to thousands of customers.

Cloud Automation

vRealize Automation is cloud automation software that speeds up the delivery of infrastructure and application resources on-premises and in the public cloud. It provides self-service and policy based automation. Many businesses use vRealize Automation to automate processes and speed up the delivery of IT services and applications. VMware Cloud Assembly and VMware Service Broker are software as a service (SaaS) offerings that address similar use cases.

Objective 2.1

Integration overview

Integrating vCenter Server with other VMware products typically involve:

- Accessing the product using credentials having a specific minimum privilege set.
- Configuring the product to connect to vCenter Server, using credentials having a specific minimum permission to specific objects in the inventory
- Specific network type (ie. UDP or TCP) and port

Horizon View

VCP6-DCV Cert Guide: Chapter 1- Determine the Correct Permissions Needed to Integrate vCenter Server with Other VMware Products

vSphere Replication

VCP6-DCV Cert Guide: Chapter 1 – Chapter 9 - Objective 1.1: Install/Configure/Upgrade vSphere Replication

vRA

Create a vCenter Endpoint in vRA. https://docs.vmware.com/en/vRealize-Automation/7.5/com.vmware.vra.prepare.use.doc/GUID-3051E04E-C809-4E29-A3BF-51053BB2D7BC.html

NOTE: you should logon to vRA as an IaaS Administrator and provide vCenter credentials having administrator privileges.

vROps

Configure a vSphere Adapter in vROps: https://docs.vmware.com/en/vRealize-Operations-Manager/6.7/com.vmware.vcom.core.doc/GUID-19DAD6AF-7262-4655-B69F-6C665E33B52F.html

NOTE: pay attention to the required privileges for the credentials used to connect to vCenter Server. For example, The vCenter credential must have Performance > Modify intervals permission enabled in the target vCenter to collect VM guest metrics.

Objective 2.2

VCP6-DCV Cert Guide: Chapter 15 – Objective 9.1

https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-esxi-vcenter-server-67-availability-guide.pdf

Proactive HA

Proactive HA integrates with select hardware partners to detect degraded components and evacuate VMs from affected vSphere hosts before an incident causes a service interruption. Hardware partners offer a vCenter Server plug-in to provide the health status of the system memory, local storage, power supplies, cooling fans, and network adapters. As hardware components become degraded, Proactive HA determines which hosts are at risk and places them into a new state, called Quarantine Mode. While in Quarantine Mode, VMs are migrated to healthy hosts, as long as affinity or anti-affinity rules are not violated and there is no impact to VM performance. In addition, the affected hosts are avoided when new VMs are added to the cluster.

vSphere HA Orchestrated Restarts

Beginning in vSphere 6.5, you can now configure rules to orchestrate the VM start order in a cluster. If a host fails, HA will automatically attempt to restart the VMs in the specified order, even when they will be spread among multiple hosts in the cluster. To configure this, select Configure > VM/Host Groups to create at least two VM groups and select Configure > VM/Host Rules to create a VM-to-VM rule that first restarts VMs in one group and then restarts VMs in the other group. You could create multiple rules that work together. For example, you can create a rule to start VM-Group-A before VM-Group-B and create another rule to start VM-Group-B before VM-Group-C. In this example, the effective boot order is VM-Group-A, then VM-Group-B, and finally VM-Group-C.

Fault Tolerance

In vSphere 6.5, the integration between vSphere Fault Tolerance (FT) and vSphere DRS is improved to enable better placement decisions by ranking the hosts based on the available network bandwidth and by recommending the datastore in which to place the secondary VMDK files. Also, the supported network latency between the primary and secondary VMs has been greatly decreased. In vSphere 6.5, multiple port groups can now be used to increase the overall bandwidth available for vSphere FT logging traffic. This is similar to the multi-NIC feature of vSphere vMotion.

HA Admission Control

In vSphere 6.5, the default admission control setting is changed to Cluster Resource Percentage, which reserves a percentage of the total available CPU and memory resources in the cluster. For simplicity, the percentage is now calculated automatically by defining the number of host failures to tolerate (FTT). The percentage is dynamically changed as hosts are added or removed from the cluster. Another new enhancement is the Performance Degradation VMs Tolerate setting, which controls the amount of performance reduction that is tolerated after a failure. A value of 0% indicates that no performance degradation is tolerated.

Understand and Describe the Architecture of VCSA HA

In vSphere 6.5, VCSA provides a native high availability solution. The solution consists of active, passive, and witness nodes that are cloned from an existing vCenter Server instance. It includes a maintenance mode that prevents planned maintenance from triggering a failover. It uses a native PostgreSQL replication for the database and a separate, asynchronous file system replication for data outside of the database.

In most scenarios, you can deploy the VCSA HA by using the basic method, where the nodes run in a single cluster. This approach is simple and automatically creates the passive and witness nodes. When you're deploying VCSA in a DRS-enabled cluster, the basic method creates an anti-affinity rule and uses DRS to place the nodes on separate hosts.

The alternative is to deploy VCSA HA using the advanced method, where the nodes can be placed in separate clusters, separate vCenter Servers, or even separate data centers. To use this method, you must manually create the passive and witness nodes by cloning the source vCenter Server instance and then migrate the nodes to the proper location.

VCSA HA supports both embedded and external PSCs. An external PSC instance is required when there are multiple vCenter Server instances in an Enhanced Linked Mode configuration. When you're using an external PSC with VCSA HA, an external load balancer is required to provide high availability to the PSC instances.

Only the active node has an active management interface (public IP). The three nodes communicate over a private vCenter HA network. The active node runs the active vCenter Server instance, replicates data to the passive node, and communicates with the witness node. The passive node constantly receives updates from the active node and automatically takes the role of the active node during a failover. The witness provides a quorum to protect against split-brain situations.

VCSA HA provides failover when a node is lost or when key services fail. For example, the failure of a host running the active node results in a failover. A recovery time objective (RTO) of 5 minutes is expected.

VCSA HA requires ESXi 5.5 or later, small-size (or larger) VCSA 6.5, the vCenter HA network on separate subnet with less than 10ms network latency, and a single vCenter Server, Standard license. VMware recommends you run the VCSA nodes on separate ESXi hosts in a DRS cluster, which may be managed by a separate vCenter Server that is version 5.5 or later. You can place the nodes in VMFS, NFS, or VSAN datastores.

Enable and Configure VCSA HA

You can use this procedure to deploy VCSA HA in the basic configuration:

1. Deploy the first VCSA, which will become the active node.

2. On each ESXi host in the cluster where the VCSA nodes will run, add a second network (port group) for vCenter HA traffic.

3. In the vSphere Web Client, right-click the VCSA and select **vCenter HA Settings** > **Configure**.

4. Start the VCSA HA configuration, select **Basic**, and then supply IP addresses and other information for the passive and witness nodes.

5. Verify the clone operations successfully create the witness and passive nodes and verify the VCSA HA is operational.

You can use this procedure to deploy VCSA HA in the advanced configuration:

1. Deploy the first VCSA, which will become the active node.
2. On each ESXi host in the cluster where the VCSA nodes will run, add a second network (port group) for vCenter HA traffic.
3. In the vSphere Web Client, right-click the VCSA and select **vCenter HA Settings** > **Configure**.
4. Start the VCSA HA configuration, select **Basic**, and then supply IP addresses and other information for the passive and witness nodes.
5. Verify the clone operations successfully create the witness and passive nodes and verify the VCSA HA is operational.

After VCSA HA is deployed, you can perform various management tasks, such as setting up SNMP traps, configuring custom certificates, initiating a VCSA HA failover, backing up the active node, and rebooting all vCenter HA nodes.

VMware Service LCM: If a vCenter service fails, VMware Service Lifecycle Manager restarts it. VMware Service Lifecycle Manager monitors the health of services and it takes preconfigured remediation action when it detects a failure. Service does not restart if multiple attempts to remediate fail.

Objective 2.3

Securing ESXi

ESXi has many built-in security features such as CPU isolation, memory isolation, and device isolation. An ESXi host is protected with a firewall that is intended to only permit required network traffic. Starting with vSphere 6.0, ESXi hosts participate in the certificate infrastructure and, by default, are provisioned with certificates that are signed by the VMware Certificate Authority (VMCA).

Optionally, you can further harden ESXI by configuring features such as lockdown mode, certificate replacement, and smart card authentication for enhanced security. You should consider limiting direct access to ESXi hosts, using security profiles, using host profiles, and managing certificates. Additionally, you can take other security measures, such as using multiple networks to segregate ESXi network functions, configuring Smart Card Authentication, and implementing UEFI Secure Boot for ESXi hosts

Built-in Security Features

- ESXi Shell and SSH are disabled by default.
- By default, ESXi runs only services that are essential to managing its functions.
- By default, all ports that are not required for management access to the host are closed.
- By default, weak ciphers are disabled and communications from clients are secured by SSL. Default certificates created on ESXi use PKCS#1 SHA-256 with RSA encryption as the signature algorithm.
- A Tomcat Web service is used internally by ESXi to support access by Web clients. ESXi is not vulnerable to the Tomcat security issues reported in other use cases, because the service has been modified to run only functions that a Web client requires for administration and monitoring
- VMware monitors all security alerts that can affect ESXi security and issues security patches when needed.
- Secure services such as SSH and SFTP are available and should be used instead of insecure counterparts, like Telnet and FTP.
- ESXi provides the option of using UEFI Secure Boot

Security Profiles

You can customize many of the essential security settings for your host through the Security Profile panel available in the vSphere Web Client. The Security Profile is especially useful for single host management. If you are managing multiple hosts, consider using one of the CLIs or SDKs and automating the customization. You can use Security Profiles to customize services and configure the ESXi firewall.

Table 9-1 lists the services and associated ports that are typically available in ESXi 6.x. On a specific host, the list of actual services and firewall ports can be impacted by the currently installed VMware Installation Bundles (VIBs).

Table 9-1 ESXi Ports

Firewall Service	Incoming Port(s)	Outgoing Port(s)
CIM Server	5988 (TCP)	
CIM Secure Server	5989 (TCP)	
CIM SLP	427 (TCP,UDP)	427 (TCP,UDP)
DHCPv6	546 (TCP,UDP)	547 (TCP,UDP)
DVSSync	8301,8302 (UDP)	8301,8302 (UDP)
HBR		44046,31031 (TCP)
NFC	902 (TCP)	902 (TCP)
WOL		9 (UDP)
Virtual SAN Clustering Service	12345,23451 (UDP)	12345,23451 (UDP)
DCHP Client	68 (UDP)	68 (UDP)
DNS Client	53 (UDP)	53 (TCP,UDP)

Fault Tolerance	8100,8200,8300 (TCP,UDP)	80,8100,8200,8300 (TCP,UDP)
NSX Distributed Logical Router Service	6999 (UDP)	6999 (UDP)
rabbitmqproxy		5671 (TCP)
Virtual SAN Transport	2233 (TCP)	2233 (TCP)
SNMP Server	161 (UDP)	
SSH Server	22 (TCP)	
vMotion	8000 (TCP)	8000 (TCP)
VMware vCenter Agent		902 (UDP)
vSphere Web Client	902,443 (TCP)	
vsanvp	8080 (TCP)	8080 (TCP)

Change Default Account Access

One step for hardening an ESXi host is to harden the password required to use its predefined, local administrator account, which is called root. By default, the ESXi host enforces passwords for its local user accounts, which may be used to access the host via the Direct Console User Interface (DCUI), the ESXi Shell, Secure Shell (SSH) or the vSphere Client. You can modify the ESXi password requirements by setting the `Security.PasswordQualityControl` advanced option for the host. For example, you can set `Security.PasswordQualityControl` configure the ESXi host to accept pass phrases, which it does not accept by default.

Add an ESXi Host to a Directory Service

You can add an ESXi host to a directory service, such as an Active Directory, and configure permissions to allow the associated users to connect directly to the ESX host using DCUI, ESXi Shell, SSH, or the vSphere Host Client. The main reason for this is to reduce the number of local ESXi user accounts that you have to create and manage. Another reason is to provide users with the means to access ESXi directly with an existing user account whose password is already hardened.

Using vSphere Authentication Proxy

You can add ESXi hosts to an Active Directory domain by using vSphere Authentication Proxy instead of adding the hosts explicitly to the Active Directory domain. To do this, you can add the host's IP address to the vSphere Authentication Proxy access control list. By default, the vSphere Authentication Proxy will authorize the host based on its IP address. You can enable client authentication to have vSphere Authentication Proxy check the host's certificate. If you are using Auto Deploy, you can configure a reference host to point to the Authentication Proxy, setup a rule that applies the reference host’s profile to others hosts provisioned by Auto Deploy, let Auto Deploy store the host’s IP address in the access control list, and join the host to the AD domain.

Controlling ESXi Host Access

You can implement lockdown mode to force operations to be performed through vCenter Server. You can choose to use strict lockdown mode, which disables the Direct Console User Interface (DCUI) service or normal lockdown mode, which allows DCUI access for some users. In normal lockdown mode, user accounts that are in the Exception Users list and have administrator privileges on the host can access the DCUI. A common use case is to provide access to service accounts, such as backup agents. Also, in normal lockdown mode, users identified in the host's `DCUI.Access` advanced option can access the DCUI. If the ESXi Shell or SSH is enabled and the host is placed in lockdown mode, accounts in the Exception Users list who have administrator privileges can use these services. For all other users,ESXi Shell or SSH access is disabled. The main use case is to provide user access in the event of catastrophic failure. Starting with vSphere 6.0, ESXi or SSH sessions for users who do not have administrator privileges are terminated.

ESXi Host Certificates

By default in vSphere 6.x, the VMware Certificate Authority (VMCA) provisions each new ESXi host with a signed certificate that has VMCA as the root certificate authority. Provisioning happens when the host is added to vCenter Server explicitly or as part of installation or upgrade to ESXi 6.0 or later.

You can view and manage ESXi certificates using the vSphere Web Client or the vim.CertificateManager API in the vSphere Web Services SDK. You cannot view or manage ESXi certificates by using certificate management CLIs that are available for managing vCenter Server certificates.

In vSphere 6.x, vCenter Server provides three certificate modes for ESXi hosts, which are VMware Certificate Authority (default), Custom Certificate Authority, and Thumbprint Mode

In vSphere 6.x, you can view information about VMCA signed and third party signed certificate expiration in the vSphere Web Client. You can view the information for all hosts that are managed by a vCenter Server or for individual 9hosts. A yellow alarm is raised if the certificate is in the Expiring Shortly state (less than eight months). A red alarm is raised if the certificate is in the Expiration Imminent state (less than two months).

ESXi hosts that boot from installation media, have an autogenerated certificate. When a host is added to the vCenter Server system, it is provisioned with a certificate that is signed by VMCA as the root CA.

For certificate management for ESXi hosts, you must have the `Certificates.Manage Certificates` privilege. In vSphere 6.x,, a host name or IP address change might affect whether vCenter Server considers a host certificate valid. If the host was added to vCenter using the host name, rather than the IP address, then a vCenter connectivity problem occurs if the ESXi hostname changes. To remediate, you could reconnect the host to vCenter Server, or you could remove and re-add the host.

If you upgrade an ESXi 5.x host to ESXi 6.x, the upgrade process replaces the self-signed (thumbprint) certificates with VMCA-signed certificates. If the ESXi host uses custom certificates, the upgrade process retains those certificates even if those certificates are expired or invalid.

.

See Chapter 11 (Installation and Configuration) for deploying implementing SSO, configuring VMCA, and applying host profiles. See Chapter 17 (Security Management) for other ESXi security management procedures.

Apply Permissions to ESXi Hosts Using Host Profiles

You can use host profiles to configure ESXi host permissions that are applied when users access the host directly. You should apply permissions to user and group accounts that exist locally on the host or that exist in a directory service to which you added the host. See Chapter 11 for details on using host profiles.

ESXi Secure Boot and TPM

UEFI Secure Boot is a mechanism that ensures that only trusted code is loaded by the EFI firmware prior to OS handoff. When Secure Boot is enabled, the UEFI firmware validates the digitally signed kernel of an OS against a digital certificate stored in the UEFI firmware. Starting with vSphere 6.5, ESXi supports secure boot if it is enabled in the hardware. ESXi version 6.5 and later supports UEFI secure boot at each level of the boot stack.

ESXi is composed of digitally signed packages called vSphere installation bundles (VIBs). These packages are never broken open. At boot time, the ESXi file system maps to the content of those packages. By leveraging the same digital certificate in the host UEFI firmware used to validate the signed ESXi kernel, the kernel then validates each VIB using the Secure Boot verifier against the firmware-based certificate, ensuring a cryptographically “clean” boot.

When Secure Boot is enabled, ESXi will prevent the installation of unsigned code on ESXi. To install unsigned code such as beta drivers, you must disable Secure Boot. When Secure Boot is enabled, the Secure Boot verifier will run, detect the unsigned VIB, and crash the system, which produces the Purple Screen of Death (PSOD) event that identifies the VIB that must be removed. To remediate, boot the ESXi host with Secure Boot disabled, remove the VIB, and reboot with Secure Boot enabled.

ESXi can use Trusted Platform Modules (TPM) chips, which are secure cryptoprocessors that enhance host security by providing a trust assurance rooted in hardware as opposed to software. TPM is an industry-wide standard for secure cryptoprocessors. TPM chips are found in most of today's computers, from laptops, to desktops, to servers. vSphere 6.7 supports TPM version 2.0. A TPM 2.0 chip attests to an ESXi host's identity. Host attestation is the process of authenticating and attesting to the state of the host's software at a given point in time. UEFI secure boot, which ensures that only signed software is loaded at boot time, is a requirement for successful attestation.

Securing vCenter Server

To harden a vCenter Server, you should consider controlling the datastore browser access, managing the security certificates, controlling MOB access, changing default account access, and restricting administrative privileges.

Create/Manage vCenter Server Security Certificates

By default, the VMware Certificate Authority (VMCA) provisions each ESXi host, each machine in the environment, and each solution user with a certificate signed by VMCA. The environment works out of the box, but if company policy requires it, you can change the default behavior.

You can use the vSphere Certificate Manager utility to replace certificates with custom certificates. To begin, use the Certificate Manager utility to create the Certificate Signing Requests (CSRs) that you need to send to your trusted Certificate Authority (CA). You could choose to replace only the machine SSL certificates and use the solution user certificates that are provisioned by the VMCA. Solution user certificates are used only for communication between vSphere components. When using custom certificates, you must replace the VMCA-signed certificates that are provisioned by VMCA on each node with custom certificates.   You can use the vSphere Certificate Manager utility or command line interfaces to replace the certificates. Certificates are stored in the VMware Endpoint Certificate Store (VECS).

Members of the CAAdmins group have administrator privileges for VMCA.

Control MOB Access

The vCenter Server Managed Object Browser (MOB) provides a means to explore the vCenter Server object model. Its primary use is for debugging. It provides the ability to make some configuration changes, so it may be considered a vulnerability for malicious attacks. As a step to harden the vCenter Server, you can disable the MOB, by setting the `enableDebugBrowse` parameter to FALSE in the `vpxd.cfg` file. You can use the vSphere Web

Client to examine the value of `config.vpxd.enableDebugBrowe` to determine if is FALSE, but the setting is read-only in the vSphere Web Client. After making the change to the `vpxd.cfg` file, restart the vCenter Server.

Change Default Account Access

When you install vCenter Server 5.1 and later you must specify the initial vCenter Server administrator user or group. For deployments where vCenter Server and Single Sign On are deployed on the same server, you can identify a local operating system group as the initial administrator group. For example, on a Windows deployment, you can set the local Windows `Administrators` group as the initial administrator group. After the installation, you can change the default permissions in vCenter Server to assign the `Administrator` role to at least one other user account or group and then delete the default permission.

In situations, where an external Platform Services Controller (PSC) is used, then a local operating system cannot be used. Instead, you should assign the `Administrator` role to user groups provided by SSO identity sources, such as the SSO domain (such as `vsphere.local`), an Active Directory domain, or an OpenLDAP domain.

Restrict Administrative Privileges

As mentioned in the previous paragraphs, you can use permissions to assign the `Administrator` role to specific users and groups in vCenter Server. You should restrict the assignment of the `Administrator` role to just those users and groups, who truly require it. You should use roles with only necessary privileges when creating permissions. In other words, you should apply the principle of least privileges when configuring permissions in vCenter Server.

Control Datastore Browser Access

Assign the **Datastore.Browser** datastore privilege only to users and user groups who truly require the privilege.

Securing vSphere Networking

You can use firewalls, segmentation, VLANs, and other measures to secure the networks used by your virtual machines and the vSphere environment.

Firewalls

You can use traditional (physical) firewalls, virtual machine based firewalls, and hypervisor based firewalls (like NSX distributed firewall) to protect inbound and outbound traffic to the vCenter Server, ESXi hosts, virtual machines, and other vSphere components. Ideally, you could use firewalls to only allow required traffic between specific vSphere components, virtual machines, and network segments.

Segmentation and Isolation

You should keep different virtual machine zones within a host on different network segments to reduce the risk of data leakage and threats. Such threats include Address Resolution Protocol (ARP) spoofing, where an attacker manipulates the ARP table to remap MAC and IP addresses, and gains access to network traffic to and from a host. Attackers use ARP spoofing to generate man in the middle (MITM) attacks, perform denial of service (DoS) attacks, and hijack the systems. You can implement segmentation by using one of two approaches.

- Use separate physical network adapters for virtual machine zones, which is probably the most secure method.

- Set up virtual local area networks (VLANs) for virtual machine zones, which is typically the most cost-effective method. It provides almost all the security benefits as using physically separate networks, but without the hardware overhead.

You should isolate the vSphere management network, which provides access to the management interface on each component. Isolation is important because attackers who gain access to this network can use it as a staging ground for further attacks. In most cases, you should place the vSphere management port group in a dedicated VLAN and ensure that the network segment is not routed, except to other management-related networks. Likewise, you should isolate IP-based storage traffic and vMotion traffic.

Internet Protocol Security

You can configure security Internet Protocol Security (IPsec) on ESXi hosts to enable authentication and encryption of incoming and outgoing packets. You can configure security associations to control *how* the system encrypts the traffic. For each association, you configure a name, source, destination, and encryption parameters. You can configure security policies to determine *when* the system should encrypt traffic. Security policies include information such as source, destination, protocol, direction, mode, and a security association.

To list the available security associations, you can use this command.

```
esxcli network ip ipsec sa list
```

To add a security association, you can use the `esxcli network ip ipsec sa add` with one or more options from Table 9-2.

Table 9-2 IPsec Options

Option	**Description**
--sa-source= *source address*	Required. Specify the source address.
--sa-destination= *destination Address*	Required. Specify the destination address.
--sa-mode= *mode*	Required. Specify the mode, either transport or tunnel
--sa-spi= *security parameter index*	Required. Specify the security parameter index as a hexadecimal
--encryption-algorithm= *encryption algorithm*	Required. Specify the algorithm as one of the following parameters. • `3des-cbc` • `aes128-cbc` • `null` (no encryption)
--integrity-algorithm= *authentication algorithm*	Required. Specify the authentication algorithm, either `hmac-sha1` or `hmac-sha2-256.`
--integrity-key= *authentication key*	Required. Specify the authentication key. You can enter keys as ASCII text or as a hexadecimal.

--sa-name=*name*	Required. Provide a name for the security association.

General Networking Security Recommendations

Here are other general networking security recommendations.

- If spanning tree is enabled, ensure that physical switch ports are configured with Portfast.
- Ensure that Netflow traffic for a Distributed Virtual Switch is only sent to authorized collector IP addresses.
- Ensure that only authorized administrators have access to virtual networking components by using the role-based access controls.
- Ensure that port groups are not configured to the value of the native VLAN.
- Ensure that port groups are not configured to VLAN values reserved by upstream physical switches.
- Ensure that port groups are not configured to VLAN 4095 except for Virtual Guest Tagging (VGT).
- Restrict port-level configuration overrides on a distributed virtual switch. Port-level configuration overrides are disabled by default.
- Ensure that distributed virtual switch port mirror traffic is sent only to authorized collector ports or VLANs..

Network Security Policies

You should connect virtual machines to standard virtual switch port group or distributed virtual switch port group that are configured with an appropriate security policy. The network security policy provides three options, which may be set to `Reject` or `Accept`, as described in Table 9-3.

Table 9-3 Network Security Policies

Guideline	**Setting**	**Description**
Promiscuous Mode	Accept	The virtual switch forwards all frames to the virtual network adapter.
	Reject	The virtual switch forwards only the frames that are address to the virtual network adapter.
MAC address changes	Accept	If the guest operating system changes the effective MAC address of the virtual adapter to a value that differs from the MAC address assigned to the adapter in the VMX file, the virtual switch allows the inbound frame to pass.
	Reject	If the guest operating system changes the effective MAC address of the virtual adapter to a value that differs from the MAC address assigned to the adapter in the VMX file, the virtual switch drops all inbound frames to the adapter. If the guest OS changes the MAC address back to its original value, then the virtual switch will stop dropping the frames and allow inbound traffic to the adapter.

Forged Transmits	Accept	The virtual switch does not filter outbound frames. It permits all outbound frames, regardless of source MAC address.
	Reject	The virtual switch drops any outbound frame from a virtual machine virtual adapter that uses a source MAC address that differs from the MAC address assigned to the virtual adapter in the VMX file.

On a distributed virtual switch, you can override the security policy per virtual port.

Securing Virtual Machines

To harden a virtual machine, you could follow best practices, configure UEFI, control the VMware Tools installation, implement security policies, protect against denial of service attacks, and implement encryption.

Virtual Machine Hardening Best Practices

General Protection – In most respects, treat the virtual machine as you would a physical server when it comes to applying security measures. For example, be sure to install guest operating systems patches, protect with anti-virus software and disable unused serial ports.

Templates – Carefully harden the first virtual machine deployment of each guest O/S and verify hardening completeness. Convert the virtual machine into a template and use the template to deploy virtual machines as needed.

Virtual machine console – Minimize the use of this console. Only use it when required. Use remote tools, such as SSH and Remote Desktop to access virtual machines. Consider limiting the number of console connections to just one.

Virtual machine resource usage – Prevent virtual machines from taking over resources on the ESXi host to minimize the risk of Denial of Service to other virtual machines. Configure each virtual machine with enough virtual hardware, but not much more virtual hardware resources than needed. For example, configure each virtual machine with a sufficient amount of virtual memory to handle its workload and meet application vendor recommendations, but do not provide much more memory than you expect it will need. Consider setting reservations or shares to ensure that critical virtual machines have access to a sufficient amount of CPU and memory.

Disable unnecessary services – Disable or uninstall any function for the guest O/S that is not required to reduce the number of components that can be attacked and to reduce its resource demand. For example, turn off screen savers, disable unneeded guest operating system services, and disconnect the CD/DVD drive.

Configure UEFI boot

Starting with vSphere 6.5, If the operating system supports secure UEFI boot, you can configure your VM to use UEFI boot. Prerequisites are UEFI firmware, Virtual hardware version 13 or later, VMware Tools version 10.1 or later, an operating system that supports UEFI secure boot. For Linux virtual machines, VMware Host-Guest Filesystem is not supported in secure boot mode and should be removed from VMware Tools before you enable secure boot. If you turn on secure boot for a virtual machine, you can load only signed drivers into that virtual machine.

Control VMware Tools

After upgrading an environment to vSphere 6.x, you should consider upgrading virtual machines. The first step in upgrading virtual machines is to upgrade VMWare Tools to the most recent version that is compatible with the ESXi host. VMware provides two methods for upgrading virtual machines. One is to use the vSphere Web Client. The other is to user VMware Update Manager.

NOTE: Do not use `vmware-vmupgrade.exe` to upgrade virtual machines.

When using the vSphere Web Client method, you could manually upgrade each virtual machine, one by one, as needed. To install a VMware Tools upgrade, you can use the same procedure that you initially used to installing VMware Tools. Upgrading VMware Tools actually installs a new version. To manually upgrade a set of virtual machines, you can select an ESXi host or cluster in the inventory, use the **Related Objects** > **Virtual Machines** tab to select a set of virtual machines, and choose **Guest OS** > **Install / Upgrade VMware Tools** in the **Actions** menu.

Or, you can configure virtual machines to automatically check and install newer versions of VMware Tools, as needed. As each virtual machine is started, the guest operating system checks the version of VMware Tools. The virtual machine status in the vSphere Web Client displays a message when a new version is available.

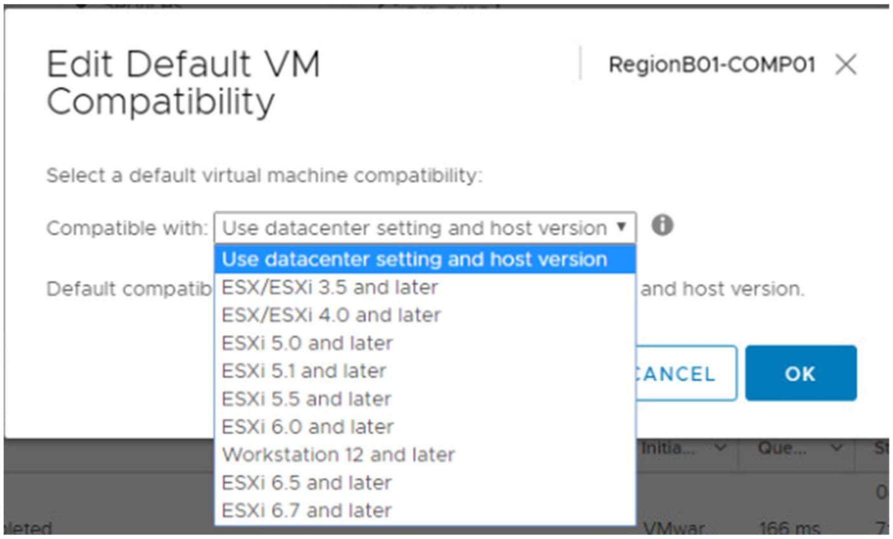
In Windows virtual machines, you can set VMware Tools to notify the Windows user when an upgrade is available, by placing a yellow caution icon with the VMware Tools icon in the Windows taskbar.

For Windows and Linux guest operating systems, you can configure the virtual machine to automatically upgrade VMware Tools. Although the version check is performed when you power on the virtual machine, on Windows guest operating systems, the automatic upgrade occurs when you power off or restart the virtual machine. The status bar displays the message `Installing VMware Tools` when an upgrade is in progress.

For Linux guest operating systems, when you upgrade VMware Tools, new network modules are available, but not used until you restart the virtual machine or reload the associated networking kernel modules. This approach avoids network interruptions and allows you to install VMware Tools over SSH, but it limits the benefit of automatically upgrading VMware Tools without restarting the virtual machine.

Using VMware Update Manager, you can perform an orchestrated upgrade of virtual machines at various levels, such as a folder or datacenter level. You may want to ensure that VM compatibility is not just based on its ESXi host, but also based on datacenter settings. As shown in Figure 9-1, at a cluster level, you can set the Default VM Compatibility setting to **Use datacenter setting and host version**. Alternatively, you could specify a specific VM Compatibility for the cluster, such as `ESXi 5.5 and later`.

Figure 9-1 Edit Default VM Compatibility



Edit Default VM Compatibility

RegionB01-COMP01

Select a default virtual machine compatibility:

Compatible with: Use datacenter setting and host version

Default compatib... and host version.

Virtual machines provide settings that impact VMware Tools. In many cases, you should change a virtual machine's VMware Tools settings to provide better security. You can use the vSphere Web Client, vSphere Client Power CLI or a text editor to change the VMware Tools settings. When using a text editor, you need to modify the virtual machine configuration file (VMX) directly. Changes to the following VMware Tools settings should be considered to address specific, potential security threats.

Disk shrinking: Because disk shrinking, which reclaims unused disk space from a virtual machine, can take considerable time to complete and its invocation can result in a temporary denial of service, disable disk shrinking using the following lines in the VMX file

```
isolation.tools.diskWiper.disable = "TRUE"

isolation.tools.diskShrink.disable = "TRUE"
```

Copy and paste: This ability is disabled by default in new virtual machines. In most cases, retain this default setting to ensure that one user of the virtual machine console cannot paste data that was originally copied from a previous user. Ensure that the following lines remain in the VMX files

```
isolation.tools.copy.disable = "TRUE"

isolation.tools.paste.disable = "TRUE"
```

Connecting devices: By default, the ability to connect and disconnect devices is disabled. One reason is to prevent one user from accessing a sensitive CD-ROM device that was left in the drive. Another reason is to prevent users from disconnecting the network adapter, which could produce a denial of service. Ensure that the following lines remain in the VMX file.

```
isolation.device.connectable.disable = "TRUE"

isolation.device.edit.disable = "TRUE"
```

Logging: Uncontrolled virtual machine logging could lead to denial of service if the associated datastore runs out of disk space. VMware recommends keeping 10 log files. To set this on a virtual machine, set the following in the VMX file

```
vmx.log.keepOld = "10"
```

Alternatively, to limit the number of log files for virtual machines on an ESXi host, add the previous line to the host's `/etc/vmware/config` file. A more aggressive measure is to disable virtual machine logging with the following statement in the VMX file

```
logging = "FALSE"
```

VMX file size: By default, the size of each VMX file is 1 MB, because uncontrolled file sizes can lead to a denial of service if the datastore runs out of disk space. Occasionally, *setinfo* messages that define virtual machine characteristics or identifiers are sent as name-value pairs from the virtual machine to the VMX file. If needed, you can increase the size of the VMX file limit by using the following statement in the VMX file, but replacing the numeric value with a larger value. But in most cases, keep the default setting as a security measure.

```
tools.setInfo.sizeLimit = "1048576"
```

Performance counters: VMware Tools provides performance counters on CPU and memory from the ESXi host into the virtual machine for use by PerfMon. This feature is disabled by default, because an adversary could potentially make use of this information to attack the host. Ensure the following line remains in the VMX files, which blocks some, but not all performance metrics.

```
tools.guestlib.enableHostInfo = "FALSE"
```

Unexposed vSphere features: Because VMware Tools is used in many VMware product, not just vSphere, the VMX file may contain parameters that do not apply in a vSphere environment. To secure the virtual machine and reduce the number of vectors through which a guest operating system could access an ESXi host, implement the following lines in the VMX file

```
isolation.tools.unity.push.update.disable = "TRUE"

isolation.tools.ghi.launchmenu.change = "TRUE"

isolation.tools.ghi.autologon.disable = "TRUE"

isolation.tools.hgfsServerSet.disable = "TRUE"

isolation.tools.memSchedFakeSampleStats.disable = "TRUE"

isolation.tools.getCreds.disable = "TRUE"
```

Configure Virtual Machine Security Policies

VMware provides the vSphere 6.0 Hardening Guide that provides guidelines for address vulnerabilities based on risk profiles. When you can apply the hardening guide to your environment, the first step is to apply the appropriate risk profile based on the sensitivity of your environment and data. The hardening guide offers three risk profiles:

Risk Profile 1: Intended to be implemented in just the most secure environments, such as top-secret government environments.

Risk Profile 2: Intended to be implemented in sensitive environments to protect sensitive data such as those that must adhere to strict compliance rules.

Risk Profile 3: Intended to be implemented in all production environments.

For example, the Table 9-4 contains the Risk Profile 3 guidelines from the hardening guide that directly impact virtual machines. Many of these guidelines are associated with the VMware Tools Configuration Parameters discussed previously in this chapter.

Table 9-4 Risk Profile 3 Virtual Machine Guidelines

Guideline	Configuration Parameter
VM.disable-console-copy	isolation.tools.copy.disable

VM.disable-console-drag-n-drop	isolation.tools.dnd.disable
VM.disable-console-gui-options	isolation.tools.setGUIOptions.enable
VM.disable-console-paste	isolation.tools.paste.disable
VM.disable-disk-shrinking-shrink	isolation.tools.diskShrink.disable
VM.disable-disk-shrinking-wiper	isolation.tools.diskWiper.disable
VM.limit-setinfo-size	tools.setInfo.sizeLimit
VM.minimize-console-VNC-use	RemoteDisplay.vnc.enabled
VM.prevent-device-interaction-connect	isolation.device.connectable.disable
VM.prevent-device-interaction-edit	isolation.device.edit.disable
VM.TransparentPageSharing-inter-VM-Enabled	sched.mem.pshare.salt
VM.verify-network-filter	ethernetn.filtern.name = filtername

For vSphere 6.7, the vSphere Hardening Guide is replaced with the *vSphere 6.7 Update 1 Security Configuration Guide*. The risk profiles are removed, primarily because only the only remaining Risk Profile 1 setting is `ESXi.enable-strict-lockdown-mode`. Instead of identifying risk profiles, the new guide simply lists the current 50 Guideline IDs alphabetically and includes a Vulnerability Discussion for each guideline. You can review the *vSphere 6.7 Update 1 Security Configuration Guide* https://bit.ly/2Zbg1H0.

Protect Virtual Machine Against Denial-of-Service Attacks

You can limit informational messages from the virtual machine to the VMX file to avoid filling the datastore and causing a Denial of Service (DoS). If you do not control the size of a virtual machine's VMX file and the amount of information exceeds the datastore capacity, a DoS can occur. The virtual machine configuration file (VMX file) limit is 1 MB by default and is applied even when the `tools.setInfo.sizeLimit` parameter is not listed in the virtual machine's advanced options.

Virtual Machine Communication Interface (VMCI) is a high-speed communication mechanism for virtual machine to ESXi host communication. In some VMware products, including ESXi 4.x, VMCI also provides high-speed communication between virtual machines on the same ESXi host. In ESXi 5.1, the guest to guest VMCI is removed. In a VMX file, the `vmci0.unrestricted` parameter is used to control VMCI isolation for virtual machines running on ESX/ESXi 4.x and ESXi 5.0, but has no effect on virtual machines running on ESXi 5.1 and later.

Non-administrative users in the guest operating system can shrink virtual to reclaim the disk's unused space. However, if you shrink a virtual disk repeatedly, the disk can become unavailable and cause a denial of service. To prevent this, you could disable the ability to shrink virtual disks using the following steps.

1. Shutdown the virtual machine
2. Modify the advanced settings in the virtual machine options.

3. Set `isolation.tools.diskWiper.disable` and `isolation.tools.diskShrink.`disable to `TRUE`

Control VM Device Connections

As discussed previously in this chapter, the ability to connect and disconnect devices is disabled by default for new virtual machines. In most cases, you should not change this behavior. You should verify that the following setting exist in your VMX files, especially if the virtual machines were deployed from a non-hardened template or were originally built on older ESXi hosts.

```
isolation.device.connectable.disable = "TRUE"

isolation.device.edit.disable = "TRUE"
```

If these parameters are set to FALSE, then in a guest operating system, all users and processes, with or without root or administrator privileges could use VMware Tools to change device connectivity and settings. They could connect or disconnect devices, such as network adaptors and CD-ROM drives. They could modify device settings. This functionality could allow them to connect a CD-ROM with sensitive data. It could allow them to disconnect a network adapter, which could cause a denial of service to other users.

Virtual Machine Encryption

Starting with vSphere 6.5, you can protect your virtual machines, virtual disks, and other virtual machines files using virtual machine encryption. You need to set up a trusted connection between vCenter Server and a key management server (KMS). vCenter Server can then retrieve keys from the KMS as needed. You can encrypt an existing virtual machine or virtual disk by changing its storage policy. Encryption works with any guest OS, because encryption occurs at the hypervisor level. Encryption keys and configuration are not contained in the VM guest OS.

You can encrypt virtual disks only for encrypted virtual machines. You cannot encrypt the virtual disk of an unencrypted VM. If you are using the vSphere Client to create an encrypted virtual machine, you can encrypt and decrypt virtual disks separate from virtual machine files. If you are using the vSphere Web Client to create an encrypted virtual machine, all virtual disks are encrypted by default. For other encryption tasks, for both clients, such as encrypting an existing virtual machine, you can encrypt and decrypt virtual disks separate from virtual machine files, using the following procedure.

Prerequisites

- Establish a trusted connection with the KMS and select a default KMS.
- Create an encryption storage policy, or use the bundled sample, VM Encryption Policy.
- Ensure that the virtual machine is powered off.
- Verify that you have the required privileges:
    - `Cryptographic operations.Encrypt new`
    - `Cryptographic.operations.Register host`, if host encryption is not enabled.
- The ESXi host must be configured with encryption mode enabled.

> NOTE: The default Administrator system role includes all Cryptographic Operations privileges

Procedure

1. In the vSphere Web Client, right-click the virtual machine that you want to change and select **VM Policies** > **Edit VM Storage Policies**. Set the storage policy for the virtual machine files, represented by VM home, and the storage policy for virtual disks.
2. Select the storage policy.
    a. To encrypt the VM and its hard disks, select an encryption storage policy and click **OK**.
    b. To encrypt the VM but not the virtual disks, toggle on **Configure per disk**, select the encryption storage policy for VM Home and other storage policies for the virtual disks, and click **OK**.

3. If you prefer, you can encrypt the virtual machine, or both virtual machine and disks, from the **Edit Settings** menu in the vSphere Client.
    a. Right-click the virtual machine and select **Edit Settings**.
    b. Select the **VM Options** > **Encryption**. Choose an encryption policy. If you deselect all disks, only the VM home is encrypted.
    c. Click **OK**

NOTE: You cannot encrypt the virtual disk of an unencrypted VM.

Two types of keys are used for virtual machine encryption.

- Data encryption keys (DEKs) are internal keys generated by the ESXi host and used to encrypt virtual machines and disks. DEKs are XTS-AES-256 keys.
- Key encryption key (KEKs) are the keys that vCenter Server requests the KMS. KEKs are AES-256 keys. vCenter Server stores only the ID of each KEK, but not the key itself.

ESXi uses KEKs to encrypt the internal keys and stores the encrypted internal keys on disk. ESXi does not store the KEK on disk. If a host reboots, vCenter Server requests the KEK with the corresponding ID from the KMS and makes it available to ESXi, who then decrypts the internal keys as needed. In addition to VMDK files, most virtual machine files that contain guest data are encrypted, such as the NVRAM, VSWP, and VMSN files. The key that vCenter Server retrieves from the KMS unlocks an encrypted bundle in the VMX file that contains internal keys and other secrets.

NOTE: Encryption keys and configuration are not contained in the VM guest OS.

VM encryption uses vSphere APIs for I/O filtering (VAIO), which is typically called the IOFilter. The IOFilter is an ESXi framework that allows for the interception of VM I/Os in the virtual SCSI emulation (VSCSI) layer, which is just below the VM and above the VMFS file system. It enables VMware and third-party developers to develop services using VM I/O, such as encryption, caching, and replication. It is implemented entirely in user space, which isolates it cleanly from the core architecture and core functionality of the hypervisor. In case of any failure, only the VM in question would be affected. Multiple filters can be enabled for a particular VM or a VMDK, which are typically chained in a manner so that I/Os are processed by each of these filters serially, one after the other, and then finally either passed down to VMFS or completed within one of the filters.

The default Administrator system role includes all Cryptographic Operations privileges. A new default role, the No Cryptography Administrator, supports all Administrator privileges except for the Cryptographic Operations privileges. You can create a custom role that contains granular Cryptographic Operations privileges such as **Cryptographic operations > Encrypt** (allows a user to encrypt a virtual machine or virtual disk) and **Cryptographic operations > Add disk** (allows a user to add a disk to an encrypted virtual machine).

The vSphere Web Client can be used to encrypt and decrypt virtual machines. The vSphere Web Services SDK can also be used to encrypt and decrypt virtual machines as well as to perform a deep recrypt (using a different DEK) or a shallow recrypt (using a different KEK) of a virtual machine. The **crypto-util** command line utility can be used to decrypt core dumps, check for file encryption, and perform management tasks on the ESXi host.

Encrypted vSphere vMotion

Encrypted vSphere vMotion secures confidentiality, integrity, and authenticity of data that is transferred with vSphere vMotion. Starting with vSphere 6.5, vSphere vMotion always uses encryption when migrating encrypted virtual machines. You cannot turn off encrypted vSphere vMotion for encrypted virtual machines. For virtual machines that are not encrypted, you can set encrypted vSphere vMotion to one of the following states. The default is **Opportunistic**.

- **Disabled**: Do not use encrypted vSphere vMotion.
- **Opportunistic**: use encrypted vSphere vMotion if source and target hosts support it.
- **Required**: If the source or destination host does not support encrypted vSphere vMotion, migration with vSphere vMotion is not allowed.

The following rules apply concerning encrypted vMotion.

- vSphere supports encrypted vMotion of unencrypted virtual machines across vCenter Server instances.
- vSphere does not support vMotion of encrypted virtual machines across vCenter Server instances. Because one vCenter instance cannot verify that another vCenter instance is connected to the same Key Management System cluster, the proper encryption keys are not available for successful VM encryption operation.
- For unencrypted virtual machines, all variants of encrypted vSphere vMotion are supported. Shared storage is required for migration across vCenter Server instances.

> NOTE: Only ESXi versions 6.5 and later use encrypted vSphere vMotion.

Available Add-on Security

Often, additional measures that are not provided natively in vSphere are implemented to further secure vSphere environments. Such measures involve additional VMware products, such as vRealize Operations Manager, NSX, and AppDefense.

Compliance using vRealize Operations Manager

You can implement vRealize Operations Manager (vROps) to provide a single pane of glass monitoring solution for your virtual infrastructure, applications, storage, and network devices. vROps provides an open and extensible platform supported by third-party management packs. It monitors performance and availability metrics, performs predictive analysis of the data, and enables pro-active remediation of emerging issues. Additionally, you can use vROps to monitor the vCenter Server instances, hosts, virtual machines, distributed port groups, and distributed switches in your environment to ensure that the settings on your objects meet your defined standards. You can use vROps to define and analyze compliance standards.

vROps includes alerts for multiple versions of the *VMware vSphere Hardening Guide*.. Additionally, hardening guides for regulatory standards are delivered as management packs (PAK files) that you can upload, license, and install. For example, you can install management packs for the following regulatory standards:

- Health Insurance Portability and Accountability Act (HIPAA)
- Payment Card Industry Data Security Standard (PCI DSS) compliance standards
- CIS Security Standards
- Defense Information Systems Agency (DISA) Security Standards
- The Federal Information Security Management Act (FISMA) Security Standards
- International Organization for Standardization Security Standards

vROps collects compliance data from your vSphere objects, generates compliance alerts, and creates reports of the compliance results. To enforce vSphere Hardening Guide compliance and the risk profiles previously described in this chapter, you can use the associated risk profiles in vROps. You can apply the three risk profiles to groups of virtual machines based on whether you must ensure a high, medium, or low level of security in your environment.

NSX Micro-segmentation

Micro-segmentation decreases the level of risk and increases the security posture of the modern data center. Micro-segmentation utilizes the following capabilities.

- Distributed stateful firewalling
- Topology agnostic segmentation
- Centralized policy control
- Granular controls
- Network based isolation

You can use VMware NSX to implement micro-segmentation by implementing a distributed (hypervisor based) firewall to only allow custom-defined network traffic for your virtual machines. Effectively, you can place a firewall on each VM network connection.

The VMware NSX platform provides a centralized firewall service using an Edge Services Gateway (ESG) and a Distributed Firewall (DFW). The NSX ESG provides layer 3 adjacencies from virtual to physical machines. The DFW, which is enabled in the hypervisor kernel as a VIB package, offers near line rate performance, virtualization, identity awareness, automated policy creation, advanced service insertion, and other network security features.

With NSX, isolation can be achieved by leveraging VXLAN technology and virtual networks (i.e., Logical Switches). Isolation can also be achieved with traditional networking methods, such as ACLs, firewall rules, and routing policies. For example, in a brownfield environment, you could choose to keep existing VLAN segmentation to isolate vmkernel traffic and VM zones while using the NSX DFW to implement application segmentation.

With NSX, you can implement virtual machine to virtual machine protection, which is commonly referred to as east-west protection, in more than one manner. For example, you could implement multiple L2 segments with L3 isolation (see figure 9.2) or implement a single L2 segment and use DFW rules (see Figure 9.3).

Fig 9-2 Multiple VXLAN L2 Segments with L3 Isolation

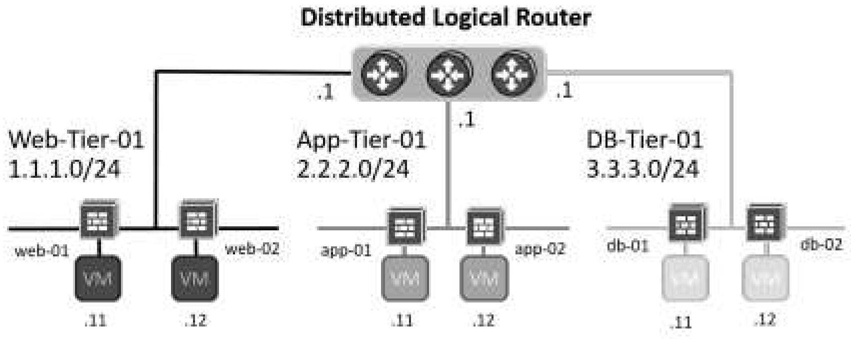


Figure 2.2 Multiple VXLAN L2 segments with L3 isolation

Fig 9-3 Single VXLAN L2 Segments with Distributed Firewall Segentation

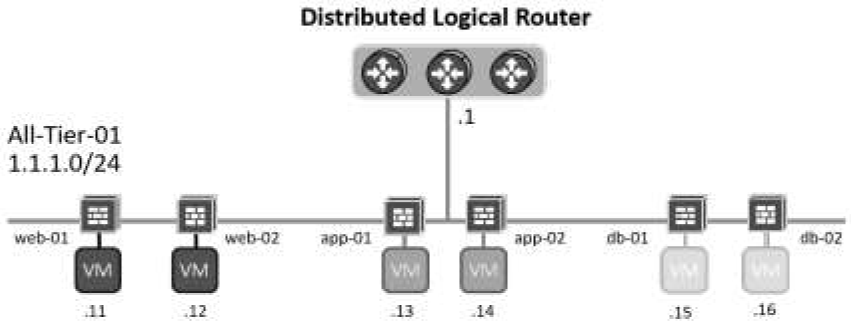
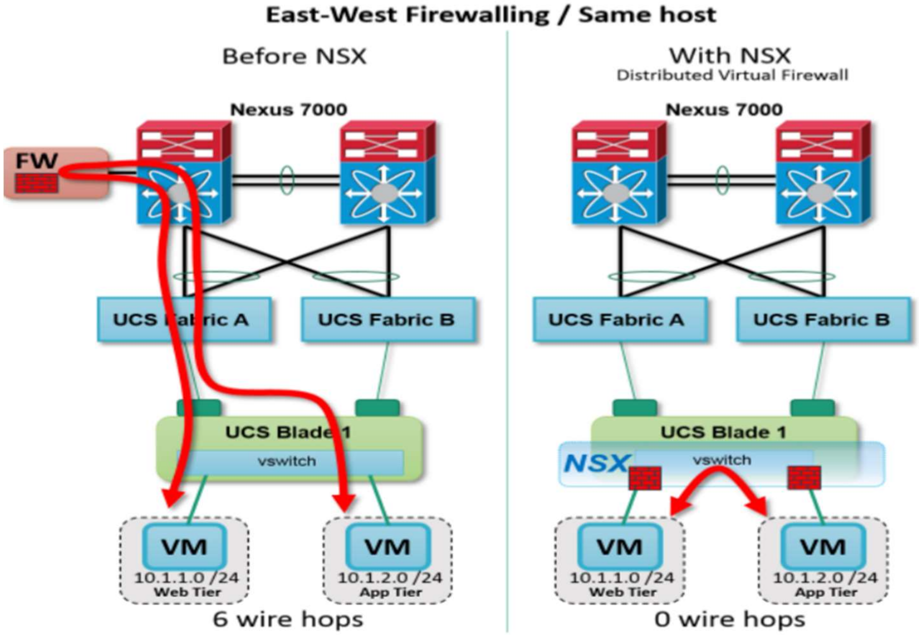


Figure 2.3 Single L2 segment with distributed firewall segmentation

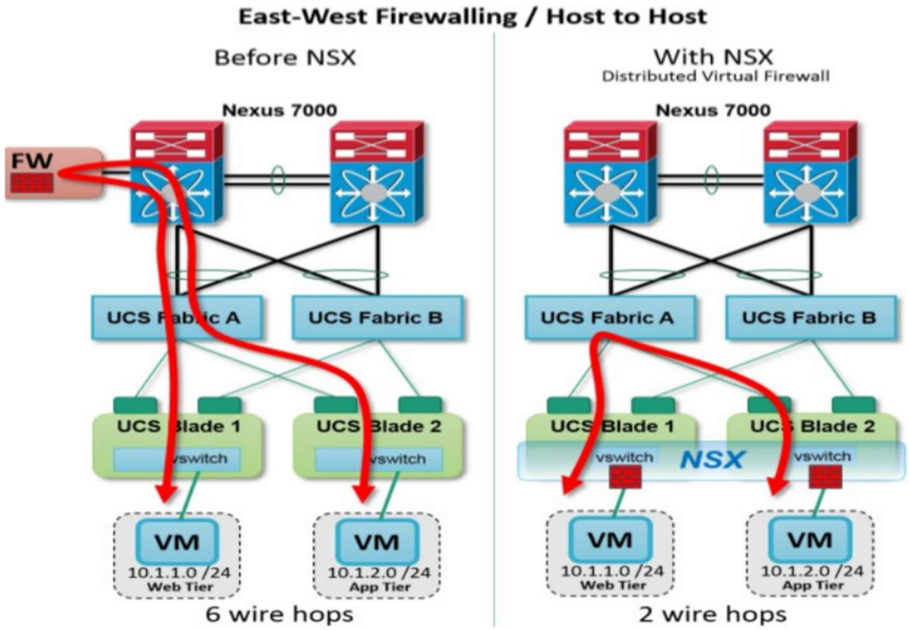
Consider a scenario where without NSX all network traffic must traverse a traditional (physical) firewall to be segmented. In this scenario, even traffic between two virtual machines running on the same ESXi host must traverse the physical network. Six hops may be required. Now consider a similar scenario where the NSX DFW provides the segmentation. Since it is hypervisor based, no hops are required, as shown in figure 9-4. (Figure 9-4 is from the *VMware NSX Micro-Segmentation 101 Presentation)*

Fig 9-4 East West Firewalling – Same Host



Now consider similar scenarios, but where the VMs are on different hosts. With NSX, only two wire hops are required. Without NSX, 6 wire hops are required, as shown in Figure 9-5. (Figure 9-5 is from the *VMware NSX Micro-Segmentation 101 Presentation)*

Fig 9-5 East West Firewalling – Host to Host



AppDefense

You can secure your vSphere environment further by using VMware AppDense. AppDefense is a data center endpoint security product that protects applications running in vSphere. AppDefense understands an application's intended state and behavior, then monitors for changes to that intended state that indicate a threat. When a threat is detected, AppDefense automatically responds based on your policies. It is available as an independent product or as part of vSphere Platinum. You can use AppDefense to define “good behavior” and to trigger automated, custom actions when other behavior is detected.

Key features of AppDense are:

- It understands the intended state of each application and runs inside the hypervisor where it has an authoritative understanding of how data center endpoints are meant to behave. This means it is the first to know when changes are made.
- Being hypervisor based, it runs in an isolated, protected environment, reducing the change that AppDefense itself will be compromised.
- When a threat is detected, it takes the action that you pre-configure, leveraging vSphere and NSX, such as:
  - Block VM network communication
  - Snapshot a VM
  - Suspend or shutdown a VM

Objective 4.1

ESXi Logs

Table 10-1 contains details on the ESXi log files, including the location and purpose of each. You should get familiar with each of these and learn which logs are useful for various troubleshooting scenarios. For example, when troubleshooting virtual machine issues, the only directly useful logs are **vmkernel**, **vmkwarning**, **hostd**, and the specific virtual machine's log files. When troubleshooting issues related to the connection between and ESXi host and the vCenter Server, the **vpxa** log is most useful.

VCP6-DCV Cert Guide: Table 1--4

vCenter Server Logs

Enhanced Logging In vSphere 6.5: , logging is enhanced to support auditing rather than just troubleshooting. Logs coming from vCenter Server via Syslog are now enriched to clearly show "before" and "after" setting changes. The enhanced logging covers VMs and all vSphere changes, such as changes to vCenter Server roles and permissions, datastore-browsing functions (including downloading a VM), and actions such as creating and modifying vCenter Server clusters and hosts

Log Management

Log Levels

The default log level setting on ESXi and vCenter Server systems is **Info**, where errors, warnings, and informational level are logged. You can change the log level to lower levels, such as **Verbose**, which is useful for troubleshooting and debugging, but not recommended for normal use in production environments. You can use the vSphere Web Client to change the logging level for vCenter Server by navigating to and selecting the vCenter Server, select **Manage** > **Settings** > **Edit** and setting the **Logging Settings** to the appropriate level. The available levels are **None**, **Error**, **Warning**, **Info**, **Verbose** and **Trivia**, as shown in Figure 10-15. You can modify use a text editor to modify the logging level on ESXi hosts by modifying the appropriate **config.xml** file. For example, to change the logging level for the hostd service, modify the **/etc/vmware/hostd/config.xml** file and change the value for

You can use the vSphere Web Client to configure the amount of detail that vCenter Server collects in its log files. To do so, you must use a user account with the **Global Setting** privilege. The steps to configure the logging settings are:

Step 1. In the inventory pane, locate and select the vCenter Server.

Step 2. Click on the **Manage** tab.

Step 3. Select **Settings** > **General** > **Edit**

Step 4. Select **Logging Settings**.

Step 5. Select the appropriate logging option as described in Table 11-8 and click **OK**.

VCP6-DCV Cert Guide: Chapter 11 – Configure vCenter Server Logging Options

Viewing Logs

VCP6-DCV Cert Guide: Chapter 10 - Locate and Analyze vCenter Server and ESXi Logs

Exporting Logs

When troubleshooting, a common practice is to create support bundles that you can send to VMware Support. To create support bundles using the vSphere Web Client:

Step 1. Select **Administration** > **System Configuration**

Step 2. In the center pane, select the Objects tab and select the vCenter Server.

Step 3. In the **Actions** menu, select **Export Support Bundles**

Step 4. In the wizard, expand the **VirtualAppliance** bundle and select the specific logs to include.

Step 5. Click the **Export Support Bundle** button.

Step 6. Click the **OK** button.

On Windows based vCenter Servers, you can generate a log bundle by logging into Windows directly as an administrator and using **Start** > **All Programs** > **VMware** > **Generate vCenter Server log bundle**. After creating the log bundle, you should be able to locate it as a ZIP file on the administrator's desktop. This is a handy means to generate a log bundle when the vSphere Web Client is not functioning correctly. Alternatively, use the command prompt and the **cd** command to change the default directory to the vCenter Server installation directory and then to **VMware\vCenter Server\bin** sub directory. From there, issue the **vc-support.bat** command to generate the log bundle.

You can use the vSphere Web Client to view diagnostic data and export system log files for selected ESXi hosts. The required privilege to view diagnostic data is **Read-Only User**. The required privilege to manage diagnostic data is **Global.Licenses**. The procedure to export the system log files is:

Step 1. In the inventory pane, select the appropriate vCenter Server in the **vCenter Servers List**.

Step 2. In the center pane, select the **Monitor > System Logs**.

Step 3. Click the **Export System Logs** button.

Step 4. In the wizard, select the appropriate ESXi hosts form which you wish to export logs.

Step 5. Optionally, select the **Include vCenter Server and vSphere Web Client logs** checkbox. Click **Next**.

Step 6. In the next wizard page, select the specific system log files to collect. Optionally, select **Gather performance data** checkbox.

Step 7. Click the **Generate Log Bundle** button.

Step 8. Click the **Download Log Bundle** button and specify a location to save the bundle. When the download completes, click **Finish**.

Collect Diagnostic Information

You may want to collect diagnostic information from vCenter Server, especially when you want to engage VMware Support. To collect diagnostic data, logon to the vSphere Web Client with a user account that has `Global.Diagnostic` privilege and follow these steps.

Step 1. In the Hosts and Clusters inventory, select a vCenter instance or an ESXi host.

Step 2. Select **Actions** > **Export System Logs…**

Step 3. If prompted, select the appropriate ESXi hosts and click **Next**.

Step 4. In the **System Log** pane, click **Select All**

Step 5. Optionally select **Gather Performance Data** and provide duration and interval parameters.

Step 6. Click **Generate log bundle**

Step 7. Click **Download log bundle**

Step 8. Upload the bundle to the VMware per VMware KB 1008525.

When troubleshooting upgrades, you may be more likely to want to gather diagnostic data from VMware Update Manager. To do so, log into Windows on the VMware Update Manager Server, select **Start** > **All Programs** > **VMware** > **Generate Update Manager log bundle**, and locate the ZIP file. The file is named `vum-support-xxxxx.zip`, where xxxxx represents the user, date and time associated with the bundle generation.

VCP6-DCV Cert Guide: Chapter 11 - Create/Locate/Analyze VMware Log Bundles

Syslog

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.esxi.upgrade.doc/GUID-9F67DB52-F469-451F-B6C8-DAE8D95976E7.html

vRealize Log Insight

https://docs.vmware.com/en/vRealize-Log-Insight/4.8/com.vmware.log-insight.administration.doc/GUID-09F7B992-D7F4-40B6-8F16-4B5A6E9BEC67.html

Objective 4.2

vSphere Inventory Overview

In vSphere, the inventory is a collection of virtual and physical objects that you can manage. You can configure, set permissions, monitor tasks and events, and set alarms on most of these objects. You can organize many of these objects by placing them into folders, making them easier to manage.

All inventory objects, except for hosts, can be renamed to represent their purposes. For example, they can be named after company departments, locations, or functions.

NOTE: In vSphere 6.7, inventory object names cannot exceed 214 bytes (UTF-8 encoded).

Data Centers

A data center is a container object in the vSphere inventory that is an aggregation of all the different types of objects used to work in virtual infrastructure. The first object that you must create in a vSphere inventory is a data center (with the exception of a folder to contain data centers) You cannot add any ESXi hosts, virtual machines, or other objects in the inventory until you create a data center.

Data centers are often used to contain all the objects in a physical data center. For example, if you use a single vCenter Server to manage vSphere assets in San Francisco and Chicago, you may wish to use corresponding virtual data centers to organize each city's assets. You could create data center objects named San Francisco and Chicago and place each ESXi host, virtual machine, and other objects in the appropriate data center.

Within each data center, there are four separate hierarchies.

- Virtual machines (and templates)
- Hosts (and clusters)
- Networks
- Datastores

The data center the namespace for networks and datastores. The names for these objects must be unique within a data center. You cannot use identical datastore names within the same name data center, but you can use identical datastore names within in two different data centers. Virtual machines, templates, and clusters need not to have unique names within the data center, but have unique names within their folder.

Folders

Folders are container objects in the vSphere inventory that allow you to group objects of a single type. A folder can contain data centers, clusters, datastores, networks, virtual machines, templates, or hosts. For example, one folder can contain hosts and a folder containing hosts, but it cannot contain hosts and a folder containing virtual machines.

You can create data center folders directly under the root vCenter Server and use them to organize your data centers. Within each data center is one hierarchy of folders for virtual machines and templates, one for hosts and clusters, one for datastores, and one for networks.

The only setting that is available for you to set on a folder is its name. Additionally, you can assign permissions and alarms.

Clusters

A cluster is a set of ESXi hosts that are intended to work together as a unit. When you add a host to a cluster, the host's resources become part of the cluster's resources. vCenter Server manages the resources of all hosts in a cluster as one unit. In addition to creating a cluster, assigning a name, and adding ESXi objects, you should enable and configure features on a cluster, such as VMware EVC, vSphere DRS, and vSphere HA.

If you enable VMware EVC on a cluster, you can ensure that migrations with vMotion do not fail because of CPU compatibility errors. If you enable vSphere DRS on a cluster, you can allow automatic resource balancing using the pooled host resources in the cluster. If you enable vSphere HA on a cluster, you can allow rapid virtual machine recovery from host hardware failures using the cluster's available host resource capacity.

Cluster features are covered in details elsewhere in this book.

Resource pools

Resource pools are container objects in the vSphere inventory that are used to compartmentalize the CPU and memory resources of a host or cluster. Virtual machines run in, and draw their resources from, resource pools. You can create multiple resource pools as direct children of a standalone host or cluster.

You can use resource pools much like a folder to organize VMs. You can delegate control over each resource pool to specific individuals and groups. You can monitor resources and set alarms on resource pools. If you need a container for strictly for organization and permission purposes, consider using a folder. If you also need resource management, then consider using a resource pool.

If DRS is enabled, you can use the vSphere Client to create resource pools in the cluster and assign resource settings, such as reservations and limits. Otherwise, you can create resource pools directly on specific ESXi hosts.

You can various options for monitoring status and managing resources in these resource pools. For example, you can configure resource reservations, limits, and shares on each resource pool.

Hosts

Hosts are objects in the vSphere inventory that represent your actual ESXi servers. After installing an ESXi host, you can choose to add it to the vSphere inventory, which requires you to provide credentials that is assigned the `Administrator` role directly on the host.

Networks

Networks are objects in the vSphere inventory that are used to connect a set of virtual adapters. Each ESXi host may have multiple VMkernel virtual network adpapters. Each virtual machine may have multiple virtual network adapters. Each virtual network adapter may be connected to a port group (on a standard virtual switch) or a distributed port group (on a vSphere distributed switch). All virtual machines that connect to the same port group belong to the same network in the virtual environment, even if they are on different physical servers. You can manage networks by monitoring, setting permissions, and setting alarms on port groups and distributed port groups.

The steps to create, configure, and manage networks are covered in other chapters of this book.

Datastores

Datastores are objects in the vSphere inventory that represent physical storage resources in the data center. A datastore is the storage location for virtual machine files. The physical storage resources can come from the local SCSI disk of the ESXi host, the Fibre Channel SAN disk arrays, the iSCSI SAN disk arrays, or Network Attached Storage (NAS) arrays. VMFS datastores can be backed by local SCSI, Fibre Channel, or iSCSI. NFS data stores can be backed by NAS. vSAN datastores can be built in VSAN clusters.

The steps to create, configure, and manage datastores are covered in other chapters of this book.

Virtual Machines

Virutal machines are represented in the vSphere inventory in a manner that reflects the current inventory view. For example, in the Hosts and Clusters view, each virtual machine is descendent of the ESXi host on which it runs. In the Networks view, each virtual machine is descendent of the network to which it connect.

Templates

Templates are objects in the vSphere inventory that effectively are non-executable virtual machines. A template is a master copy of a virtual machine that can be used to create and provision new virtual machines. Templates can have a guest operating system and application software installed. They are often customized during deployment to ensure that each new virtual machine has a unique name and network settings.

You can convert a virtual machine to a template and vice versa. But, the main use case for templates is for rapid deployment of new, similar vitual machines from a single template. In this case, you are effectively cloning the template again and again, allowing the template to remain unchanged and ready for future use. To update a template, such as to install the most recent guest OS updates, you can temporarily convert the template to a virtual machine, apply the updates, and convert back to template.

vApps

A vApp is a container object in vSphere that provides a format for packaging and managing applications. Typically, a vApp is a virtual machines that runs a single application and allows you to manage the application as a single unit. For example, you can specify a unique boot order for the VMs in a vApp to facilitate the graceful startup of an application spanning multiple virtual machines.

Objective 4.3

VCP6-DCV Cert Guide – Chapter 16 – Objective 10.2

Content Library Improvements in vSphere 6.5:

In vSphere 6.5, you can now mount an ISO directly from the Content Library, apply a guest OS customization specification during VM deployment, and update existing templates. The Content Library performance is improved. The new Optimized HTTP Sync option stores content a compressed format, which reduces the synchronization time. The Content Library leverages new features in vCenter Server 6.5, including VCSA HA and backup/restore.

Objective 4.4

To begin your vSphere deployment, you should install and configure at least one ESXi host using the information in this section. Optionally, you can apply the information here to install and configure additional ESXi hosts. In many cases, administrators choose to deploy the first ESXi host, then deploy vCenter Server, and use vCenter Server along with other tools, such as host profiles, to facilitate the deployment and configuration of the remaining ESXi hosts.

You have several choices for installing ESXi, such as using the interactive wizard, using scripts, and using Auto Deploy. These choices are covered in this section. Using host profiles to configure ESXi hosts after installation is covered in a separate section later in this chapter.

Install ESXi Interactively

You can use the following procedure to install ESXi interactively, which is very useful in small environments with fewer than five ESXi hosts.

Preparation:

- Gather and record the information that will be required during the installation as shown in table 11-2
- Verify that the server hardware clock is set to UTC. This setting is in the system BIOS.
- Download the ESXi Installer ISO and prepare the hardware system to boot from it.

Procedure

1. Start the machine so that it boots from the ESXi Installer
2. On the Select a Disk page, select the drive on which to install ESXi and press Enter.
3. Select the keyboard type for the host.
4. Enter a password to be used by the root account.
5. Press Enter to start the installation
6. When prompted, remove the bootable media and press Enter to reboot the host.

Table 11-2 Required Information for ESXi Installation

Information	Required or Optional	Details
Keyboard layout	Required	Default: US English
VLAN ID	Optional	Range: 0-4094 Default: None
IP address	Optional	Default: DHCP
Subnet mask	Optional	Default: Based on the configured IP address
Gateway	Optional	Default: Based on the configured IP address and subnet mask
Primary DNS	Optional	Default: Based on the configured IP address and subnet mask
Secondary DNS	Optional	Default: None
Host name	Required for static IP settings	Default: None

Install location	Required	Must be at least 5 GB if you install on a single disk. Default: None
Migrate existing ESXi settings. Preserve VMFS datastore.	Required if you are installing ESXi on a drive with an existing ESXi installation.	Default: None
Root password	Required	Must contain at least 8 to 40 characters in addition to other requirements. Default: None

The default behavior is to configure the ESXi management network using DHCP. You can override the default behavior and use static IP settings for the management network after the installation is completed. If your host is not yet assigned an IP address or if you wish to change it, you can use the following procedure to select the appropriate network adapter, configure the VLAN, and configure the IP configuration for the host's management network interface.

Procedure

1. Logon to the Direct Console User Interface (DCUI), which appears on the host's monitor
2. If needed, use the DCUI to change the network adapter used for management
   a. Select **Configure Management Network** and press Enter.
   b. Select **Network Adapters** and press Enter.
   c. Select a network adapter and press Enter.
3. If needed, use the DCUI to change the VLAN used for management
   a. Select **Configure Management Network** and press Enter.
   b. Select **VLAN** and press Enter.
   c. Enter the appropriate VLAN ID number for your network connection.
4. If needed, use the DCUI to change the IP configuration used for management
   a. Select **Configure Management Network** and press Enter.
   b. Select **IP Configuration** and press Enter.
   c. Select **Set static IP address and network configuration.**
   d. Enter the IP address, subnet mask, and default gateway and press Enter.

You can use the DCUI to configure DNS following this procedure.

1. Select **Configure Management Network** and press Enter.
2. Select **DNS Configuration** and press Enter.
3. Select **Use the following DNS server addresses and hostname**.

4. Enter the primary server, an alternative server (optional), and the host name.

After ESXi is installed and the management network is configured, you can manage the host and make other configuration changes using the vSphere Host Client.

Scripted ESXi Installation

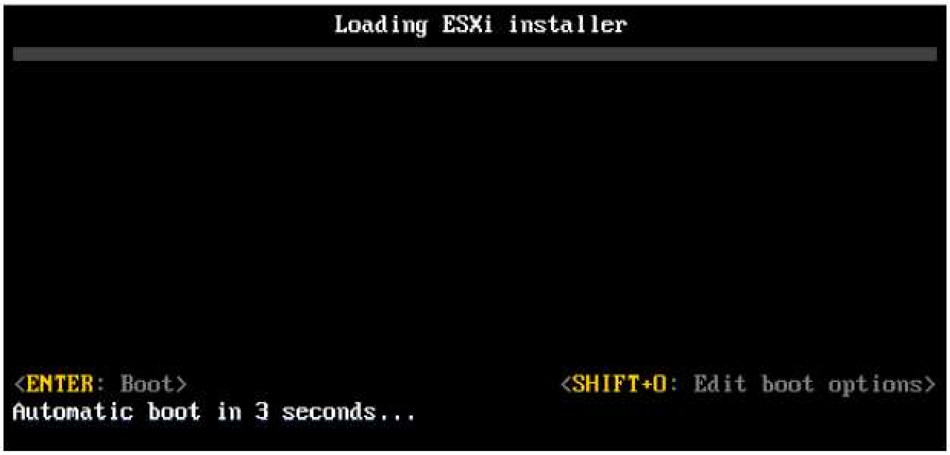
You can quickly deploy ESXi hosts by using scripted, unattended installations or upgrades. Scripted installations or upgrades provide an efficient way to deploy multiple hosts.

The installation script contains the installation settings for ESXi. You can apply the script to all hosts that you want to have a similar configuration. You must use the supported commands in the script. You modify the script to accommodate settings that are unique for each host. The installation script can reside in one of the following locations:

- FTP server
- HTTP/HTTPS server
- NFS server
- USB flash drive
- CD-ROM drive

To start the installation script, you can enter boot options at the ESXi installer boot command line. At boot time you can press Shift+O in the boot loader (see Figure 1-1) to specify boot options and access the kickstart file.. For a PXE boot installation, you can pass options through the `kernelopts` line of the boot.cfg file. To specify the location of the installation script, set the `ks=filepath` option, where filepath indicates the location of your Kickstart file.. If `ks=filepath` is omitted, the text installer is run.

Figure 1-1 ESXi Installer



```
Loading ESXi installer

<ENTER: Boot>                          <SHIFT+O: Edit boot options>
Automatic boot in 3 seconds...
```

Procedure

1. Start the host.
2. When the ESXi installer window appears, press Shift+O to edit boot options.
3. At the runweasel command prompt, enter `ks=` along with the path to the installation script and the command line options. For example, you could enter the following options to boot the host from a script named ks-script-01 residing on the server 192.168.1010.10 and to set the host's IP address to 192.168.100.101

```
ks=http://192.168.100.10/kickstart/ks-script-01.cfg
nameserver=192.168.1.100 ip=192.168.100.101 netmask=255.255.255.0
gateway=192.168.100.101
```

To successfully perform a scripted installation, you may need to enter boot options to access the script file. Table 11-3 shows some of these options.

Table 11-3 Boot Options for ESXi Scripted Installation

Boot Option	**Description**
`BOOTIF=`*`hwtype-MAC address`*	Similar to the `netdevice` option, except in the PXELINUX format
`gateway=`*`ip address`*	Gateway used for downloading the installation script.
`ip=`*`ip address`*	IP address used for downloading the installation script.
`ks=cdrom:/`*`path`*	Performs a scripted installation with the script at *path*, which resides on the CD in the CD-ROM drive.
`ks=file://`*`path`*	Performs a scripted installation with the script at *path*.

ks=*protocol*://*serverpath*	Performs a scripted installation with a script located on the network at the given URL. The *protocol* can be http, https, ftp, or nfs, as in this example, ks=nfs://*host*/*porturl-path*.
ks=usb	Performs a scripted installation, accessing the script from an attached USB drive
ks=usb:/*path*	Performs a scripted installation with the script file at the specified path, which resides on USB.

ksdevice=*device*
or
netdevice=*device*
	Tries to use a network adapter *device* when looking for an installation script and installation media. Specify as a MAC address, for example, 00:50:56:C0:00:01. This location can also be a vmnic## name.

nameserver=*ip address*	Specifies a domain name server to be used for downloading the installation script and installation media.
netmask=*subnet mask*	Subnet mask used for downloading the installation script.
vlanid=*vlanid*	VLAN used used for downloading the installation script.

The ESXi Installer includes a default installation script that can be used to install ESXi to the first detected disk. The default ks.cfg installation script is in the initial RAM disk at /etc/vmware/weasel/ks.cfg. You can specify the location of the default ks.cfg file with the ks=file://etc/vmware/weasel/ks.cfg boot option. When you install ESXi using the ks.cfg script, the default root password is myp@ssw0rd. You cannot modify the default script on the installation media. After the installation, you can use the vSphere Web Client to log in to the vCenter Server that manages the ESXi host and modify the default settings.

Here are the contents of the default script

```
#

# Sample scripted installation file

#

# Accept the VMware End User License Agreement

vmaccepteula

# Set the root password for the DCUI and Tech Support Mode
```

```
rootpw myp@ssw0rd

# 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',
```

In the default script, you can see it sets the root password to `myp@ssw0rd`, installs on the first disk, overwites any existing VMFS datastore, and set the network interface to use use DHCP. When creating your own script, you can specify many options, a few are shown in Table 11-4

Table 11-4 Sample Options for ESXi Installation Script.

Command	**Options**	**Descript**
clearpart	--ignoredrives=	Removes partitions on all drives except those specified
	--overwritevmfs	Allows overwriting of VMFS partitions on the specified drives.
dryrun	N/A	Parses and checks the installation script, but does not perform the installation.
install	--disk=	Specifies the disk to partition. Acceptable values can use various forms, like these examples. **Path**: --disk=/vmfs/devices/disks/mpx.vmhba1:C0:T0:L0 **MPX name**: --disk=mpx.vmhba1:C0:T0:L0 **VML name**: --disk=vml.000000034211234 **vmkLUN UID**: --disk=vmkLUN_UID

	--ignoressd	Excludes solid-state disks from eligibility for partitioning
	--overwritevsan	You must use the --overwritevsan option when you install ESXi on a disk, either SSD or HDD (magnetic), that is already in a vSAN disk group.
network	--bootproto=	Specifies if the IP address should be set statically of via DHCP.
	--ip=	Sets an IP address for the machine to be installed, in the form xxx.xxx.xxx.xxx. Required with the `bootproto=static` option and ignored otherwise
	--nameserver	Designates the primary name server as an IP address. Used with the `bootproto=static` option. You can omit this option if you do not intend to use DNS.

Auto Deploy

VCP6-DCV Cert Guide: Chapter 14 - Deploy ESXi Hosts Using Auto Deplo

Objective 4.5

Create and configure Standard Virtual Switches

Standard Virtual Switches can be created in both the vSphere Client and vSphere Web Client. They can also be created via ESXi shell, vCLI, or PowerShell scripts. When created in the graphical interfaces of the vSphere Web Client and vSphere Clients, virtual switches adhere to a vSwitch# naming convention. However, when creating virtual switches via one of the command line interfaces, virtual switches can be created with any name desired.

To create a virtual switch using the vSphere Client, the process is identical to creating one in the vSphere Web Client:

Step 1. While in the Hosts and Clusters inventory view, select an ESXi host from the Navigator pane.

Step 2. Select the **Configure** tab

Step 3. Select **Networking**

Step 4. Select **Virtual switches**

Step 5. Click the **Add Networking** button

Step 6. Select the radio button for **VMkernel Network Adapter** (If you want to create a new VMkernel port), or the radio button for **Virtual Machine Port Group for a Standard Switch** (If you want to create a virtual machine port group for use by virtual machines) and click Next.

Step 7. Select the option **New standard switch** and click **Next**.

Step 8. In order to connect a physical network adapter port to the virtual switch and enable physical network connectivity, select the Plus button under Assigned adapters.

Step 9. Choose the appropriate network adapter port, identified by a vmnic# designation, and select **OK**.

Step 10. Verify the adapter selected is listed under **Active Adapters** and click **Next**.

Step 11. Enter the VM Port Group name, and, if desired, enter a VLAN ID and click Next.

Step 12. Verify the virtual switch configuration and click Finish.

Create and configure Standard Virtual Switch Port Groups

Creating Standard Virtual Switch port groups hasn't really changed in the different VMware interfaces over the years. If an administrator wanted to create a new VM port group (or VMkernel port) on a new vSwitch, the process is outlined in the prior section "Create and configure Standard Virtual Switches." If an administrator wanted to create a new VMkernel port or port group on an existing vSwitch, however, you would just substitute **Select an existing network** as opposed to **New standard switch** from step 7.

Port groups have various additional configurations which can be accessed from the same place in the vSphere Client. If you select the elipses next to the port group name, you will see 3 options: **View Settings**, **Edit**, and **Remove**. Select **Edit** to enter the configuration options for the vSwitch.

The first screen of the port group Edit Settings window is Properties. Here, you can change the port group name and define a VLAN ID. The default VLAN ID is None (0), which means no traffic in this port group will be tagged with 802.1q VLAN tags. The valid VLAN ID this window will select are 1-4094. However, it will also accept 4095. VLAN 4095 is not a typical VLAN, however, as it is a trunk. This means that all VLANs defined on this switch would be able to communicate to virtual machines attached to a port group tagged for VLAN 4095, and vice-versa. Additionally, if the VLAN 4095 port group was in Promiscuous mode, then a VM attached to this port group could capture packets for any VM on the virtual switch regardless of VLAN.

The next section of the port group Edit Settings window is Security. Here, we can define Promiscuous mode, MAC address changes, and Forged transmits. These settings are defined at the vSwitch level, and whatever defined there is the default for all port groups on the vSwitch. However, administrators can override the vSwitch setting at the port group level. This more specific setting (the port group vs the vSwitch) will take precedence. Let's look at each one of these settings:

- Promiscuous Mode: by default a VM cannot receive packets not destined for it. As a result, if a VM were to run a packet capture utility such as tcpdump or Wireshark, it wouldn't be able to see any traffic except for its own. But, if we needed to be able to analyze traffic usage with a tool like this, we would need to have Promiscuous mode enabled on the port group.

- MAC address changes: this setting allows a VM, or more specifically an OS, to be able to change its own MAC address
- Forged transmits: this setting would allow a VM to be able to send a packet that appears to come from a different MAC address (spoofing). This setting and MAC address changes have two main uses:
    - To make it relatively easy to configure MS Failover Clustering, and MS Network (MSNLB).
    - If an administrator performed a Physical to Virtual conversion (P2V), and the application in the OS is licensed by MAC address.

The next setting in the Edit Settings window is Traffic shaping. Traffic shaping is disabled by default. If enabled, traffic shaping can be performed on outbound traffic only on standard virtual switches, and can be configured for either or both on distributed virtual switches. Administrators can define an average bandwidth in kb/s, peak bandwidth in kb/s, and burst size measure in KB. This would allow a workload to enjoy whatever is defined as the average bandwidth, but would allow to burst, if needed, and if the additional bandwidth is available, to the burst size. Once the burst size has been reached, whatever using this port group would be forced back down to whatever is defined as the average. In order for a workload to be able to burst, it also has to earn a "bonus," which means, there has to be a period of time the VM or groups of VMs have used less than what is defined as the average. Any amount of bandwidth below the average gets counted towards this bonus.

The last setting on the Edit Settings window is Teaming and failover. Here, administrators can configure the load balancing mechanism, change network failure detection, enable or disable the notify switches option, enable or disable failback, and define a custom failover order.

vSphere Standard Switches have 3 load balancing mechanisms:

- Route based on originating virtual port. This is the default setting. When a virtual machine (or VMkernel port) is attached to a virtual switch, it gets an associated virtual switch port ID. This port ID doesn't change regardless of power state. When a VM sends a packet, based on the virtual machine's port ID, the vSwitch forwards that packet out one physical uplink. Because that VM's port ID doesn't change, all of the additional packets that are sent to/from that virtual machine will use the same physical uplink. The virtual switch will then forward the next port ID's traffic out the next physical uplink on the team, and continue in this manner, rotating through all of the uplinks.
- Route based on source MAC hash. This mechanism generates a hash based on the source, or virtual machine's MAC address. Based on this hash, the vSwitch then sends that virtual machine' traffic out one physical uplink. Because the VM's MAC addresses shouldn't be changing, the hash doesn't get regenerated, and future packets from the same virtual machine use the same physical uplink. Functionally, this works the same was as route based on source port ID, but adds overhead with the calculation of the hash values.
- Route based on source and destination IP hash. This mechanism builds a hash value based on the source *and* destination IP address. When a virtual machine sends a packet, based on source and destination address, a hash gets generated. Based on this hash value, the vSwitch forwards that packet out one physical uplink. However, when that same virtual machine sends another packet, the source IP address probably won't change, but the destination address certainly could. In this case, the hash would be re-generated, and the packet would be forwarded out the next physical uplink in the team.

All three mechanisms rotate through all of the physical uplinks, however, route based on source port or source MAC only rotate based on identifiers which don't change. With route based on source and destination hash, the hash value changes, resulting in more frequent rotation through the physical uplinks, and achieving more balanced resource distribution. When looking at these three mechanisms, it can be helpful to think of them as load distributors versus load balancers.

The next option, network failure detection, is set to link status only by default. So, the only way a vSwitch will stop forwarding traffic out a physical uplink is if link is lost. This is useful when there is a cable, switch, or card failure, but if a physical switch port stops forwarding traffic, but the link light is still on, a failover will not occur, and anything on that switch port would drop off the network.

The second option is link status plus beaconing. Beaconing, or beacon probing, introduces a 62-byte packet sent every second per physical uplink on the team. Each uplink on the team is sending and receiving these packets. After 10 missed packets, the vSwitch will identity the uplink has failed. As a result, if a physical switch port stops forwarding traffic, the vSwitch will then fail over to other physical uplinks on the team. VMware recommends at least 3 physical uplinks on the team to use beaconing, to accurately detect a port failure.

The next setting is Notify switches. This determines when the virtual switches will notify physical switches with an ARP update. If there is a failure like the one described in beacon probing occurs, the virtual switch will notify the physical switch of this failure. Also, whenever virtual machines are attached to a virtual switch, their MAC addresses are forwarded to the physical switch via ARP. There is only one particular reason why someone would not want a VM's MAC address to be forwarded to the physical network: when using MSNLB in unicast mode.

The next setting is failback. This is set to yes by default, so if a failover occurs, but then the original uplink comes back up, whatever was using that uplink returns to using that uplink. It could be helpful to turn this to no, if flapping occurs (failover, failback, failover, etc). This would then stop the flapping, allowing an administrator time to investigate and resolve the issue, then set failback to yes.

The last setting is failover order. By default, this is set to Active/Standby when more than one physical uplink is attached to the vSwitch. Changing to Active/Active will allow one of the load balancing mechanisms to be used. Adapters can be moved from active, to standby, to unused. This can be configured per port group and/or at the virtual switch as a whole.

Create and configure Distributed Virtual Switches

Now that we have taken a look at the features that vDS make available, we can look at the steps involved in creating a vDS. When creating a vSphere distributed switch, it is important to determine whether you are working with a mixed environment of multiple different versions of vSphere. When building a vDS, you can define the compatibility of the switch. It is also important to be aware of the licensing restrictions: Distributed virtual switches are available only with Enterprise Plus vSphere licensing.

To create a vDS, you need to be in the networking inventory view of the vSphere Client or Web Client

In the networking inventory view, right-click the datacenter object and select Distributed Switch > **New Distributed Switch**

In the New Distributed Switch wizard that appears, enter the name of the new vDS and then select the version. The version selection is critical if you are working with an environment of mixed ESXi versions. If all of the ESXi servers are at version 6, then you can safely select Distributed Switch 6.0.0. If, however, some of the ESXi servers are older versions, then in order to have vDS functionality, you must choose only the version of vDS that matches the lowest version of ESXi that is in the environment. If your environment has some ESXi 5.1, 5.5, and 6.0 servers, for example, you need to choose Distributed Switch: 5.1.0.

The next screen of the wizard is the Edit Settings screen. In this screen, you define the number of uplinks, enable/disable Network I/O Control, and create a VM port group. The number of uplinks defined on this screen determines the maximum number of physical uplinks each ESXi server can contribute to the vDS. The default value here is 4, which allows every ESXi server (a maximum of 1000 per vDS) to contribute up to 4 physical uplinks to the vDS. The minimum value here is 1, and the maximum value is 32. By defining a maximum, you restrict future growth by only allowing ESXi servers to be able to contribute up to the maximum number of uplinks defined here. This value can, however, be modified after the creation of the distributed switch. This would allow for delegation. For example, a vSphere networking admin may be responsible for building and maintaining/modifying virtual switches but may be unable to modify ESXi servers or their physical uplinks. In this case, the network admin might need to restrict the number of uplinks. If at some point in the future it is determined that each ESXi server needs to contribute more physical uplinks to the switch, the vSphere network admin can then adjust the maximum number of uplinks per host.

Network I/O Control allows prioritization of network traffic if there is contention on the network. This setting is enabled by default, and administrators can create network resource pools. There are several network resource pools configured by default: one for each different type of VMkernel network traffic, and one for all of the virtual machines. Administrators can create additional network resource pools for varying levels of priorities for virtual machines as needed. Network resource pools prioritize network traffic by utilizing a share mechanism (the same mechanism used for prioritization in the event of CPU, memory, or storage contention). They can also be used to define limits, if needed.

The last setting, Create a Default Port Group, creates a virtual machine port group on the new vDS during creation. By default, this option is selected and the port group name is DPortGroup; you can modify the name either in this wizard or after the vDS has been created.

At the end of the wizard is a completion screen that indicates what is going to be created and also gives tips on next steps. These suggestions include creating new port groups and adding ESXi servers to the vDS.

Once the vDS has been created, it shows up in the networking inventory view, along with two port groups: one for virtual machines (the one that was created by default) and one for the physical uplinks, with a name in the format <vDS name>-DVUplinks-<number>.

To delete a vDS, you simply right-click the vDS and choose Delete. However, if it has been pushed to ESXi servers first and has virtual machines residing on it, you have to migrate them off first and remove the vDS from the ESXi servers.

Create and configure Distributed Virtual Switch Port Groups

Whether a virtual machine port group was created on a vDS during creation or not, you can create or modify additional ones after the fact. To create a dvPort group (distributed virtual port group), you need to be in the networking inventory view of the vSphere Client or vSphere Web Client.

Right-click the vDS and select Distributed Port Group > New Distributed Port Group.

The first screen of the New Distributed Port Group wizard that appears asks you for a name. The default dvPort group name is DPortGroup#. If there is already one dvPort Group on the virtual switch, the vSphere Client starts appending numbers at the end of new dvPort group names, such as DPortGroup1, DPortGroup2, and so on. In this example, you need to create a dvPort group with the name Production.

The next screen of the wizard allows you to determine how virtual machines will be bound to virtual switch ports, how you want the virtual switch ports to be allocated, what default number of virtual switch ports you want, whether you want to tie the port group to a network resource pool, whether you want to define a VLAN for the port group, and whether you want to modify the default security, traffic shaping, or teaming and failover policies.

The first option on this screen, Port Binding, determines how virtual machines are bound or attached to virtual switch ports. It has three possible settings:

- Static Binding: With Static Binding (which is the default), when a virtual machine's network adapter is attached to a dvPort group, that connection is static[md]meaning the virtual machine is always attached to the virtual switch port, regardless of power state. The VM never releases or is disconnected from the virtual switch port, so the statistics and policies migrate with the virtual machine during a migration.
- Dynamic Binding: The next option, Dynamic Binding, means that virtual machine network adapters are only connected to the dvPort when the virtual machine is powered on. Powering off the virtual machine disconnects it from a virtual switch port. As a result, you could potentially have more virtual machines than virtual switch ports as long as there aren't any more virtual machines powered on than there are virtual switch ports. It is important to realize that dynamic port binding has been deprecated as of vSphere 5.0. So although it is still available as an option, using it is no longer recommended.
- Ephemeral: The third option, Ephemeral, means there is no binding. With this setting, virtual switch ports are created and deleted on demand. In the past, the Ephemeral setting seemed like the easiest way to go because it required the least administrative effort to address an ever growing environment. However, as of vSphere 5.1, static port binding is also "elastic" by default.

The first option on this screen, Port Allocation, is set to Elastic by default. This essentially has the same effect as ephemeral port binding: Ports are created and removed on demand. This means a virtual switch with only eight ports configured by default will automatically expand and add more ports as you try to attach more virtual machines to it. If you remove virtual machines from the vDS, the ports are automatically removed.

If you set Port Allocation to Static, then once the ports defined on the dvPort group are consumed by virtual machines (powered on or not), there will not be any additional ports created or available for new virtual machines to use, which means you can effectively run out of ports. So you can see why the default is eight ports, with elastic static port binding: to allow the lowest amount of overhead by not configuring a virtual switch with a lot of ports unused up front and allowing an automatic expansion of switch ports as needed, as well as the ability of virtual machines to always maintain which dvPort they are attached to, regardless of power state or which ESXi server the VM resides on. The default settings work well, without requiring vSphere administrators to have any additional knowledge or training on how port binding should be configured.

Another thing to be aware of is the maximum number of ports that can be configured. The maximum number of static or dynamic ports that can be defined per vDS is 10,000, while each ESXi server can manage only up to 4096 total switch ports (for standard as well as distributed virtual switches). The maximum number of ephemeral ports per vDS is only 1016, which may be another reason the default static port binding seems to be a better option.

Note: If a vCenter Server is a virtual machine attached to a vDS, it is recommended that the port group the vCenter Server is attached to be configured for ephemeral port binding. With static or dynamic port binding, vCenter is responsible for assigning virtual switch ports, and if the vCenter Server needs to be replaced with a new VM, this wouldn't be possible. With ephemeral port binding, the new VM would be able to attach to the vDS, however.

The next option is Network Resource Pool. There are no network resource pools defined for virtual machines other than the default (which groups all virtual machine traffic together in the event of contention). You can modify this only when a different network resource pool has been defined.

VLAN Type is, by default, set to None. You can define VLANs for logical segmentation on dvPort groups, as you can with standard virtual switch port groups, but there is more functionality available for distributed virtual switches, as we explore later in this chapter, in the section "Configure VLAN/PVLAN Settings for VMs Given Communication Requirements."

If you want to customize the policies (security, traffic shaping, and teaming and failover policies), you can select the last check box in the Configure Settings screen. Otherwise, you can always define these settings after the new port group has been created.

The last screen of the New Distributed Port Group wizard allows you to check all the selections that have been made for the new port group.

Once a dvPort group has been created, you can remove it, as long as you move all the virtual machines off the dvPort group first. An interesting thing to be aware of is that if you try to remove a dvPort group when there are virtual machines attached to it, even if those virtual machines aren't powered on, you receive an error message that the selected resource is in use. The easiest way to see what VMs are attached to the dvPort group is to select the port group, select the Related Objects tab, and select Virtual Machines. You get a list of all the VMs attached to the dvPort group, and you can then move them to another dvPort group or standard vSwitch port group. Once there are no virtual machines attached to the dvPort group, you right-click the dvPort group under the vDS in the Navigator pane and select Delete

Create and configure NSX Logical Switches

Logical switches are the virtual layer 2 domains created in NSX-T. Logical switches are part of one of two transport zones: Overlay transport zones, and VXLAN transport zones. Logical switches are visible as dPortGroups on vSphere Distributed Switches.

In order to create and use Logical Switches, NSX Controllers and Managers have to be deployed first.

To create a new NSX Logical Switch:

1. Log into the vSphere Web Client
2. Navigate to Home -> Networking and Security
3. Select Logical Switches from the Navigator pane.
4. Select the appropriate NSX Manager from the drop down menu at the top of the screen.
5. Click the green plus "Add Logical Switch" button.
6. Provide the following details:
   a. Name
   b. Description
   c. Transport Zone
   d. Replication Mode
      i. Multicase
      ii. Unicast
      iii. Hybrid
   e. Select whether you want to enable IP Discovery and/or MAC Learning.

Objective 4.6

You can install vCenter Server and Platform Services Controller (PSC) on Windows systems or you can deploy the vCenter Server Appliance and the PSC Appliance.

vCenter Server Database

For vCenter Server on Windows, you can either use the bundled PostgreSQL database and install it together with vCenter Server or you can set up an external database prior to installing vCenter Server. vCenter Server for Windows supports Oracle and Microsoft SQL Server as external databases. You could use the following steps to configure a Microsoft SQL Server database for on the same or separate Windows Server as the vCenter Server.

1. Log in to the Microsoft SQL Server Management Studio as the sysadmin (SA) or a user account with sysadmin privileges.
2. Create a database and user for vCenter Server.
   a. In the master database, create a database for vCenter Server.

    b. Create a database user for vCenter Server and map it to the vCenter Server and msdb databases.
3. Assign the db_owner role to the vCenter Server database user on both the vCenter Server and msdb databases.
4. Enable database monitoring for the vCenter Server database user

See the *vCenter Server Installation and Setup* guide for sample scripts that can be used to create and configure the SQL database.

> **NOTE**: If you use the embedded PostgreSQL database uninstalling vCenter Server on Windows uninstalls the embedded database and all data is lost.

Deploy Platform Services Controller (PSC)

To use external PSC instances, you need to install PSC in a Windows server or deploy a PSC appliance. The PSC provides required services, such as SSO, VMCA and License Service, which can be shared by multiple vCenter Servers. When deploying multiple PSC instances to share the same SSO domain, you must deploy each instance one at a time. VMware does not support concurrent deployments of PSC in the same SSO domain.

Deploy a PSC Appliance using the GUI Installer

You can use the GUI Installer to deploy a PSC appliance as the first instance in a new SSO domain or as a replication partner in an existing SSO domain using this procedure.

1. In the vCenter Server Appliance installer, navigate to the `vcsa-ui-installer` directory, go to the subdirectory for your operating system, and run the installer executable file.
    a. For Windows OS, go to the `win32` subdirectory, and run the `installer.exe` file.
    b. For Linux OS, go to the `lin64` subdirectory, and run the `installer` file.
    c. For Mac OS, go to the `mac` subdirectory, and run the `Installer.app` file.
2. On the Home page, click **Install** to start the deployment wizard.
3. Review the Introduction page to understand the deployment process and click **Next**.
4. Read and accept the license agreement and click **Next**.
5. On the Select deployment type page, select **Platform Services Controller** and click **Next**.
6. Connect to the target server, where you want to deploy the appliance. You have two choices.
    a. Provide the FQDN (or IP address) and credentials for the target ESXi host and provide the appropriate certificate.
    b. Provide the FQDN (or IP address) and credentials for the target vCenter Server (that is managing the hosts on which this new vCenter Server will be deployed), provide the appropriate certificate, and specify the appropriate location in the vSphere Inventory.
7. On the next page of the wizard, set the appliance's name and root user password, following these rules.
    a. The appliance name must not contain a percent sign (%), backslash (\), or forward slash (/) and must be no more than 80 characters in length.
    b. The password must contain only lower ASCII characters without spaces, at least eight characters, a number, uppercase and lowercase letters, and a special character, for example, an exclamation mark (!), hash key (#), at sign (@), or brackets (()).
8. Select an available datastore.

9. On the Configure Network settings page, setup the network settings, such as virtual switch port group, IP configuration, and communication ports.
10. On the Ready to complete page, select **Finish**.
11. Wait for the OVA to deploy, then click Continue to proceed with Stage 2.
12. In Stage 2, click Next on the Introduction page.
13. For time configuration, choose an option.
    a. Synchronize time with the ESXi host.
    b. Synchronize time with NTP Servers
14. Optionally, enable SSH connections into the appliance.
15. Create a new SSO domain or join an existing domain.
    a. Create new: enter domain (such as vsphere.local), set the SSO Administrator (administrator@vsphere.local by default) account password, provide a SSO site name, and confirm the password.
    b. Join an existing SSO domain: enter the PSC FQDN containing the SSO Server, provide the HTTPS port that PSC will use, provide the target SSO domain name (such as vsphere.local), and enter the SSO Administrator account password.
16. Optionally, choose the option to join the VMware Customer Experience Improvement Program (CEIP)
17. On the Ready to complete page, click **Finish** and **OK**.

NOTE: The PSC appliance compute size is set automatically. You cannot set its size during deployment.

NOTE: The supported characters for SSO site names are alphanumeric characters and dash (-). Site names cannot be changed after the PSC installation.

To install a PSC in a Windows server, you can use this procedure.

Step 1. Logon to the Windows Server, double-click on **autorun.exe**.

Step 2. Select **vCenter Server for Windows** and click **Install**.

Step 3. Follow the prompts in the wizard to review the welcome page and accept the license agreement.

Step 4. Select **Platform Services Controller** and click **Next**.

Step 5. Follow the prompts in the wizard to configure the system name, SSO domain, and SSO site name, much like in previous procedure. Additionally, either accept or modify the default ports.

Step 6. When prompted, optionally change the installation folder path and click **Next**.

Step 7. On the Summary page click **Install**.

Step 8. After the installation completes, click **Finish**.

To install vCenter Server and Embedded PSC in a Windows Server, use similar steps as in the previous procedure, but in step 4, select **vCenter Server and Embedded Platform Services Controller**. Also, follow the additional prompts in the wizard to configure the vCenter Server options including the service user account and database.

Configure/Manage VMware Certificate Authority (VMCA)

The VMware Certificate Authority (VMCA), which runs in the PSC, is responsible for issuing certificates for VMware solution users, certificates for machines running required services, and certificates for ESXi hosts. The VMware End Point Certificate Service (VECS) is a local repository for certificates and private keys. VECS is a mandatory component that will be used when VMCA is not signing certificates. The VECS includes a set of keystores including machine SSL certificates, trusted roots, CRLs, and solution users (machine, vpxd, vpx-extension, vSphere-webclient). VECS does not store ESXi Certificates. ESXi certificates are stored locally on the ESXi hosts in the `/etc/vmware/ssl` directory. Table 11-7 describes the stores included in VECS.

Table 11-7 VECS stores.

Store	**Description**
Machine SSL store (MACHINE_SSL_CERT)	Used by the reverse proxy service on each ESXi host and by the vmdir service (on the PSC)
Trusted root store (TRUSTED_ROOTS)	Contains all trusted root certificates.
Solution user stores • Machine • Vpxd • vpxd-extension • vsphere-webclient	VECS includes one store for each solution user.
vSphere Certificate Manager Utility backup store (BACKUP_STORE)	Used by VMCA (VMware Certificate Manager) to support certificate revert.
Other stores	Other stores might be added by solutions. For example, the Virtual Volumes solution adds an SMS store.

With VMCA, you can deal with certificates in three different manners. You can let VMCA operate in a default manner, where it uses a self-signed root certificate, issues certificates to the vSphere components, and serves as the certificate authority (CA) to vSphere. You can configure VMCA to operate as a subordinate CA on behalf of the enterprise's CA and to use a subordinate CA signing certificate. You can bypass VMCA and use only 3rd party certificates, which you will need to store in the VECS, except for ESXi hosts certificates. When necessary, you can use `vecs-cli` commands to explicitly manage certificates and keys.

NOTE: The VMCA in vSphere 6.x does NOT support the use of CRLs nor does it have the concept of certificate revocation. If you suspect one certificate was compromised, you should remove it and consider replacing all certificates.

Using VMCA in the default manner, where it acts as the CA for vSphere, no real configuration is required, other than to configure web browsers to trust VMCA. The VMCA can handle all certificate management in vSphere environments, where historically the administrator has elected not to replace certificates. During an upgrade to vSphere 6.0, all self-signed certificates are replaced with certificates signed by VMCA.

Using VMCA in a subordinate CA manner requires you to replace the VMCA root certificate with a certificate signed by a third party CA, making the VMCA certificate an intermediate certificate of the CA. To use VMCA in the subordinate CA manner, follow this procedure.

Step 1. Launch the vSphere 6.0 Certificate Manger.

Step 2. Select **Option 2**, which is to replace the VMCA root certificate with a custom signing certificate and replace all certificates.

Step 3. When prompted provide the password for the SSO domain administrator account.

Step 4. Select **Option 1**, to generate a Certificate Signing Request (CSR) and key. When prompted, provide a directory to save the CSR and key.

Step 5. Provide the CSR (`root_signing_cert.csr`) to you CA to generate the subordinate signing certificate.

Step 6. Use a text editor to copy content of intermediate CA certificates and the root CA certificate into a single file (`root_signing_chain.cer`).

Step 7. In the Certificate Manger, select **Option 1**, to continue to the step to import custom certificates.

Step 8. Import the root signing certificate (`root_sigining_chain.cer`) and root signing key (`root_signing_cert.key`).

Step 9. When prompted, provide a value for each item, such as country, name, and organization.

Step 10. After completing these steps, the VMCA root certificate is replaced with a custom signing certificate.

For more details on this procedure see VMware KB 2112016.

vCenter Server Installation

You can deploy vCenter Server with an embedded or external Platform Services Controller, but you must install or deploy the Platform Services.

Windows vCenter Server

NOTE: vCenter Server for Windows is deprecated in vSphere 6.7 and will not be available in future releases. To ensure continuous support, deploy a new vCenter Server Appliance or migrate current vCenter Server for Windows installations to vCenter Server Appliance deployments.

To prepare for a vCenter Server for Windows installation, deploy a Windows Server (physical or virtual), meeting the hardware specifications defined in Chapter 1.

Prepare to either use the Local System account to run the vCenter Service or prepare a user account that is a member of the local Administrators group, has the **log on as a service** and **act as part of the operating system** permissions.

Prepare to use the bundled PostgresSQL database or prepare a supported (Microsoft SQL or Oracle) database.

To install vCenter Server with an embedded PSC on Windows, you can follow these steps.

1. Download the installer ISO from VMware and mount the ISO on the Windows Server.
2. In the software installer directory, double-click the `autorun.exe` file to start the installer.
3. Select vCenter Server for Windows and click **Install**.
4. Follow the prompts of the installation wizard to review the welcome page and accept the license agreement.
5. Select **vCenter Server and Embedded Platform Services Controller**,and click **Next**.
6. Enter the system network name, preferably an FQDN, and click **Next**.
7. Set up the new vCenter Single Sign-On domain by providing a domain name (such as `vsphere.local`) and password for the administrator account. Click **Next**.
8. Select the vCenter Server service account and click **Next**.
9. Select the type of database that you want to use and click **Next**.
10. For each component, accept the default port numbers, or if another service is using the defaults, enter alternative ports, and click **Next**.
11. (Optional) Change the default destination folders and click **Next**.
12. Choose if you want to join the VMware Customer Experience Improvement Program (CEIP) page and click **Next**.
13. Review the summary of the installation settings and click **Install**.
14. (Optional) After the installation finishes, click **Launch vSphere Client** to start the vSphere Client and log in to vCenter Server.
15. Click **Finish**

NOTE: Starting with vSphere 6.5, the vCenter Server services run as child processes of the VMware Service Lifecycle Manager service.

vCenter Server Appliance

To prepare for a vCenter Server Appliance or Platform Services Controller (PSC) appliance deployment, you should download the vCenter Server Appliance installer ISO file and mount it to a virtual machine or physical machine from which you want to perform the deployment. To use the vCenter Server Appliance GUI (or CLI) installer, you can use a machine that is running a supported version of Windows, Linux, or Mac operating system, as shown in Table 11.8

Table 11-8– Requirements for GUI / CLI installers

OS	Supported Versions	Minimal Hardware Configuration
Windows	• Windows 7, 8, 8.1, 10 • Windows 2012 x64 bit • Windows 2012 R2 x64 bit • Windows 2016 x64 bit	4 GB RAM, 2 CPU having 4 cores with 2.3 GHz, 32 GB hard disk, 1 NIC
Linux	• SUSE 12 • Ubuntu 14.04	4 GB RAM, 1 CPU having 2 cores with 2.3 GHz, 16 GB hard disk, 1 NIC CLI Installer requires 64 bit OS.
Mac	• MacOS v10.9, 10,10, 10.11 • MacOS Sierra	

Using the GUI or CLI Installers, you can:

- Deploy and upgrade the vCenter Server Appliance and PSC appliance.
- Migrate Windows installations of vCenter Server, vCenter Single Sign-On, and PSC to the vCenter Server Appliance and PSC appliance.
- Restore a vCenter Server Appliance from a file-based backup.

Deploy VCSA with Embedded PSC Using the GUI Installer

You can use the GUI installer to deploy a vCenter Server Appliance with an embedded Platform Services Controller, a Platform Services Controller appliance, or a vCenter Server Appliance with an external Platform Services Controller. To perform a GUI based deployment, you download the vCenter Server

Appliance installer on a network client machine, run the deployment wizard from the client machine, and provide the required information.

NOTE: For topologies with external Platform Services Controller instances, you must deploy the replicating Platform Services Controller instances in a sequence. After you successfully deploy all the Platform Services Controller instances in the domain, you can perform concurrent deployments of multiple vCenter Server appliances.

Using the GUI Installer involves two stages. In the first stage, you navigate through the installation wizard, choose the deployment type, provide the appliance settings, and deploy the OVA. Alternatively, you could use the vSphere Web Client or the vSphere Host Client to deploy the OVA.

In the second stage using the GUI Installer, you use a wizard to configure the appliance time synchronization, configure vCenter Single Sign-On (SSO), and start the services in the newly deployed appliance. Alternatively, you can use a web browser to access the appliance's VMware Appliance Management Interface (VAMI) at https://*FQDN*:5480. If you use the alternative approach in the first stage, then you must use it in the second stage. To use the GUI Installer to deploy the VCSA with an Embedded PSC, you can follow these steps.

1. In the vCenter Server Appliance installer, navigate to the `vcsa-ui-installer` directory, go to the subdirectory for your operating system, and run the installer executable file.
    a. For Windows OS, go to the `win32` subdirectory, and run the `installer.exe` file.
    b. For Linux OS, go to the `lin64` subdirectory, and run the `installer` file.
    c. For Mac OS, go to the `mac` subdirectory, and run the `Installer.app` file.
2. On the Home page, click **Install** to start the deployment wizard.
3. Review the Introduction page to understand the deployment process and click **Next**.
4. Read and accept the license agreement and click **Next**.
5. On the Select deployment type page, select **vCenter Server with an Embedded Platform Services Controller** and click **Next**.
6. Connect to the target server, where you want to deploy the appliance. You have two choices.
    a. Provide the FQDN (or IP address) and credentials for the target ESXi host and provide the appropriate certificate.
    b. Provide the FQDN (or IP address) and credentials for the target vCenter Server (that is managing the hosts on which this new vCenter Server will be deployed), provide the appropriate certificate, and specify the appropriate location in the vSphere Inventory.
7. On the next page of the wizard, set the appliance's name and root user password, following these rules.
    a. The appliance name must not contain a percent sign (%), backslash (\), or forward slash (/) and must be no more than 80 characters in length.
    b. The password must contain only lower ASCII characters without spaces, at least eight characters, a number, uppercase and lowercase letters, and a special character, for example, an exclamation mark (!), hash key (#), at sign (@), or brackets (()).
8. Select the deployment size. Choose Tiny, Small, Medium, Large, X-Large as explained in Chapter 1.
9. Select the storage size for the appliance as explained in Chapter 1.
10. Select an available datastore.
11. On the Configure Network settings page, setup the network settings, such as virtual switch port group, IP configuration, and communication ports.
12. On the Ready to complete page, select **Finish**.
13. Wait for the OVA to deploy, then click **Continue** to proceed with Stage 2.

14. In Stage 2, click **Next** on the Introduction page.
15. For time configuration, choose an option.
    a. **Synchronize time with the ESXi host**.
    b. Sychronize time with NTP Servers
16. Optionally, enable SSH connections into the appliance.
17. Create a new SSO domain or join an existing domain.
    a. Create new: enter domain (such as vsphere.local), set the SSO Administrator (administrator@vsphere.local by default) account password, provide a SSO site name, and confirm the password.
    b. Join an existing SSO domain: enter the PSC FQDN containing the SSO Server, provide the HTTPS port that PSC will use, provide the target SSO domain name (such as vsphere.local), and enter the SSO Administrator account password.
18. Optionally, choose the option to join the VMware Customer Experience Improvement Program (CEIP)
19. On the Ready to complete page, click **Finish** and **OK**.

Deploy a PSC Appliance using the GUI Installer

You can use the GUI Installer to deploy a PSC appliance as the first instance in a new SSO domain or as a replication partner in an existing SSO domain using this procedure.

18. In the vCenter Server Appliance installer, navigate to the `vcsa-ui-installer` directory, go to the subdirectory for your operating system, and run the installer executable file.
    a. For Windows OS, go to the `win32` subdirectory, and run the `installer.exe` file.
    b. For Linux OS, go to the `lin64` subdirectory, and run the `installer` file.
    c. For Mac OS, go to the `mac` subdirectory, and run the `Installer.app` file.
19. On the Home page, click **Install** to start the deployment wizard.
20. Review the Introduction page to understand the deployment process and click **Next**.
21. Read and accept the license agreement and click **Next**.
22. On the Select deployment type page, select **Platform Services Controller** and click **Next**.
23. Connect to the target server, where you want to deploy the appliance. You have two choices.
    a. Provide the FQDN (or IP address) and credentials for the target ESXi host and provide the appropriate certificate.
    b. Provide the FQDN (or IP address) and credentials for the target vCenter Server (that is managing the hosts on which this new vCenter Server will be deployed), provide the appropriate certificate, and specify the appropriate location in the vSphere Inventory.
24. On the next page of the wizard, set the appliance's name and root user password, following these rules.
    a. The appliance name must not contain a percent sign (%), backslash (\), or forward slash (/) and must be no more than 80 characters in length.
    b. The password must contain only lower ASCII characters without spaces, at least eight characters, a number, uppercase and lowercase letters, and a special character, for example, an exclamation mark (!), hash key (#), at sign (@), or brackets (()).
25. Select an available datastore.
26. On the Configure Network settings page, setup the network settings, such as virtual switch port group, IP configuration, and communication ports.
27. On the Ready to complete page, select **Finish**.
28. Wait for the OVA to deploy, then click Continue to proceed with Stage 2.
29. In Stage 2, click Next on the Introduction page.
30. For time configuration, choose an option.
    a. Synchronize time with the ESXi host.
    b. Synchronize time with NTP Servers
31. Optionally, enable SSH connections into the appliance.

32. Create a new SSO domain or join an existing domain.
    a. Create new: enter domain (such as vsphere.local), set the SSO Administrator (administrator@vsphere.local by default) account password, provide a SSO site name, and confirm the password.
    b. Join an existing SSO domain: enter the PSC FQDN containing the SSO Server, provide the HTTPS port that PSC will use, provide the target SSO domain name (such as vsphere.local), and enter the SSO Administrator account password.
33. Optionally, choose the option to join the VMware Customer Experience Improvement Program (CEIP)
34. On the Ready to complete page, click **Finish** and **OK**.

> **NOTE**: The PSC appliance compute size is set automatically. You cannot set its size during deployment.

Deploy VCSA with External PSC Using the GUI Installer

To use the GUI Installer to deploy the VCSA with an External PSC, you can first deploy the PSC and then use the previously provided procedure for deploying VCSA with Embedded PSC, but with the following modifications.

- On Step 5, choose **vCenter Server (Requires External Platform Services Controller)**
- On Step 17, provide the FQDN (or IP address) of the PSC with which you want to register the VCSA. Enter the SSO port, domain name, and administrator credentials.

CLI Deployment

You can use the CLI installer to perform a silent deployment of a VCSA or PSC appliance. The CLI deployment process includes downloading the installer, preparing a JSON configuration file with the deployment information, and running the deployment command. The VCSA Installer contains JSON templates for all deployment types. This enables you to deploy an appliance with minimum effort by copying the appropriate JSON template, changing a few values, and using it with the CLI Installer. The steps are:

1. In the vCenter Server Appliance installer, navigate to the `vcsa-cli-installer` directory, and then to the `templates` subfolder.
2. Copy the templates from the `install` subfolder to your desktop.
3. Use a text editor to modify the JSON template for your use case. Modify the default parameter values with your appropriate values and add additional parameters as necessary. For example, to use an IPv4 DHCP assignment, in the `network` subsection of the template, change the value of the `mode` parameter to `dhcp` and remove the default configuration parameters that are used for a static assignment, as shown here.

```
      "network": {

              "ip_family": "ipv4",

               "mode": "dhcp"

              },
```

4. Save the file in UTF-8 format.

Table 11-9 shows some of the available JSON templates.

Table 11-9 JSON Templates Sample.

Template	Description
PSC_replication_on_ESXi.json	Contains the minimum configuration parameters that are required for deployment of a Platform Services Controller appliance joining an existing vCenter Single Sign-On domain on an ESXi host.
PSC_replication_on_VC.json	Contains the minimum configuration parameters that are required for deployment of a Platform Services Controller appliance joining an existing vCenter Single Sign-On domain on a vCenter Server instance.
vCSA_on_ESXi.json	Contains the minimum configuration parameters that are required for deployment of a vCenter Server Appliance with an external Platform Services Controller on an ESXi host.
vCSA_on_VC.json	Contains the minimum configuration parameters that are required for deployment of a vCenter Server Appliance with an external Platform Services Controller on a vCenter Server instance.

NOTE: When using the CLI Installer, you must strictly use only ASCII characters for the command line and JSON configuration file values, including usernames and passwords.

Prior to running the deployment command, you can run the pre-deployment check using this command.

```
vcsa-deploy install --verify-only  path-to-JSON-file
```

When ready, you can run the deploy command.

```
vcsa-deploy install --accept-eula --acknowledge-ceip
optional_arguments path-to-JSON-file
```

Post installation

After installing vCenter Server, you should be able to access the vSphere Client.at `https://vcenter_server_ip_address_or_fqdn/ui` and the vSphere Web Client at `https://vcenter_server_ip_address_or_fqdn/vsphere-client.`

The VMware Enhanced Authentication Plug-in provides Integrated Windows Authentication and Windows-based smart card functionality. In the vSphere 6.5 release, the VMware Enhanced Authentication Plug-in replaced the Client Integration Plug-in.

To install the VMware Enhance Authentication Plug-in you can use the following procedure. If you install the plug-in from an Internet Explorer browser, you must first disable Protected Mode and enable pop-up windows on your Web browser.

1. In a Web browser, open the vSphere Web Client.
2. At the bottom of the vSphere Web Client login page, click **Download Enhanced Authentication Plug-in**.
3. Save the plug-in to your computer and run the executable.
4. Follow the wizard for both the VMware Enhanced Authentication Plug-in and the VMware Plug-in Service which are run in succession.
5. When the installations are complete, refresh the browser.
6. On the External Protocol Request dialog box, click **Launch Application**

If an external PSC stops responding or you want to distribute the load, you can repoint vCenter Server to another External PSC in the same domain and site. For example, to repoint a vCenter Server Appliance (VCSA) to a PSC named `psc01.domain.local`, you can enter the following command in the VCSA shel.

```
   cmsso-util repoint --repoint-psc psc01.domain.local
```

You can move one vCenter Server to a Platform Services Controller in another vSphere domain. The use cases are:

- Consolidate vSphere domains
- Migrate a vCenter Server to a newly created domain
- Migrate a vCenter Server using an embedded PSC to an external PSC

Prerequisite: PSC and vCenter Server 6.7

Procedure:

1. (Optional) Run the pre-check mode command, which creates a JSON file containing any discovered conflicts.
2. (Optional) Edit the conflict file and apply resolutions for all conflicts or apply a separate resolution for each conflict. The conflict resolutions are:
    a. Copy: Create a duplicate copy of the data in the target Platform Services Controller.
    b. Skip: Skips copying the data in the target Platform Services Controller.
    c. Merge: Merges the conflict without creating duplicates.
3. Run the execute mode command and provide the administrator password when prompted.

The syntax for for the execute command is:

```
cmsso-util domain-repoint --mode execute --src-psc-admin
SourcePSC_admin_userid --dest-psc-fqdn

target_PSC_FQDN --dest-psc-admin target_PSC_admin_userid

--dest-domain-name target_FQDN --dest-vcfqdn target_vCenter_server
```

> **NOTE**: The `-mode` value for the `cmsso-util` command can be set to `pre-check` or `execute`.

If you deployed or installed a standalone vCenter Server instance with an embedded Platform Services Controller and you want to extend your vCenter Single Sign-On domain with more vCenter Server instances, you can reconfigure and repoint the existing vCenter Server instance to an external Platform Services Controller. To do so, ensure the target PSC is running and is a replication partner with the embedded PSC and use the following command.

```
cmsso-util reconfigure --repoint-psc external_psc_fqdn

--username username -- domain-name domain_name --passwd passwor>
```

Objective 4.7

In addition to deploying one or more PSCs and creating an SSO domain in a new vSphere environment, you need to configure SSO. Configuring SSO includes adding and editing SSO identity sources, configuring SSO users, and configuring SSO policies.

SSO and Identity Sources Overview

To access vCenter Server, users must login using SSO domain user accounts or user accounts from identity sources registered in SSO. The acceptable identity sources are Active Directory (Integrated Windows Authentication), Active Directory as a LDAP Server, Open LDAP and Local OS.

The Local OS identity source is available immediately following the installation of SSO. Local operating system users are local to the operating system where the vCenter Single Sign-On server is running. The local operating system identity source exists only in basic vCenter Single Sign-On server deployments and is not available in deployments with multiple vCenter Single Sign-On instances. Only one local operating system identity source is allowed.

The SSO domain is also available immediately as an identity source.   This domain was called vsphere.local in vSphere 5.5, but in vSphere 6.x, you may assign the SSO domain name during installation.

Add/Edit/Remove SSO Identity Sources

You can use the vSphere Client to add SSO identity sources using the following procedure

1. Log in with the vSphere Client to the vCenter Server connected to the PSC using administrator@vsphere.local or another member of the SSO Administrators group.
2. Navigate to **Home** > **Administration** > **Single Sign On** > **Configuration**
3. Click **Identity Sources** and click **Add Identity Source**
4. Select one of the following available identity sources and enter the appropriate settings
    a. Active Directory (Integrated Windows Authentication)
    b. Active Directory over LDAP
    c. OpenLDAP
    d. Local operating system of SSO server.
5. Click **Add**.

To remove an SSO identity source, you can use the vSphere Web Client to select the identity source on the **Identity Sources** tab at **Administrator** > **Single Sign-On** > **Configuration** and click the **Delete Identity Source** icon. When prompted, click **Yes** to confirm.

You can configure a default domain for SSO. The default SSO domain allows users to authenticate without identifying a domain name. Users from other identity sources must identify the domain name during authentication. To configure a default domain using the vSphere Client, you can use these steps.

1. Navigate to **Home** > **Administration** > **Single Sign On** > **Configuration**
2. Click **Identity Sources** and click **Add Identity Source**
3. Select an identity source and click **Set as Default**.

How to Add an Active Directory Identity Source

To permit Active Directory authentication in vSphere, add one or more Active Directory domains as identity sources in SSO. In scenarios, where the SSO server is a member of an Active Directory domain, that domain may be added as **Active Directory (Integrated Windows Authentication)** identity source. You can add other Active Directory domains to SSO as **Active Directory LDAP Server** identity sources.

As a requirement to add an integrated Active Directory identity source, you need to ensure the server where SSO is installed is in the domain. For a Windows server, you should ensure the server is added to the Active Directory domain prior to installing SSO. For a PSC appliance or vCenter Server appliance, you can add the appliance to the domain using the following procedure.

Step 1. Logon to the vSphere Web Client using the SSO domain administrator account, such as administrator@vsphere.local.

Step 2. In the left pane, select **Administration** > **System Configuration** > **Nodes**.

Step 3. Select the appropriate node and click the **Manage** tab.

Step 4. Select **Active Directory** and click **Join**.

Step 5. Enter the Active Directory details such as **Domain**, **Organizational Unit**, **User Name**, and **Password**.

Step 6. Click **OK**.

Step 7. Right-click the node and select **Reboot**.

After the appliance reboots, you add it as an Active Directory (Integrated Windows Authentication) identity source

When adding an Active Directory (Integrated Windows Authentication) identity source, provide the following information

- **Domain Name**: FDQN of the domain
- **Use Machine Account**: Select this option to use the local machine account as the Server Principal Name (SPN). Do not use this option if you plan to rename the machine.
- **Use Service Principal Name (SPN):** Select this option, instead of Use Machine Account, if you prefer to specify a unique SPN, instead of using the machine name as the SPN. If you choose this option, then you mast also provide the SPN, UPN, and password as follows.
- **Service Principal Name (SPN):** If you selected the `Use Service Principal Name` option, then provide a unique name that includes the domain name, such as `STS/domain.com`.

- **User Principal Name (UPN):** If you selected the `Use Service Principal Name` option, then provide a user name that can authenticate to the Active Directory domain.
- **Password**: If you selected the `Use Service Principal Name` option, then provide a password that is associated with the UPN.

> **NOTE**: The user account must have read access to the OUs containing users and group. If the user account does not have sufficient permission or is locked or disabled, then authentications and searches in the Active Directory domain fail

When adding an Active Directory over LDAP identity source, provide the following information

- **Name**: Logical name for the identify source
- **Base DN for users**: Base Distinguished Name for users.
- **Domain Name**: FDQN of the domain
- **Domain Alias**: The Domain's NetBIOS name.
- **Base DN for groups**: Base Distinguished Name for groups.
- **Primary Server URL:** The primary domain controller's URLin the form of ldap://*hostname*:*port*, or ldaps://*hostname*:*port*.
- **Secondary Server URL:** The secondary domain controller's URLin the form of ldap://*hostname*:*port*, or ldaps://*hostname*:*port*.
- **Choose certificate:** When using LDAPS in the URL parameters, specify the certificate.
- **Username:** User name in the domain that has at least read access to the specified user and group base DNs.
- **Password**: Password that is associated with the user name.

You can add additional user accounts from other Identity Sources to the SSO Administrators group. To add additional user accounts from other Identity Sources to the `Administrators` group in the SSO domain, you can follow these steps:

Step 1. Login to vSphere Web Client with the SSO domain administrator account.

Step 2. Select **Administration** > **Users / Groups**

Step 3. Select the **Group tab** > **Administrators** > **Add Member** icon from the Group Members section

Step 4. **Select the additional Identity Source from the Domain drop down menu.**

Step 5. **Select the account you would like to add**

Step 6. **Click OK**

Objective 4.8

Enable/Disable Single Sign-On (SSO) Users

To manage SSO users, you could use the vSphere Client. For example, to add a SSO user, follow these steps.

1. Logon to the vCenter Server connected to PSC using administrator@vsphere.local or another user in the SSO Adminstrators group.
2. Navigate to **Home** > **Administration** > **Single Sign-On** > **Users and Groups**
3. Select the **Users** tab and click **Add User.**
4. Provide the **User Name** and **Password**. Optionally provide values for the other fields.
5. Click **OK**.

In a similar manner, you can create an SSO group by selecting **Users** tab in Step 3 and providing details in Step 4. You can also use the **Groups** tab to select a group and use the **Add Member** icon (in the details section) to add users to the group. When adding a user to a group, use the **Domain** dropdown to select the SSO domain or another identity source and select a user account from the provided list.

To disable or enable an SSO user account, select the user account in **Users and Groups,** click the ellipsis icon**,** and click **Disable** or **Enable.**

The SSO domain (`vsphere.local` in vSphere 5.5, but may be named differently in vSphere 6.0) provides several pre-defined groups. You can add users from Active Directory domains or other identity sources to these pre-defined groups. Some SSO privileges are determined solely by the membership of these groups. For example, a user who is a member of the `CAAdmins` group can manage VMCA and a user who is a member of the `LicenseService.Administrators` group can manage licenses.

The SSO domain contains many pre-defined groups, including the following:

- Users: Contains all users in the SSO domain
- DCAdmins: Members can perform Domain Controller Administrator actions on the VMware Directory Service.
- SolutionUsers: Each solution user authenticates individually to vCenter Single Sign-On with a certificate. By default, VMCA provisions solution users with certificates. Do not add members to this group explicitly.
- CAAdmins: Members have administrator privileges for VMCA. Adding members to these groups is not usually recommended, but a user must be a member of this group to perform most certificate management operations, such as using the `certool` command.

- SystemConfiguration.BashShellAdministrators: Only applies to vCenter Server Appliance deployments. Members can enable and disable access to the BASH Shell.
- SystemConfiguration.Administrators: Members can view and manage the system configuration and perform tasks such as restarting services.
- LicenseSevice.Administrators: Members have full write access to all licensing related data and can add, remove, assign, and un-assign serial keys for all product assets registered in licensing service.
- Administrators: Members can perform SSO administration tasks for the VMware Directory Service (vmdir)

Configure SSO Policies

VCP6-DCV Cert Guide: Chapter 1 – Configure SSO Policies

Extra: Installing Other vSphere Components

Other important VMware components in a vSphere environment are VMware Update Manager, vSphere Client, and vSphere Web Client.

vSphere Client Implementation

The vSphere Client is HTML5 based and uses the "Clarity" style user interface. You should use this as your primary GUI and only use the legacy vSphere Web Client when necessary. The vSphere Client is a service that is installed automatically as you install vCenter Server.

vSphere Web Client Implementation

In vSphere 6.7 the vSphere Client (HTML5 "Clarity" Client) has 95% feature parity with the vSphere Web Client. For example, although the vSphere Client can now be used for some VMware Update Manager (VUM) operations, the Sphere Web Client is still available and required for the following VUM operations:

- Update Manager configuration changes
- VMware Tools & VM Hardware updates
- Viewing Events and Notifications from the Update Manager interface
- Indicating which hosts are Quick Boot capable or disabling Quick Boot

The vSphere Web Client is Flex based and is typically only used when necessary. The vSphere Web Client is a service that is installed automatically as you install vCenter Server.

VMware Update Manager Implementation

Update Manager enables centralized, automated patch and version management for VMware vSphere and offers support for VMware ESXi hosts, virtual machines, and virtual appliances. The VMware vSphere Update Manager Extension is an optional service of the vCenter Server Appliance 6.7. Starting in vSphere 6.5, you can no longer connect Update Manager instance that is installed on a Windows Server machine with the vCenter Server Appliance.

The Update Manager server 6.7 for Windows is a 64-bit application, which can only be installed on a 64-bit Windows operating system. Before you install Update Manager, you must install vCenter Server on Windows. The Update Manager server has the same requirements for a host OS like the vCenter Server.

VMware Update Manager requires a database. You can select from the embedded (MS SQL Express) database, an external MSSQL database, or an Oracle database,

Configure ESXi Using Host Profiles

Host Profile Overview

During the implementation of vSphere, you can use Host Profiles to efficiently deploy a standard configuration to a set of ESXi hots. To do so, you could configure a single ESXi host, create a host profile from that host, then apply the profile to other recently deployed hosts. This process reduces the time required to configure ESXi hosts and minimizes the risk of misconfigured hosts. The host profile contains all the networking, storage, security, and other host-level settings. The host from which the profile is created is known as the reference host. You can attach a host profile to individual hosts, a cluster, or all the hosts and clusters managed by a vCenter Sever. Applying a Host Profile. After attaching the profile, you can check compliance of the associated hosts and remediate as necessary.

You can use this procedure to create a host profile from a reference host.

1. Navigate to the Host Profiles main view and click **Extract Host Profile**.
2. Select the host that acts as the reference host and click **Next**.
3. Enter the name and description for the new profile and click **Next**.
4. Review the summary information for the new profile and click **Finish**.

You can use this procedure to attach a profile to ESXi hosts and clusters.

1. From the Host Profiles main view, select the host profile to be applied to the host or cluster.
2. Click **Attach/Detach** a host profile to hosts and clusters.
3. Select the host or cluster from the expanded list and click **Attach**.
4. Optionally, click **Attach All** to attach all listed hosts and clusters to the profile.
5. If you **enable Skip Host Customization** you will not need to customize hosts during this process.
6. Click **Next**.
7. Optionally, you can update or change the user input parameters for the Host Profiles policies by customizing the host.
8. Click **Finish** to complete attaching the host or cluster to the profile.

You can use this procedure to remediate an ESXi host.

1. Navigate to Host Profiles main view.
2. Right-click the host profile and select **Remediate**.
3. Select the host or hosts you want to remediate with the host profile.
4. Optionally, enter the host customizations to specify host properties or browse to import a host customization file.
5. Click **Pre-check Remediation** to check if the selected hosts are ready for remediation.
6. Select the checkbox to reboot the host if it is required in order to complete the remediation process. If you wish to manually reboot the host after the process, do not select the checkbox.
7. Review the tasks that are necessary to remediate the Host Profile and click **Finish**.

Edit Host Profiles

To edit a host profile, you can use this procedure

1. Navigate to Host Profiles main view.
2. Select the host profile that you want to edit and click the **Configure** tab.
3. Click **Edit Host Profile**.
4. Optionally, click the **Name and description** tab to change the profile name and description.
5. In the Edit host profile page expand each category to view or edit a specific policy or setting.
6. View **All** host profile configurations or only **Favorites** configurations.
7. Optionally, in the search field, filter the configuration names and values you want to view. For example, enter `SNMP`. All configurations that relate to SNMP are displayed.
8. Optionally, customize the hosts. Make any changes to the available configuration values for this profile and click **Save**

Apply Permissions to ESXi Hosts Using Host Profiles

You can use host profiles to apply ESXi host permissions to be used when users access the host directly. To configure the host profile with the appropriate permissions, you can use the vSphere Client (not the vSphere Web Client) and follow this procedure.

Step 1. Select **View** > **Management** > **Host Profiles**.

Step 2. Select an existing profile and click **Edit Profile**.

Step 3. In the profile tree, locate and expand **Security configuration**.

Step 4. Right click on the **Permission rules** folder and click **Add Profile**.

Step 5. Expand **Permissions rules** and select **Permission**.

Step 6. On the **Configuration Details** tab, click the **Configure permission** drop down menu and select **Require a Permission Rule**.

Step 7. Enter the name of a user or group. Use the format `domain\name`, where `domain` is the domain name and `name` is the user or group name.

Step 8. If a group name is used, select the **Name refers to a group of users** checkbox.

Step 9. Enter the assigned role name, which is case sensitive. This can the name of a built-in role on the host or a custom role that you created on the host. For system roles, use the non-localized role name, such as `Admin` for the Administrator role or `ReadOnly` for the Read-only role.

Step 10. Optionally, select **Propagate permission**.

Step 11. Click **OK**.

After configuring the host profile, you can use it to apply the permissions to new or existing ESXi hosts.

VMware Tools

Ideally, you should install VMware Tools in all your virtual machines. When deploying a new vSphere Environment, you should install VMware Tools in any virtual machines that you deployed as part of the virtual infrastructure and management. For example, if you use virtual machines to run Active Directory domain controllers, DNS servers, or DHCP servers, consider installing VMware Tools.

VMware Tools is a suite of utilities that you install in the operating system of a virtual machine. VMware Tools enhances the performance and management of the virtual machine. You can use the following procedure to install VMware Tools in a virtual machine using the VMware Host Client. This procedure is useful for installing VMware Tools in a DNS, Active Directory domain controller, database server, or other virtual machine that you may deploy prior to deploying vCenter Server.

Procedure

1. Click **Virtual Machines** in the VMware Host Client inventory.
2. Select a powered-on virtual machine from the list. (The virtual machine must be powered on to install VMware Tools)_
3. Open a console to the virtual machine and log with administrator or root privileges.
4. Click **Actions**, select **Guest OS** from the drop-down menu, and select **Install VMware Tools**.
5. Use the guest OS to complete the installation.

Objective 5.1

VCP6-DCV Cert Guide: Chapter 9 – Objective 6.1 - Create/delete/consolidate virtual machine snapshots

Use Cases

Snapshots are useful as a short term solution for testing software with unknown or potentially harmful effects. For example, you can use a snapshot as a restoration point during a linear or iterative process, such as installing update packages, or during a branching process, such as installing different versions of a program. Using snapshots ensures that each installation begins from an identical baseline

What a snapshot preserves

A snapshot preserves the following information:

- Virtual machine settings. The virtual machine directory, which includes the disks added or changed after you take the snapshot.

- Power state. The virtual machine can be powered on, powered off, or suspended.

- Disk state. State of all the virtual machine's virtual disks.

- (Optional) Memory state. The contents of the virtual machine's memory.

Operations

Several operations for creating and managing virtual machine snapshots and snapshot trees are available in the vSphere Client. These operations enable you to create snapshots, restore any snapshot in the snapshot hierarchy, delete snapshots, and more. You can create snapshot trees where you save the virtual machine state at any specific time so that you can restore that virtual machine state later. Each branch in a snapshot tree can have up to 32 snapshots.

A snapshot consists of files that are stored on a supported storage device. A Take Snapshot operation creates .vmdk, -delta.vmdk, .vmsd, and .vmsn files. By default, the first and all delta disks are stored with the base .vmdk file. The .vmsd and .vmsn files are stored in the virtual machine directory.

vSphere Replication preserves snapshot history. If a snapshot was created and replicated, you can recover to the application consistent snapshot. If you enabled multiple point-in-time instances when you configured replication for the virtual machine, vSphere Replication presents the retained instances as standard snapshots after a successful recovery. You can select one of these snapshots to revert the virtual machine. vSphere Replication does not preserve the memory state when you revert to a snapshot. After performing a successful recovery on the target vCenter Server site, you can perform a failback. To do so, you log in to the target site and manually configure replication in the reverse direction—from the target site to the source site. The disks on the source site are used as replication seeds. Before doing this, you must unregister the virtual machine from the inventory on the source site.

Objective 5.2

VCSA monitoring (VAMI): https://<FQDN/IP>:5480

monitor, access, networking, time, services, update, admin, syslog, backup

VCP6-DCV Cert Guide: Chapter 10 – Monitor Status of vCenter Server Database

VCP6-DCV Cert Guide: Chapter 10 – Monitor Status of ESXi Agents

VCP6-DCV Cert Guide: Chapter 10 – Monitor ESXi System Health

VCP6-DCV Cert Guide: Objective 7.4—Troubleshoot and Monitor vSphere Performance

Objective 5.3

VCP6-DCV Cert Guide: - Configure Advanced vSphere Virtual Machine Settings

Consider the impact to the VM in each of these cases:

Compute oversize / undersize

Virtual disk oversize / undersize

VMDK provisioning types

Resource reservations

Independent disks

Guest OS type

VMware Tools version

Permissions

Objective 7.1

Add and Configure vDS dPort Groups

VCP6-DCV Cert Guide: Chapter 2: Configure vSphere Distributed Switch General and dvPort Group Setting

VCP6-DCV Cert Guide: Chapter 2: Configure VLAN/PVLAN Settings for VMs Given Communication Requirements

Adding a Host to a vDS

VCP6-DCV Cert Guide: Chapter 2 - Add/Remove ESXi Hosts from a vSphere Distributed Switch

Steps required per vDS to support a change on a physical switch

Configure LACP on dvUplink and dvPort Groups

Only vDS support Link Aggregation Control Protocol (LACP). This support was first introduced in vSphere 5.1 and, as with all features introduced in 5.1 or later, is accessible only via the vSphere Web Client. LACP allows a logical grouping of physical network links to achieve greater bandwidth by using the multiple links as one logical connection. Unfortunately, when LACP is utilized, you cannot configure port mirroring, and you cannot include LACP in host profiles.

In order to configure uplink port groups for LACP, you must create a link aggregation group (LAG). Make sure you are in the networking inventory view and then select the vDS switch on which you want to set up LACP. Next, select Manage>Settings >LACP. Now click the green plus sign to create the LAG and provide a descriptive name, such as LAG-Production. Then select the number of ports you want to participate in the LAG group and set Mode to Passive.

The other setting for Mode is Active, which means that all LAG ports are in an active negotiating mode. The LAG ports initiate negotiations with the LACP port channel on the physical switch by sending LACP packets.

LAG ports that are in passive negotiating mode respond to LACP packets they receive but do not initiate LACP negotiation.

Then select the desired option from the Load Balancing Mode drop-down. In this case, you want the setting Source and Destination IP Address, but there are many other options, including the following:

- Destination, Source, or Source and Destination IP Address

- Destination, Source, or Source and Destination IP Address and TCP/UDP Port
- Destination, Source, or Source and Destination IP Address and VLAN
- Destination, Source, or Source and Destination IP Address, TCP/UDP Port and VLAN
- Destination, Source, or Source and Destination MAC Address
- Destination, Source, or Source and Destination TCP/UDP Port
- Source Port ID
- VLAN

NOTE LACP must be configured on a physical switch in order to successfully establish a connection.

To configure LACP on the dvUplink:

Step 1. Right-click vDS-01 and select Add and Manage Hosts.

Step 2. Select Manage Host Networking and click Next.

Step 3. Click the Attached Hosts icon and select esxi03.vsphere.local

Step 4. Ensure that only Manage Physical Adapters is selected and click Next.

Step 5. Select vmnic1 and click Assign Uplink

Step 6. Select LAG-Production-0, where 0 is the first port in the LAG group, and click OK.

Step 7. Select vmnic2 and click Assign Uplink.

Step 8. SelectLAG-Production-1, where 1 is the second port in the LAG group, and click OK.

Migrate VMs between vSS and vDS

VCP6-DCV Cert Guide: Migrate Virtual Machines to/from a vSphere Distributed Switch

Restore vDS

While the vDS configuration is stored in the vCenter database, the ability to backup and restore vDS configurations separate from restoring the entire vCenter database can be very helpful. Especially when organizations need to perform upgrades or configuration changes and roll back if necessary.

The process to backup a vDS configuration:

Step 1. Open the vSphere Client and navigate to the Networking inventory view.

Step 2. Right-click on the vDS you wish to back up.

Step 3. Select **Settings** -> **Export Configuration.**

Step 4. Select the appropriate option:

1. export the vDS configuration only
2. export the vDS configuration and all port groups

Step 5. Enter information into the **Descriptions** box to provide more information, if desired.

Step 6. Click **OK**.

Step 7. Select **Yes** to save the vDS configuration (and define location to save this file).

The process to restore a vDS configuration:

Step 1. Open the vSphere Client and navigate to the Networking inventory view.

Step 2. Right-click the vDS you wish to restore.

Step 3. Select **Settings** -> **Restore Configuration.**

Step 4. Select the appropriate option:

1. **Restore distributed switch and all port groups**
2. **Restore distributed switch only**

Step 5. Click **Next**.

Step 6. Verify the restoration summary and click **Finish.**

Objective 7.2

VCP6-DCV Cert Guide:

- Chapter 4 – Objective 3.1
- Chapter 5 – Objective 3.3
- Chapter 6 - Objective 3.4

- *when* to use iSCSI port binding is found in KB 2038869: Port binding is used in iSCSI when multiple VMkernel ports for iSCSI reside in the same broadcast domain and IP subnet to allow multiple paths to an iSCSI array that broadcasts a single IP address.
- *when not to* use iSCSI port binding: Array Target iSCSI ports are in a different broadcast domain and IP subnet. VMkernel ports used for iSCSI connectivity exist in a different broadcast domain, IP subnet and/or vSwitch.

Pp 90: Best Practices for Configuring Networking with Software iSCSI:

Software iSCSI Port Binding

You can bind the software iSCSI initiator on the ESXi host to a single or multiple VMkernel ports, so that iSCSI traffic flows only through the bound ports. When port binding is configured, the iSCSI initiator creates iSCSI sessions from all bound ports to all configured target portals.

See the following examples.

2 bound VMkernel ports, 2 target portals: 4 sessions (2 x 2)

4 bound VMkernel ports, 1 target portal: 4 sessions (4 x 1)

2 bound VMkernel ports, 4 target portals: 8 sessions (2 x 4)

Objective 7.3

VCP6-DCV Cert Guide: Chapter 5 – Objective 3.2

vSphere 6.7 Storage guide

SPBM pp 241

Within a software-defined data center, Storage Policy Based Management (SPBM) plays a major role by helping to align storage with application demands of your virtual machines. As an abstraction layer, SPBM abstracts storage services delivered by Virtual Volumes, vSAN, I/O filters, or other storage entities.

Diagram on pp 241

SPBM offers the following mechanisms:

- Advertisement of storage capabilities and data services that storage arrays and other entities, such as I/O filters, offer.

- Bidirectional communications between ESXi and vCenter Server on one side, and storage arrays and entities on the other.

- Virtual machine provisioning based on VM storage policies.

VM storage policies pp 242

Virtual machine storage policies are essential to virtual machine provisioning through SPBM. The policies control which type of storage is provided for the virtual machine and how the virtual machine is placed within storage. They also determine data services that the virtual machine can use.

You use the VM Storage Policies interface to create a storage policy. When you define the policy, you specify various storage requirements for applications that run on the virtual machines. You can also use storage policies to request specific data services, such as caching or replication, for virtual disks.

You apply the storage policy when you create, clone, or migrate the virtual machine. After you apply the storage policy, the SPBM mechanism assists you with placing the virtual machine in a matching datastore.

Workflow for VM Storage Policy:

- Populate the VM Storage Policies interface with appropriate data.

- Create predefined storage policy components.

- Create VM storage policies.

- Apply the VM storage policy to the virtual machine.

- Check compliance for the VM storage policy.

About rules and rule sets. Pp 247

Rule: The rule is a basic element of the VM storage policy. Each individual rule is a statement that describes a single requirement for virtual machine storage and data services.

Rule Set: Within a storage policy, individual rules are organized into collections of rules, or rule sets. Typically, the rule sets can be in one of the following categories: rules for host-based services and datastore-specific rules.

Each rule set must include placement rules that describe requirements for virtual machine storage resources. All placement rules within a single rule set represent a single storage entity. These rules can be based on storage capabilities or tags. In addition, the datastore-specific rule set can include optional rules or storage policy components that describe data services to provide for the virtual machine. Generally, these rules request such services as caching, replication, other services provided by storage systems. To define the storage policy, one datastore-specific set is required. Additional rule sets are optional. A single policy can use multiple sets of rules to define alternative storage placement parameters, often from several storage providers.

Creating and managing VM storage policies pp 249

To create and manage storage policies for your virtual machines, you use the VM Storage Policies interface. Depending on whether you use the vSphere Web Client or the vSphere Client, the appearance of the VM Storage Policy interface and its options might change.

Define a Storage Policy

Prerequisites

- Make sure that the VM Storage Policies interface is populated with information about storage entities and data services that are available in your storage environment.

- Define appropriate storage policy components.

- Required privileges: VM storage policies.Update and VM storage policies.View.

Procedure

1 Start VM Storage Policy Creation Process

To define a virtual machine storage policy, use the Create New VM Storage Policy wizard.

2 Define Common Rules for a VM Storage Policy

On the Common rules page, specify which data services to include in the VM storage policy. The data services are provided by software components that are installed on your ESXi hosts and vCenter Server. The VM storage policy that includes common rules activates specified data services for the virtual machine.

3 Create Storage-Specific Rules for a VM Storage Policy

Use the Rule Set page to define storage placement rules. If your storage provides additional data services, such as replication, use the page to specify which data services to include in the VM storage policy.

4 Finish VM Storage Policy Creation

You can review the list of datastores that are compatible with the VM storage policy and change any

storage policy settings.

What to do next: You can apply this storage policy to virtual machines. If you use object-based storage, such as vSAN and Virtual Volumes, you can designate this storage policy as the default.

Storage policy components pp 258

A VM storage policy can include one or several reusable and interchangeable building blocks, called storage policy components. Each component describes a particular data service to be provided for the virtual machine. You can define the policy components in advance and associate them with multiple VM storage policies.

You cannot assign the predefined component directly to a virtual machine or virtual disk. Instead, you must add the component to the VM storage policy, and assign the policy to the virtual machine. The component describes one type of service from one service provider. The services can vary depending on the providers that you use, but generally belong in one of the following categories: compression, caching, encryption, replication.

Example: Table 20-2 Storage Policy Components

Default storage policies pp 267

When you provision a virtual machine on a datastore, you must assign to the virtual machine a compatible VM storage policy. If you do not configure and explicitly assign the storage policy to the virtual machine, the system uses a default storage policy.

- VMware-Provided Default Storage Policy: The generic default storage policy that ESXi provides applies to all datastores and does not include rules specific to any storage type. In addition, ESXi offers the default storage policies for object-based datastores, vSAN or Virtual Volumes. These policies guarantee the optimum placement for the virtual machine objects within the object-based storage.

- User-Defined Default Storage Policies: You can create a VM storage policy that is compatible with vSAN or Virtual Volumes. You can then designate this policy as the default for vSAN and Virtual Volumes datastores. The user-defined default policy replaces the default storage policy that VMware provides. Each vSAN and Virtual Volumes datastore can have only one default policy at a time. However, you can create a single storage policy with multiple placement rule sets, so that it matches multiple vSAN and Virtual Volumes datastores. You can designate this policy as the default policy for all datastores. When the VM storage policy becomes the default policy for a datastore, you cannot delete the policy unless you disassociate it from the datastore.

IMPORTANT. You cannot change the storage policy if you clone an encrypted virtual machine. For information about cloning an encrypted virtual machine, see *vSphere Security*. (pp 25 vSphere Admin Guide)

Objective 7.4

Managing ESXi Host Security

VMware imposes constraints on several parameters, settings, and activities within vSphere to protect against unauthorized intrusion and malicious activities. If you need to loosen the constraints to meet your unique requirements, ensure that you have a trusted environment or take other security measures.

Common Security Measures

In Chapter 9, you learned that vSphere has built-in security features and that you can take additional steps to harden ESXi, such as configuring lockdown mode, certificate replacement, and smart card authentication for enhanced security. Some common, additional security measures are stated here.

- Limit access to the Direct Console User Interface (DCUI), the ESXi Shell, and SSH. If you allow access to these items, which have privileged access to certain ESXi components, ensure that only trusted users have access and that timeouts are set.
- Do not directly access ESXi hosts that are managed by vCenter Server. Although it may be possible to access the host via DCUI, SSH, ESXi Shell, API or VMware Host Client, you should not normally do so. Instead, use the vSphere Client (or vSphere Web Client) or API connected to vCenter Server to manage the ESXi. Host.
- Only use the DCUI for troubleshooting. Likewise, only use root access to the ESXi Shell for troubleshooting.
- When upgrading ESXi components, only use VMware sources. Although the host runs several third-party packages, VMware only supports upgrades to those packages from VMware sources. Check third-party vendor sites and the VMware knowledge base for security alerts.

NOTE: YOU SHOULD FOLLOW THE VMWARE SECURITY ADVISORIES AT HTTP://WWW.VMWARE.COM/SECURITY/.

Configure ESXi Using Host Profiles

In addition to using host profiles when implementing a new vSphere environment as discussed in Chapter 11, you can leverage host profiles for on-going vSphere security management. For example, you can configure key security settings in host profiles, attach host profiles to existing ESXi hosts, and check compliance.

Key settings to include in the host profile

You can consider any setting that is applied by a host profile to be important to ensuring that your hosts are secured. Some settings, like direct ESXi permissions, may be obvious. Other settings, like NTP settings may not be obvious, but time synchronization issues impact integration with Active Directory which impacts user authentication. Network settings, like physical NIC speed, could impact the ability of the host to connect to the proper management network.

Attach Host Profiles to ESXi Hosts

You can leverage host profiles to set up standard ESXi host configuration settings and automate compliance to those settings. As covered in Chapter 11, host profiles can be used to apply many host configuration settings, including security measures, such as ESXI level permissions. You can use the vSphere Client to configure a host profils for a reference host and apply the host profile to a set of hosts. You can also use host profiles to monitor hosts for host configuration changes. You can attach the host profile to a cluster to apply it to all hosts in the cluster. The high level steps are:

1. Set up the reference host to specification and create a host profile.
2. Attach the profile to a host or cluster.
3. Apply the host profile of the reference host to other hosts or clusters

Check Compliance

To ensure that an ESXi host is properly configured according to your standards, you can check for its compliance to its attached host profile. You can use the results to identify non-compliant settings on the host and remediate with the host profiles settings. You can use these steps to check compliance.

1. Navigate to Host Profiles main view.
2. Right click a host profile.
3. Click **Check Host Profile Compliance**

The compliance status is for each ESXi host is `Compliant`, `Unknown`, or `Non-compliant`. `Non-compliant` status indicates specific inconsistency between the profile and the host, which you should remediate. `Unknown` status indicates that the compliance of the host is not known, because it could not be verified. A common root cause is that the host is disconnected. You should resolve the issue and re-check compliance.

Use Scripts to Manage Host Configuration Settings

Another means to establish a standard, secured configuration for ESXi hosts in your vSphere environment is to use scripts. In environments with many hosts, managing hosts with scripts is faster and less error prone than managing the hosts from the vSphere Client. You can use vSphere PowerCLI, vSphere Command Line Interface (vCLI), and ESXCLI commands.

You can install the vSphere CLI command set on a supported Linux or Windows system and then use it to manage ESXi hosts. vCLI host management commands from earlier versions have been replaced with commands that have equivalent functionality. vCLI 4.1 commands that begin with “vicfg-“ have been replaced with ESXCLI commands. For example, the legacy command `vicfg-advcfg` which performs advanced configuration has been replaced with `esxcli system settings advanced`.

Generally speaking, you use ESXCLI commands to interact with a single, specific ESXi host, either by directly connecting to the host or by specifying the host using the `--vihost` parameter when connecting to a vCenter Server. You should not expect to use ESXCLI to perform operations that require vCenter Server or multiple ESXi hosts, such as vMotion operations.

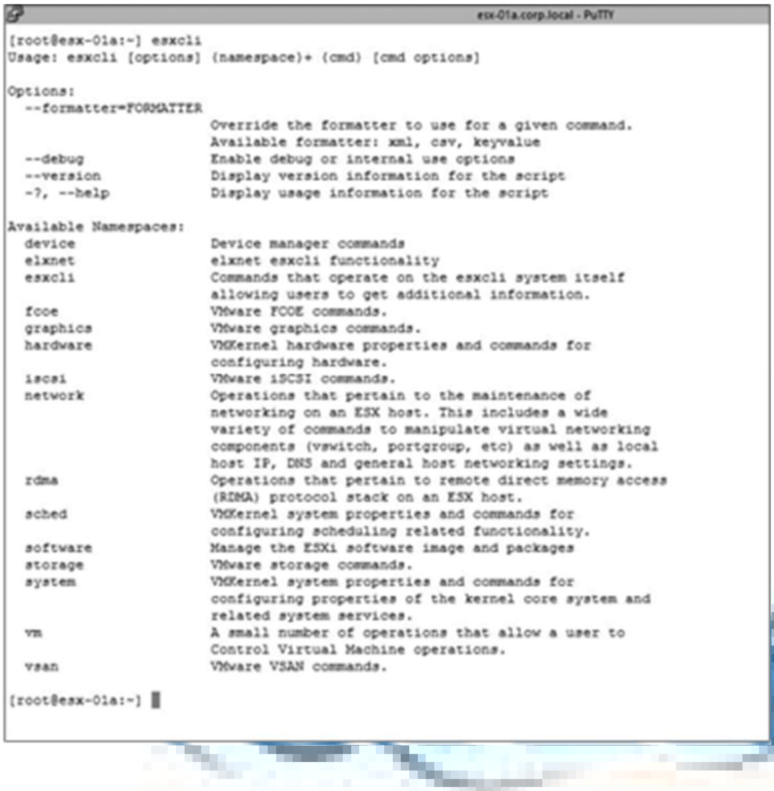
In a vSphere environment, commands can be run directly on an ESXi host in the ESXi Shell or via Secure Shell (SSH). Commands can also be run in the vSphere Command Line Interface (vCLI). VMware provides a Windows version and a Linux version of the vCLI to allow you to install it on the desktop of your choice. Running commands in the ESXi Shell is mostly useful for troubleshooting, especially when tackling issues that relate to ESXi host connectivity.

The VMware recommended command line approach is to use the ESXCLI command name space in the vCLI. The ESXi Shell provides ESXCLI as well as the legacy **esxcfg** and **vicfg** command sets. Generally speaking, you should use ESXCLI commands when available and only use **vicfg** and **esxcfg** commands when required. For example, when using a command to troubleshoot NTP configuration, you should use **vicfg-ntp**, because ESXCLI has no command for configuring NTP.

The ESXCLI command set is intended to provide a single set of commands to perform all ESXi host-based administrative tasks. It provides a collection of namespaces as a mechanism for an administrator to quickly discover the precise command necessary for a specific task. For example, all the commands to configure networking exist in the **esxcli network** namespace, and all the commands to configure storage exist in the **esxcli storage** namespace. Each namespace is further divided into child namespaces that comprise various functions performed under the parent namespace. For example, the **esxcli storage** parent namespace contains a **core** namespace that deals with storage adapters and devices and a **nmp** namespace that deals with path selection and storage array types. Therefore, a typical ESXCLI command is composed of multiple namespaces, where each additional namespace is used to narrow the scope of the command, ending with the actual operation to be performed.

You can use the following method to identify the proper ESXCLI command to perform a specific task. First, simply enter `esxcli` at the command prompt in the ESXi Shell or vCLI instance. Because it is not a command by itself, just the entry point to the namespace hierarchy, the results will show the first level of the namespace hierarchy. The first level of available namespaces includes `esxcli`, `fcoe`, `hardware`, `iscsi`, `network`, `sched`, `software`, `storage`, `system`, and `vm`. The results include a brief description of each namespace as shown in Figure 17-5. Next, identify which namespace is most likely to serve your need. Press the up-arrow key on the keyboard to retrieve the last entered namespace and add the name for the next namespace. For example, if you are seeking a network-related command, you could enter **esxcli network**, which provides the next level of the namespace hierarchy, as shown in Figure 17-6.

Figure 17-5 First-level hierarchy of the ESXCLI namespace.



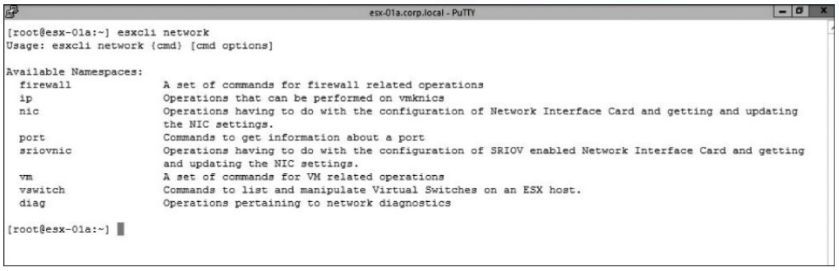
```
[root@esx-01a:~] esxcli
Usage: esxcli [options] {namespace}+ {cmd} [cmd options]

Options:
  --formatter=FORMATTER
                        Override the formatter to use for a given command.
                        Available formatter: xml, csv, keyvalue
  --debug               Enable debug or internal use options
  --version             Display version information for the script
  -?, --help            Display usage information for the script

Available Namespaces:
  device                Device manager commands
  elxnet                elxnet esxcli functionality
  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.
  rdma                  Operations that pertain to remote direct memory access
                        (RDMA) protocol stack on an ESX host.
  sched                 VMKernel system properties and commands for
                        configuring scheduling related functionality.
  software              Manage the ESXi software image and packages
  storage               VMware storage commands.
  system                VMKernel system properties and commands for
                        configuring properties of the kernel core system and
                        related system services.
  vm                    A small number of operations that allow a user to
                        Control Virtual Machine operations.
  vsan                  VMware VSAN commands.

[root@esx-01a:~] █
```

Figure 17-6 Available Namespaces at **esxcli network**.



```
[root@esx-01a:~] esxcli network
Usage: esxcli network {cmd} [cmd options]

Available Namespaces:
  firewall              A set of commands for firewall related operations
  ip                    Operations that can be performed on vmknics
  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
  vswitch               Commands to list and manipulate Virtual Switches on an ESX host.
  diag                  Operations pertaining to network diagnostics

[root@esx-01a:~] █
```

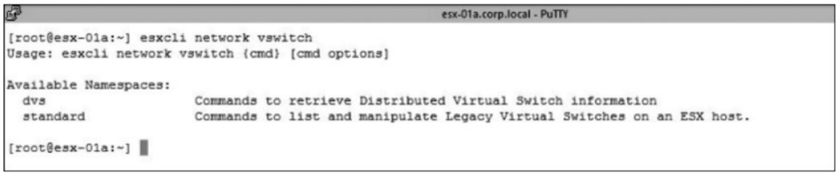
For a more thorough example, if you are seeking a command to list all standard vSwitches, you could use these steps:

Step 1. Enter **esxcli**, and examine the results shown in Figure 17-5.

Step 2. Enter **esxcli network**, and examine the results shown in Figure 17-6.

Step 3. Enter **esxcli network vswitch**, and examine the results shown in Figure 17-7.

Figure 17-7 Available Namespaces at **esxcli network vswitch** .



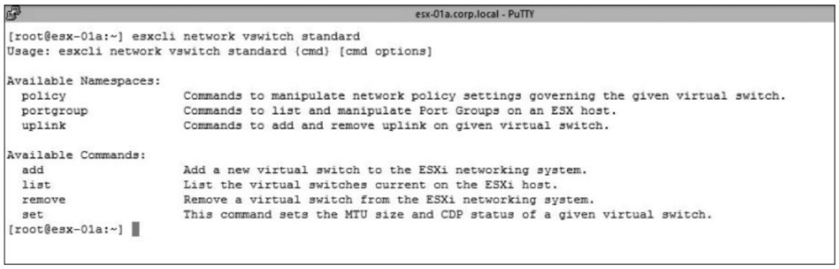
```
[root@esx-01a:~] esxcli network vswitch
Usage: esxcli network vswitch {cmd} [cmd options]

Available Namespaces:
  dvs                   Commands to retrieve Distributed Virtual Switch information
  standard              Commands to list and manipulate Legacy Virtual Switches on an ESX host.

[root@esx-01a:~]
```

Enter **esxcli network vswitch standard**, and examine the results shown in Figure 17-8. Notice that at this level, some Available Commands are now displayed. These commands are **add**, **list**, **remove**, and **set**. For this example, the list command seems to be the most appropriate.

Figure 17-8 Available Namespaces at **esxcli network vswitch standard**



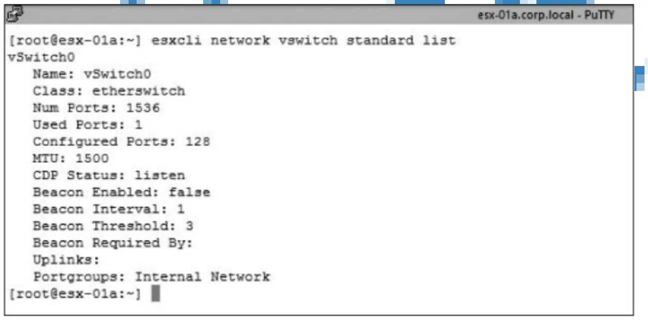
```
[root@esx-01a:~] esxcli network vswitch standard
Usage: esxcli network vswitch standard {cmd} [cmd options]

Available Namespaces:
  policy                Commands to manipulate network policy settings governing the given virtual switch.
  portgroup             Commands to list and manipulate Port Groups on an ESX host.
  uplink                Commands to add and remove uplink on given virtual switch.

Available Commands:
  add                   Add a new virtual switch to the ESXi networking system.
  list                  List the virtual switches current on the ESXi host.
  remove                Remove a virtual switch from the ESXi networking system.
  set                   This command sets the MTU size and CDP status of a given virtual switch.
[root@esx-01a:~]
```

Enter the **esxcli network vswitch standard list** command, which executes the command and produces results, as shown in Figure 17-9.

Figure 17-9 Results of Command to List Standard Virtual Switches



```
[root@esx-01a:~] esxcli network vswitch standard list
vSwitch0
   Name: vSwitch0
   Class: etherswitch
   Num Ports: 1536
   Used Ports: 1
   Configured Ports: 128
   MTU: 1500
   CDP Status: listen
   Beacon Enabled: false
   Beacon Interval: 1
   Beacon Threshold: 3
   Beacon Required By:
   Uplinks:
   Portgroups: Internal Network
[root@esx-01a:~]
```

NOTE: When using the previously discussed approach to discover the appropriate command for a given task, pay attention to commands versus namespaces. Entering a namespace at the command prompt is always safe because it will simply display the next level of available namespaces and commands. However, entering a command at the command prompt will execute that command. You should be careful not to enter a command without fully understanding the consequences. Table 17-5 contains some sample ESXCLI commands.

Table 17-5 Sample ESXCLI commands

Commands	Purpose and Details
`esxcli system account add`	Create a ESXi host local user account
`esxcli system account set`	Configure an ESXi host local user account

`esxcli system account list`	List ESXi host local user accounts
`esxcli system account remove`	Delete an ESXi host local user accounts
`esxcli network ip dns server list`	List DNS servers.
`esxcli network nic list`	List the ESXi host physical network adapters

Likewise, you can use PowerCLI to manage, configure, and operate a vSphere environment. Generally speaking, when connecting to a vCenter Server environment, the functionality scope of PowerCLI is similar to the functionality scope of using the vSphere Client to logon to vCenter Server. Table 17-6 describes a few popular PowerCLI commands.

Table 17-6 Sample PoweCLI commands

Command	Purpose	Example
Connect-VIServer	Connect to a vCenter Server	`Connect-VIServer vc01  -User administrator@vsphere.local` Connects to a vCener Server named vc01 as user `administrator@vsphere.local`
Get-VMHost	Retrieve information about one or more ESXi hosts	`Get-VMHost -Location MyDC` Retrieves details about all ESXi hosts in a datacenter named `MyDC`.
New-Snapshot	Create a virtual machine snapshot	`Get-ResourcePool MyRP | Get-VM | New-Snapshot -Name Snap01` For each virtual machine discovered in a resource pool named MyRP, create a snapshot called Snap01

ESXi Passwords and Account Lockout

For direct ESXi host access, you can use local user accounts. You can use the root account and you can create additional local user accounts. When setting a password on these accounts, you must comply with or modify the predefined requirements. ESXi uses the Linux PAM module `pam_passwdqc` for password management and control. You can change the required length, change character class requirement, and allow pass phrases using the `Security.PasswordQualityControl` advanced option.

> **Note** The default requirements for ESXi passwords can change from one release to the next. You can check and change the default password restrictions using the `Security.PasswordQualityControl` advanced option.

One step to harden an ESXi host is to harden the password required to use its predefined, local administrator account, which is called `root`. By default, the ESXi host enforces passwords for its local user accounts, which may be used to access the host via the Direct Console User Interface (DCUI), the ESXi Shell, Secure Shell (SSH) or the vSphere Client. Starting with ESXi 6.0, the default password policy must contain characters from at least three character classes (of the four character classes, which are lowercase letters, uppercase letters, numbers and special characters) and must be at least seven characters long.

> **Note** An uppercase character that begins a password and a number that ends a password do not count toward the number of used character classes. The password cannot contain a dictionary word or part of a dictionary word.

For example, `xQaT3!A` is a an acceptable password, because it contains 4 character classes and 7 characters. But, `Xqate!3` is not an acceptable password, because it only contains two character classes as the leading `X` and ending `3` do not count toward the number of used character classes. You can modify the ESXi password requirements using the ESXi host `Security.PasswordQualityControl` advanced option. You can also set `Security.PasswordQualityControl` configure the ESXi host to accept pass phrases, which it does not accept by default. The key to changing the password and pass phrase requirements is understanding the syntax and functionality of the `Security.PasswordQualityControl` parameter, whose default value is

```
retry=3 min=disabled,disabled,disabled,7,7
```

The first part of the value used for this parameter identifies the number of retries allowed for the user following a failed attempt to logon. In the default value, `retry=3` indicates that three additional attempts are permitted following a failed logon. The remainder of the value can be abstracted as

`min=`*`N0`*`,`*`N1`*`,`*`N2`*`,`*`N3`*`,`*`N4`*

where:

- *`N0`* is the minimum number of accepted characters for passwords that contain characters from only one class or `disabled` to disallow passwords that contain characters from only one class.
- *`N1`* is the minimum number of accepted characters for passwords that contain characters from only two classes or `disabled` to disallow passwords that contain characters from only two classes.
- *`N2`* is the minimum number of accepted characters for pass phrases or `disabled` to disallow passphrases. Additionally, to require a passphrase, append `passphrase=`*`N`* to the end of the value, where *`N`* specifies the minimum number of words, separated by spaces, in the passphrase.

- *N3* is the minimum number of accepted characters for passwords that contain characters from only three classes or `disabled` to disallow passwords that contain characters from only three classes.
- *N4* is the minimum number of accepted characters for passwords that contain characters from all four classes.

For example, to require a passphrase with a minimum of 16 characters and 3 words, set the `Security.PasswordQualityControl` to

```
retry=3 min=disabled,disabled,16,7,7,passphrase=3
```

The password requirements in ESXi 6.0 are implemented by `pam_passwdqc`. For more details, see the man pages for `pam_passwdqc`.

In vSphere 6.x, account locking is supported for access through SSH and through the vSphere Web Services SDK. The Direct Console Interface (DCUI) and the ESXi Shell do not support account lockout. By default, a maximum of ten failed attempts is allowed before the account is locked. The account is unlocked after two minutes by default. You can modify the lockout behavior using the host's advanced options:

- `Security.AccountLockFailures`: Maximum number of failed login attempts before a user's account is locked. Zero disables account locking.
- `Security.AccountUnlockTime`: Number of seconds that a user is locked out.

SSH and ESXi Shell Security

You can use SSH to remotely log in to the ESXi Shell and perform troubleshooting tasks for the host. SSH configuration in ESXi is enhanced to provide a high security level. VMware does not support Version 1 SSH protocol and uses Version 2 protocol exclusively. SSH supports only 256-bit and 128-bit AES ciphers for your connections.

The ESXi Shell is disabled by default on ESXi hosts. If necessary, you can enable local and remote access to the shell. But, to reduce the risk of unauthorized access, you should only enable the ESXi Shell when troubleshooting If the ESXi Shell or SSH is enabled and the host is placed in lockdown mode, accounts on the Exception Users list who have administrator privileges can use these services. For all other users, ESXi Shell or SSH access is disabled. Starting with vSphere 6.0, ESXi or SSH sessions for users who do not have administrator privileges are closed.

If the ESXi Shell is enabled, you can still log in to it locally, even if the host is running in lockdown mode. To enable local ESXi Shell access, enable the ESXi Shell service. To enable remote ESXi Shell access, enable the SSH service.

Note: The root user and users with the Administrator role can access the ESXi Shell. Users who are in the Active Directory group ESX Admins are automatically assigned the Administrator role. By default, only the root user can run system commands (such as `vmware -v`) by using the ESXi Shell

You can use the vSphere Web Client to enable local and remote access to the ESXi Shell.

1. Browse to the host in the vSphere Web Client inventory.
2. Click **Configure**.
3. Under System, select **Security Profile**.
4. In the Services panel, click **Edit**.
5. Select a service from the list.
   a. ESXi Shell
   b. SSH
   c. Direct Console UI
6. Optionally, click **Service Details** and select the startup policy **Start and stop manually** or **Start and stop with host**
7. Select **Start** to enable the service.
8. Click **OK**

To increase security, you can set ESXi Shell Timeout. The availability timeout setting is the amount of time that can elapse before you must log in after the ESXi Shell is enabled. After the timeout period, the service is disabled, and users are not allowed to log in. To set the timeout, you can use this procedure.

1. Browse to the host in the vSphere Web Client inventory.
2. Click **Configure**.
3. Under System, select **Advanced System Settings**.
4. Select **UserVars.ESXiShellTimeOut** and click **Edit**.
5. Enter the idle timeout setting.
6. You must restart the SSH service and the ESXi Shell service for the timeout to take effect.
7. Click **OK**

Likewise, you can set a timeout for idle ESXi Shell sessions. The idle timeout is the amount of time that can elapse before a user is logged out of an idle interactive session. You can control the amount of time for both local and remote (SSH) session from the Direct Console Interface (DCUI) or from the vSphere Web Client using the following procedure.

1. Browse to the host in the vSphere Web Client inventory.
2. Click **Configure**.
3. Under System, select **Advanced System Settings**.
4. Select **UserVars.ESXiShellInteractiveTimeOut**, click the **Edit** icon, and enter the timeout setting.
5. Restart the ESXi Shell service and the SSH service for the timeout to take effect.
6. If the session is idle, users are logged out after the timeout period elapses.

PCI and PCIe Devices and ESXi

You can use the VMware DirectPath I/O feature to pass through a PCI or PCIe device to a virtual machine, but this results in a potential security vulnerability. This could be triggered when buggy or malicious code, such as a device driver, is running in privileged mode in the guest OS. So, you should use PCI or PCIe passthrough to a virtual machine *only* if a trusted entity owns and administers the virtual machine. Otherwise, you risk that the host may be compromised by the following:

- The guest OS might generate an unrecoverable PCI or PCIe error
- The guest OS might generate a Direct Memory Access (DMA) operation that causes an IOMMU page fault on the ESXi host.
- If the operating system on the ESXi host is not using interrupt remapping, the guest OS might inject a spurious interrupt into the ESXi host on any vector

Disable the Managed Object Browser

The managed object browser (MOB) provides you with a means to explore the VMkernel object model. Starting with vSphere 6.0 the MOB is disabled by default to avoid malicious configuration changes or actions. You can enable and disable the MOB manually. VMware recommends that you do not enable MOB in production systems.

To enable the MOB using the vSphere Client, you can use the following procedure.

1. In the vSphere Client, select the host in the inventory.
2. In the right pane, click the **Configuration** tab.
3. Under Software, select **Advanced Settings**.
4. From the left pane of the Advanced Settings dialog box, select **Config > HostAgent > plugins > solo**.
5. Select or deselect **Config.HostAgent.plugins.solo.enableMob** to enable or disable the Managed Object Browser.

ESXi Network Security

You should connect the ESXi host to several networks and isolate traffic for each purpose. For example, ensure that management, vMotion, virtual machine, and iSCSI traffic travel over separate networks. Virtual infrastructure traffic, such as storage, vMotion, and FT typically does not require routing to networks outside a single physical server rack.

Using a management network allows you to isolate client traffic, command-line interface (CLI), API traffic, and third- party software traffic from other traffic. This network should be accessible only by system, network, and security administrators. Consider using a jump box or virtual private network (VPN) to secure access to the management network.

Modifying ESXi Web Proxy Settings

You should consider the encryption options and user security guidelines when modifying Web proxy settings.

- Do not set up certificates that use a password or pass phrases. ESXi does not support Web proxies that use passwords or pass phrases.
- To support encryption for user names, passwords, and packets, SSL is enabled by default for vSphere Web Services SDK connections. To change this, you could disable SSL for your vSphere Web Services SDK connection by switching the connection from HTTPS to HTTP.

vSphere Auto Deploy Security Considerations

When using vSphere Auto Deploy, you should pay careful attention to networking security, boot image security, and potential password exposure through host profiles to protect your environment.

Auto Deploy transfers data over SSL, but the authenticity of the client or of the Auto Deploy server is not checked during a PXE boot. You should consider completely isolating the network where Auto Deploy is used.

The boot image used by vSphere Auto Deploy contains VMware Information Bundle (VIB) packages and may contain host profiles and host customizations. The administrator (root) password and user passwords that are included with host profile and host customization are MD5 encrypted. Any other password associated with profiles are not encrypted. Consider using the vSphere Authentication Proxy to avoid exposing the Active Directory passwords.

ESXi public and private SSL key and certificate are included in the boot image.

Control Access for CIM-Based Hardware Monitoring Tools

The Common Information Model (CIM) system provides an interface that enables hardware-level management from remote applications. To ensure it is secure, provide only the minimum access necessary to these remote applications. Avoid using a root or Administrator account for this purpose because if the remote application is compromised, the virtual environment can be compromised.

ESXI Certificate Management

Initially, in vSphere 6.x, ESXi hosts boot with an autogenerated certificate. When the host is added to a vCenter Server system, it is provisioned with a certificate signed by the VMware Certificate Authority (VMCA). You can view and manage ESXi certificates using the vSphere Web Client or the `vim.CertificateManager` API in the vSphere Web Services SDK. You cannot view or manage ESXi certificates by using certificate management CLIs that are available for managing vCenter Server certificates.

By default, in vSphere 6.x, ESXi hosts are provisioned with certificates by VMCA. You can instead use custom certificate mode or the legacy thumbprint mode. In most cases, mode switches are disruptive and not necessary. If you do require a mode switch, review the potential impact before you start. You should only use the thumbprint mode for debugging.

To perform certificate management for ESXi, you must have the `Certificates.Manage Certificates` privilege

Change the Certificate Mode

If necessary, you can change the vCenter Server certification mode from VMCA to custom CA mode or thumbprint mode. For example, if you wish to use custom certificates or instead of VMCA to provision ESXi hosts, you need to edit the vCenter Server `vpxd.certmgmt.mode` advanced option. From the vSphere client you can use this procedure to change the certificate mode.

1. Select the vCenter Server and click **Configure**.
2. Click **Advanced Settings**, and click **Edit**.
3. In the Filter box, enter `certmgmt` to display only certificate management keys.
4. Change the value of `vpxd.certmgmt.mode` to `custom` and click **OK**.
5. Restart the vCenter Server service

Using Custom ESXi Certificates

You can switch the certificate mode from VMCA to a different root CA using these steps.

1. Obtain the certificates from the trusted CA..
2. Place the host or hosts into maintenance mode and disconnect them from vCenter Server.
3. Add the custom CA's root certificate to VECS.
4. Deploy the custom CA certificates to each host and restart services on that host.
5. Change the Certificate Mode to Custom CA mode. (See the Change the Certificate Mode procedure)
6. Connect the host or hosts to the vCenter Server system.

Switch Back to VMCA Mode

If you are using the custom CA mode, you can switch back to VMCA mode using this procedure.

1. Remove all hosts from the vCenter Server system.
2. On the vCenter Server system, remove the third-party CA's root certificate from VECS.
3. Change the Certificate Mode to VMCA mode. (See the Change the Certificate Mode procedure)
4. Add the hosts to the vCenter Server system.

Certificate Expiration

In vSpere 6.x, you can view information, including certificate expiration, for all certificates that are signed by VMCA or a third party CA in the vSphere Web Client. A yellow alarm is raised if the certificate is expiring shortly (less than 8 months). A red alarm is raised if the certificate is in the Expiration Imminent state (less than two months).

Customizing ESXi Hosts with Security Profile

ESXi Firewall Configuration

By default, the ESXi firewall is configured to block incoming and outgoing traffic, except traffic for services that are enabled in the hosts' security profile. Prior to opening any ports on the firewall, you should consider the impact it may have for potential attacks and unauthorized user access. You can reduce this risk by configuring the firewall to only allow communication on the port with authorized networks. To modify the firewall's rule set, you can use the vSphere Web Client to modify the host's security profile using this procedure.

Step 1. Select the ESXi host in the inventory, click **Manage** > **Settings**.

Step 2. Click **Security Profile**.

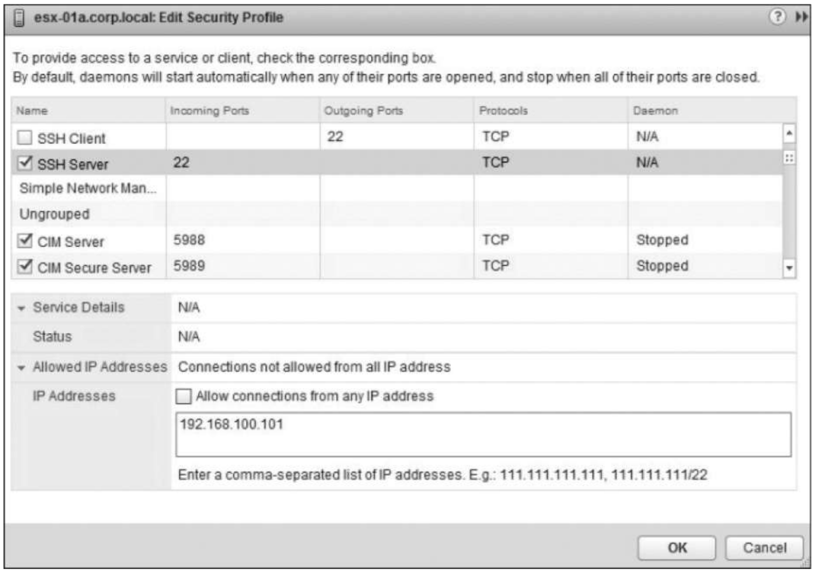
Step 3. In the Firewall section, click **Edit**.

Step 4. Examine the rule set. Change the state of any rule by selecting the rule (place a check in the rule's box) to enable the rule or de-select the rule to disable.

Step 5. Optionally, un-check the **Allow connections from any IP address box** and enter specific IP addresses in the accompanying text box to restrict use to only those IP addresses as illustrated in Figure 17-10.

Step 6. Click **OK**.

Figure 17-10 Restrict IP Addresses for a Firewall Rule.



esx-01a.corp.local: Edit Security Profile

To provide access to a service or client, check the corresponding box.
By default, daemons will start automatically when any of their ports are opened, and stop when all of their ports are closed.

Name	Incoming Ports	Outgoing Ports	Protocols	Daemon
□ SSH Client		22	TCP	N/A
☑ SSH Server	22		TCP	N/A
Simple Network Man...				
Ungrouped				
☑ CIM Server	5988		TCP	Stopped
☑ CIM Secure Server	5989		TCP	Stopped

Service Details	N/A
Status	N/A
Allowed IP Addresses	Connections not allowed from all IP address
IP Addresses	□ Allow connections from any IP address 192.168.100.101 Enter a comma-separated list of IP addresses. E.g.: 111.111.111.111, 111.111.111/22

Note: The firewall also allows Internet Control Message Protocol (ICMP) pings and communication with DHCP and DNS (UDP only) clients.

Note: The behavior of the NFS Client rule set (nfsClient) is different from other rule sets. When the NFS Client rule set is enabled, all outbound TCP ports are open for the destination hosts in the list of allowed IP addresses.

The ESXi software firewall is enabled by default. It should never be disabled while running production virtual machines. In rare cases, such as a temporary troubleshooting measure, you can disable the ESXi firewall using the `esxcli network firewall set --enabled false` command.

Customizing ESXi Services

Several optional services that are provided in an ESXi host are disabled by default. VMware disables these services to provide strong security out of the box. In a default installation, you can modify the status of the following services from the vSphere Web Client.

Running Services: Direct Console UI, Load-Based Teaming Daemon, CIM Server, VMware vCenter Agent

Stopped Services: ESXi Shell, SSH, Active Directory Service, NTP Daemon, PC/SC Smart Card Daemon, SNMP Server, Syslog Server, X.Org Server

In some circumstances, you may wish to configure and enable these services. A good example of an optional service that you may decide to configure and enable in most environments is NTP, because solid time synchronization is vital for many services. For another example, you may wish to temporarily enable Secure Shell (SSH) service while troubleshooting. To enable, disable, and configure these services, you can edit the host's security profile in the vSphere Web Client.

Step 1. Select the ESXi host in the inventory, click **Manage** > **Settings**.

Step 2. Click **Security Profile**.

Step 3. In the Services section, click **Edit**.

Step 4. Examine the set of services. Select a service that you wish to modify and use its Service Details pane to make changes. Use the **Start**, **Stop** or **Restart** buttons to immediately change the state of the service

Step 5. Use the **Startup Policy** menu to change the status permanently. You can choose from the following options:

- **Start automatically if any ports are open and stop when all ports are closed:** This is the default setting for the defined services. If at least one of the associated ports are open, then the service starts, but may fail if at least one associated port is closed. If all associated ports are closed, then the service stops.
- **Start and stop with host.** The service starts automatically each time the host starts. The service stops automatically each time a host shutdown begins.
- **Start and stop manaually.** The service does not start automatically when the host starts or when ports are opened. It does not stop automatically when ports are closed. But, it does shutdown automatically each time a host shutdown begins.

Step 6. Click **OK**.

Manage the Acceptance Levels of Hosts and VIBs

VIBs are software packages that include a signature from VMware or a VMware partner. To protect the integrity of the ESXi host, do not allow users to install unsigned (community-supported) VIBs. An unsigned VIB contains code that is not certified by, accepted by, or supported by VMware or its partners. Community-supported VIBs do not have a digital signature. The host's acceptance level must be the same or less restrictive than the acceptance level of any VIB you want to add to the host. For example, if the host acceptance level is `VMwareAccepted`, you cannot install VIBs at the `PartnerSupported` level. You should use extreme caution when allowing `CommunitySupported` VIBs. The following list contains details on defined VIB acceptance levels.

- **VMwareCertified**: VIBs go through thorough testing equivalent to VMware in-house Quality Assurance testing. for the same technology. Only I/O Vendor Program (IOVP) program drivers are published at this level. VMware takes support calls for VIBs with this acceptance level.

- **VMwareAccepted**: VIBs go through testing that is run by a partner and verified by VMware. CIM providers and PSA plug-ins are among the VIBs published at this level. VMware directs support calls for VIBs with this acceptance level to the partner's support organization.

- **PartnerSupported**: VIBs that are published by a partner that VMware trusts. The partner performs all testing, but VMware does not verify. VMware directs support calls for VIBs with this acceptance level to the partner's support organization.

- **CommunitySupported**: VIBs that have not gone through any VMware-approved testing program and are not supported by VMware Technical Support or by a VMware partner.

Other Security Management

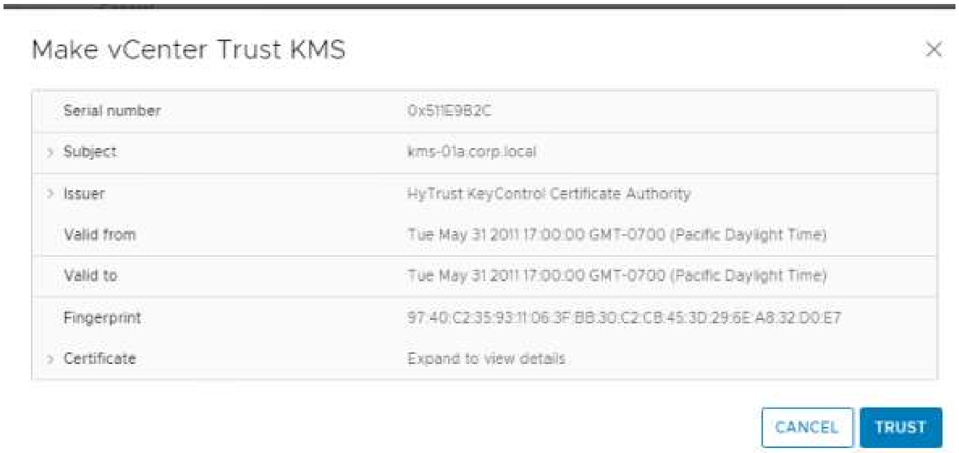
Managing vSphere security can involve other tasks, such as those described here.

Key Management Server

In order to use encryption in vSphere, you must be running a Key Management Server (KMS) that has a trust relationship with vCenter Server. To add a KMS server to vCenter Server, you can use the following procedure.

- In the vShere Client, select the vCenter Server in the list of vCenter Servers
- Select the Configuration tab
- Click on **Key Managent Servers**
- Click **ADD**
- Provide the server name, server address (FQDN), and server port.
- Optionally, provide other appropriate details, such as proxy details and user credentials
- If you are adding the first KMS server in a cluster, provide a cluster name
- Click the radius button next to the KMS server name
- In the Make vCenter Trust KMS window, click **TRUST** (see Figure 17-11)
- Click **MAKE KMS TRUST VCENTER** (see Figure 17-12)
- Select **KMS Certificate and private key**, and click **Next**.
- In the next window, next to KMS Certificate, click **Upload File** and open an available certificate PEM file.
- In the same window, next to KMS Private Key, click **Upload File** and open an available certificate PEM file.
- Click the **ESTABLISH TRUST** button

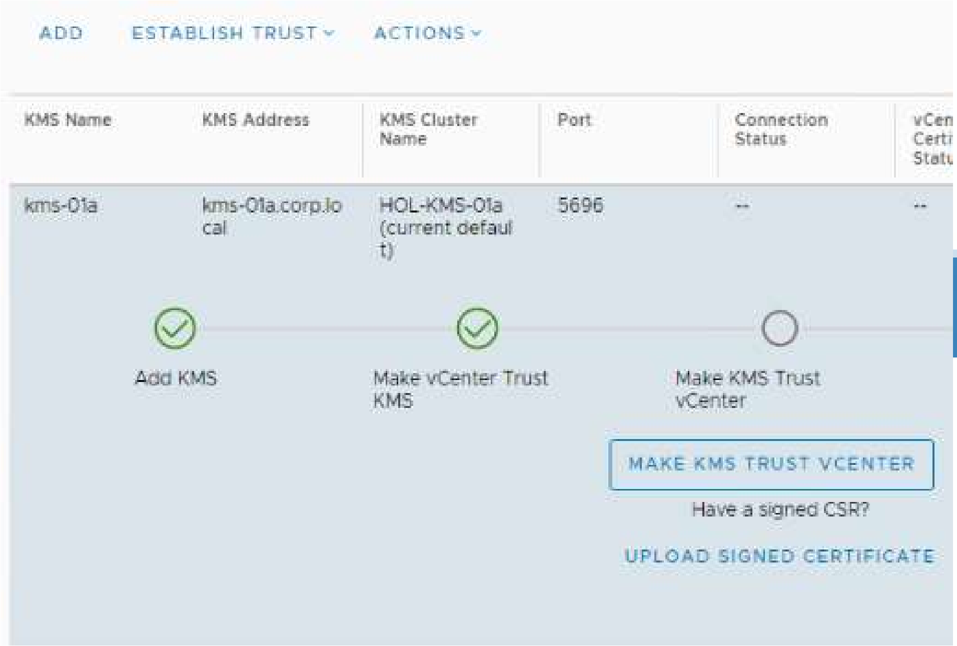
Figure 17-11 Make vCenter Trust KMS



Make vCenter Trust KMS

Serial number	0x511E9B2C
› Subject	kms-01a.corp.local
› Issuer	HyTrust KeyControl Certificate Authority
Valid from	Tue May 31 2011 17:00:00 GMT-0700 (Pacific Daylight Time)
Valid to	Tue May 31 2011 17:00:00 GMT-0700 (Pacific Daylight Time)
Fingerprint	97:40:C2:35:93:11:06:3F:BB:30:C2:CB:45:3D:29:6E:A8:32:D0:E7
› Certificate	Expand to view details

Figure 17-12 Make KMS Trust vCenter

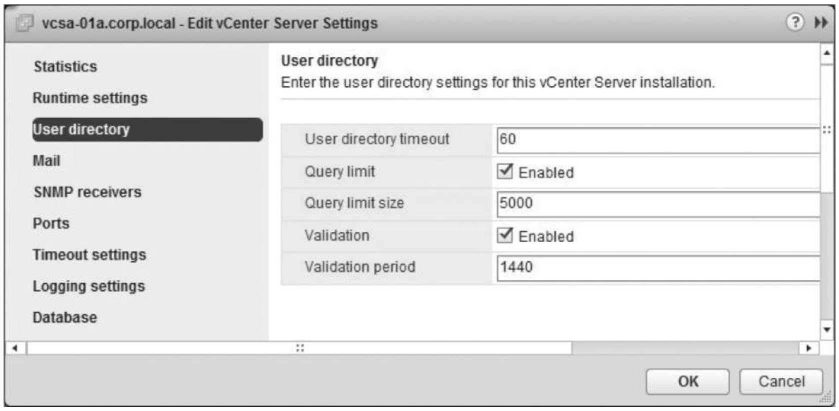


KMS Name	KMS Address	KMS Cluster Name	Port	Connection Status	vCen Certi Statu
kms-01a	kms-01a.corp.local	HOL-KMS-01a (current default)	5696	--	--

Change Permission Validation Settings

Periodically, vCenter Server validates its user and group lists against the users and groups in the Windows Active Directory domain. It removes users and groups that no longer exist in the domain. You can change the behavior of this validation by using the vSphere Web Client edit the general settings of the vCenter Server and change the **Validation** and **Validation Period** options. If you want to want to disable the validation, deselect the **Validation > Enable** checkbox, shown in Figure 17-13. If you want to adjust the frequency in which this validation is performed, enter a value in the **Validation Period** text box to specify a time, in minutes, between validations.

Figure 17-13 User Directory Validation Setting



vcsa-01a.corp.local - Edit vCenter Server Settings

User directory

Enter the user directory settings for this vCenter Server installation.

User directory timeout	60
Query limit	☑ Enabled
Query limit size	5000
Validation	☑ Enabled
Validation period	1440

Lockdown Mode

NOTE: If you upgrade a host that is in lockdown mode to ESXi version 6.0 without exiting lockdown mode, and if you exit lockdown mode after the upgrade, all permissions defined before the host entered lockdown mode are lost. The system assigns the administrator role to all users who are found in the DCUI.Access advanced option to guarantee that the host remains accessible.

ESXi 6.x provides different levels of lockdown and introduces the `Exception Users` list. In strict lockdown, the DCUI is disabled and no one can use it. In normal lockdown, some users can use the DCUI. `Exception Users`, who are administrators and users who are identified in the `DCUI.Access` advanced system setting can access the DCUI.

`Exception Users`, who have administrator privileges, can use the ESXi Shell or SSH, if these services are enabled, even if the host is placed in strict or normal lockdown mode. For all other users, ESXi Shell or SSH access is disallowed when a host is in strict or normal lockdown mode. Starting with vSphere 6.0, ESXi or SSH sessions for users who do not have administrator privileges are terminated when lockdown mode is enabled

To enable lockdown mode, use the vSphere Web Client and follow this procedure.

Step 1. In the inventory pane, select the ESXi host.

Step 2. In the middle pane, select **Manage** > **Settings** > **Security Profile**.

Step 3. In the Lockdown Mode panel, click the **Edit** button.

Step 4. Click **Lockdown Mode** and choose either **Normal** or **Strict**.

Step 5. Click **OK**.

To disable lockdown mode, repeat the procedure, but choose **Disabled** for the lockdown mode.

NOTE: In vSphere 5.0 and earlier, only the `root` account can log into the DCUI on an ESXi host that is in lockdown mode.

In vSphere 5.1 and later, you can add a user to the DCUI.Access advanced system setting to grant the user access to the DCUI on a host that is in lockdown mode, even if the user is not granted the Administrator role on the host. The main purpose of this feature is to prepare for catastrophic failures of vCenter Server.

In vSphere 6.0, also includes an Exception Users list, whose main purpose is to support the use of lockdown mode, but still support service accounts, which must logon directly to the ESXi host. User accounts in the Exception Users list, who have administrator privileges can logon to the DCUI and ESXi Shell.

By default, the root account is included in DCUI.Access. You could consider removing the root account from DCUI.Access and replacing it with another account for better auditability.

Table 17-7 provides details for the behavior of an ESXi host in lockdown mode.

Table 17-7 ESXi Lockdown Mode Behavior

Service	Normal Mode	Normal Lockdown Mode	Strict Lockdown Mode
vSphere Web Services API	All users, based on permissions	vCenter (vpxuser) Exception users, based on permissions vCloud Director (vslauser, if available)	vCenter (vpxuser) Exception users, based on permissions vCloud Director (vslauser, if available)
CIM Providers	Users with administrator privileges on the host	vCenter (vpxuser) Exception users, based on permissions vCloud Director (vslauser, if available)	vCenter (vpxuser) Exception users, based on permissions vCloud Director (vslauser, if available)
Direct Console UI (DCUI)	Users with administrator privileges on the host, and users in the DCUI.Access advanced option	Users defined in the DCUI.Access advanced option Exception users with administrator privileges on the host	DCUI service is stopped.
ESXi Shell (if enabled)	Users with administrator privileges on the host	Users defined in the DCUI.Access advanced option Exception users with administrator	Users defined in the DCUI.Access advanced option Exception users with administrator

		privileges on the host	privileges on the host
SSH (if enabled)	Users with administrator privileges on the host	Users defined in the `DCUI.Access` advanced option Exception users with administrator privileges on the host	Users defined in the `DCUI.Access` advanced option Exception users with administrator privileges on the host

Configure VMware Directory Service

The VMware Directory Service (vmdir) is the component of SSO in vSphere 6.x that provides the SSO domain (directory service) that you create during the installation of SSO. The vmdir service is included on each Platform Services Controller (PSC), including each embedded vCenter Server deployment. It is a multi-tenanted, multi-mastered directory service that provides an LDAP directory on port 389. It also uses port 11711 for backward compatibility with vSphere 5.5.

In environment with multiple instances of PSC, updates to vmdir data in one instance is replicated to the other instances. Starting in vSphere 6.0, vmdir not only stores SSO data, but certificate information as well.

If you decide to use a new VMCA root certificate and you un-publish the VMCA root certificate that was used initially, you must replace the machine certificates, the solution user certificates, and some internal services certificates. You must replace the SSL signing certificate that is used by SSO. You must also replace the VMware Directory Service (vmdir) certificate.

You can use the follow procedure to replace the VMware Directory Service Certificate. Each step provides details for Linux based deployments of SSO and Windows based deployments of SSO.

Step 1. Stop vmdir

Linux: service-control --stop vmdird

Windows: service-control --stop VMWareDirectoryService

Step 2. Copy the certificate and key to the vmdir location

Linux:

```
cp vmdir.crt /usr/lib/vmware-vmdir/share/config/vmdircert.pem
cp vmdir.priv /usr/lib/vmware-vmdir/share/config/vmdirkey.pem
```

Windows:

```
copy vmdir.crt C:\programdata\vmware\vCenterServer\cfg\vmdird\vmdircert.pem
copy vmdir.priv C:\programdata\vmware\vCenterServer\cfg\vmdird\vmdirkey.pem
```

Step 3. Restart vmdir

Linux: `service-control --start vmdird`

Windows: `service-control --start VMWareDirectoryService`

If vmdir replication is interrupted due to a broken link, it should resume once the root cause is fixed. Once the root cause is fixed, the nodes should begin replication and eventually reach a converged state, which may be slow. Do not be concerned unless it does not reach its consistent state in an hour.

Assigning Privileges for ESXi Hosts

Typically, users should access vSphere via vCenter Server, where privileges are managed centrally. For use cases where some users access ESXi hosts directly, you can manage privileges directly on the host. The following roles are predefined directly in ESXi

- Read Only: Ability to view, not change assigned objects
- Administrator: Ability to change assigned objects
- No Access: No access to assigned objects. This role is the default role, which you can override.

In vSphere 6.x, you can use the ESXCLI to manage local user accounts and to configure permissions on local and Active Directory accounts. In vSphere 6.7, you can connect directly to the ESXi host using the vSphere Host Client and navigate to **Manage** > **Security & users** > **Users** to create, edit, and remove local user accounts.

The following user accounts exists on an ESXi host that is not added to a vCenter System

- **root**: A user account that is created and assigned the Administrator role by default on each ESXi host
- **vpxuser**: A local ESXi user account that is created, managed, and used for management activities by vCenter Server.
- **dcui**: A user account that acts as an agent for the direct console and cannot be modified or used by interactive users.

NOTE: You can remove the access privileges for the root user. But, you should first create another user account on the root level and assign it the Administrator role.

Much like vCenter Server, each ESXi host uses role-based permission for users who logon directly to the ESXi host rather than accessing the host through vCenter Server. ESXi allows the creation of custom roles, but these roles are only applied when a user logs directly onto the host, such as when the user uses the vSphere Host Client to connect to the host directly. In most cases, managing roles and permissions at the host level should be avoided or minimalized. You can connect directly to an ESXi 6.7 host using the vSphere Host Client and navigate to **Manage** > **Security & users** > **Roles** to create, edit, and remove roles.

Using Active Directory to Manage ESXi Users

In scenarios where multiple users need to access multiple ESXi hosts directly (rather than accessing vCenter Server), you face challenges in synchronizing user names and passwords. To address the challenges, consider joining the hosts to Active Directory and assigning roles to specific AD users and groups. Then require users to provide their Active Directory credentials when logging directly to the host.

In scenarios where Active Directory users need to access an ESXi directly, you need to add the host to a directory service and apply permissions to those users. You can use the following steps to add an ESXi Host to a Directory Service

To configure an ESXi host to use Active Directory, you can use these steps.

Step 1. Verify that an Active Directory domain is available.

Step 2. Ensure that the host name of the ESXi host is fully qualified with the domain name that matches the domain name of the Active Directory forest. For example, if the Active Directory domain name is `mydomain.com` and the ESXi host name is `host-01`, then the host fully qualified name is `host-01.domain.com`.

Step 3. Synchronize time between the ESXi host and domain controllers using NTP. For details see VMware KB 1035833.

Step 4. Ensure the DNS servers that are configured for the ESXi host can resolve the host names of the Active Directory domain controllers.

Step 5. In the vSphere Web Client, select the ESXi host and select **Authentication Services** on the **Manage** > **Settings** tab.

Step 6. Click the **Join Domain** button.

Step 7. In the dialog box, specify the domain and user credentials. Optionally, specify a proxy server.

Step 8. Enter a domain, either in the form *`name.tld`* or in the form *`name.tld/container/path`*, where `name.tld` is the domain name and `/container/path` is an optional path to an organization unit, where the host computer object should be created. For example, you can use `domain.com/ou01/ou02`. to add the host to an organization unit named `ou02` that resides in an organization unit named `ou01` in a domain named `domain.com`.

Step 10. Click **OK**.

Configuring Smart Card Authentication for ESXi

You can use smart card authentication to log in to the ESXi Direct Console User Interface (DCUI) by using a Personal Identity Verification (PIV), Common Access Card (CAC) or SC650 smart card instead specifying a user name and password. In this case, the DCUI prompts for a smart card and PIN combination instead of prompting for a user name and password. To configure smart card authentication, you should setup the smart card infrastructure (AD domain accounts, smart card readers, smart card, etc.), configure ESXi to join an AD domain that supports smart card authentication, use the vSphere Web Client to add root certificates, and follow these steps.

1. In the vSphere Web Client, browse to the host.
2. Click **Configure**.
3. Under System, select **Authentication Services**.
4. In the Smart Card Authentication panel, click **Edit**.
5. In the Edit Smart Card Authentication dialog box, select the Certificates page.
6. Add trusted Certificate Authority (CA) certificates, for example, root and intermediary CA certificates.
7. Open the Smart Card Authentication page, select the **Enable Smart Card Authentication** check box, and click **OK**

UEFI Secure Boot for ESXi Hosts

Starting with vSphere 6.5, ESXi supports secure boot, which is part of the UEFI firmware standard, if it is enabled in the hardware. With secure boot enabled, a machine refuses to load any UEFI driver or app unless the operating system bootloader is cryptographically signed. In vSphere 6.5 and later, the ESXi bootloader contains and uses a VMware public key to verify the signature of the kernel and a small subset of the system that includes a secure boot VIB verifier that verifies each VIB packages installed on the host.

> NOTE: You cannot perform a secure boot on ESXi servers that were upgraded by using esxcli commands because the upgrade does not update the bootloader.

To resolve issues with secure boot, you can follow these steps.

1. Reboot the host with secure boot disabled.
2. Run the secure boot verification script
3. Examine the information in the `/var/log/esxupdate.log` file

Securing ESXi Hosts with Trusted Platform Module

ESXi 6.7 can use Trusted Platform Modules (TPM) version 2.0 chips, which are secure cryptoprocessors that enhance host security by providing a trust assurance rooted in hardware. A TPM 2.0 chip attests to an ESXi host's identity. Host attestation is the process of authenticating and attesting to the state of the host's software at a given point in time. UEFI secure boot, which ensures that only signed software is loaded at boot time, is a requirement for successful attestation. The TPM 2 chip securely stores measurements of the software modules loaded in the ESXi host and vCenter Server remotely verifies. The automated high level steps of the attestation process are:

1. Establish the trustworthiness of the remote TPM and create an Attestation Key (AK) on it.
2. Retrieve the Attestation Report from the host.
3. Verify the host's authenticity

To use TPM 2.0 chips, you should ensure your vSphere environment meets these requirements:

- vCenter Server 6.7
- ESXi 6.7 host with TPM 2.0 chip installed and enabled in UEFI
- UEFI Secure Boot enabled

Additionally, you should Ensure that the TPM is configured in the ESXi host's BIOS to use the SHA-256 hashing algorithm and the TIS/FIFO (First-In, First-Out) interface and not CRB (Command Response Buffer).

During the boot of an ESXi host with an installed TPM 2.0 chip, vCenter Server monitors the host's attestation status. The vSphere Client displays the hardware trust status in the vCenter Server's Summary tab under Security with the following alarms:

- Green: Normal status, indicating full trust.
- Red: Attestation failed

ESXi Log Files

To increase the security of the host, take the following measures.

- Configure persistent logging to a datastore. By default, the logs on ESXi hosts are stored in the in-memory file system, which only stores 24 hours of data and are lost when you reboot the host. When you enable persistent logging, you have a dedicated activity record for the host.
- Remote logging to a central host allows you to gather log files on a central host. From that host, you can monitor all hosts with a single tool, do aggregate analysis, and search log data.
- Configure the remote secure syslog on ESXi hosts by using vCLI, PowerCLI, or an API client.
- Query the syslog configuration to make sure that the syslog server and port are valid.

Objective 7.5

Role-based User Management - Basic Concepts

This section explains basic concepts of the role-based permissions used in a vSphere environment.

Authentication and Authorization

vCenter Single Sign-On (SSO) provides authentication, which means that it identifies users as they login and validates their credentials. vCenter Server uses permissions and roles to perform authorization, which controls what an authenticated user can do. It allows you to assign a permission to an object in the vCenter Server inventory, by specifying which privileges a specific user or group has on that object.

The default SSO domain name is `vsphere.local`, but you can change it during the domain creation. Initially, only the SSO domain administrator is authorized to log into vCenter Server. By default, the SSO domain administrator is `administrator@vsphere.local`.

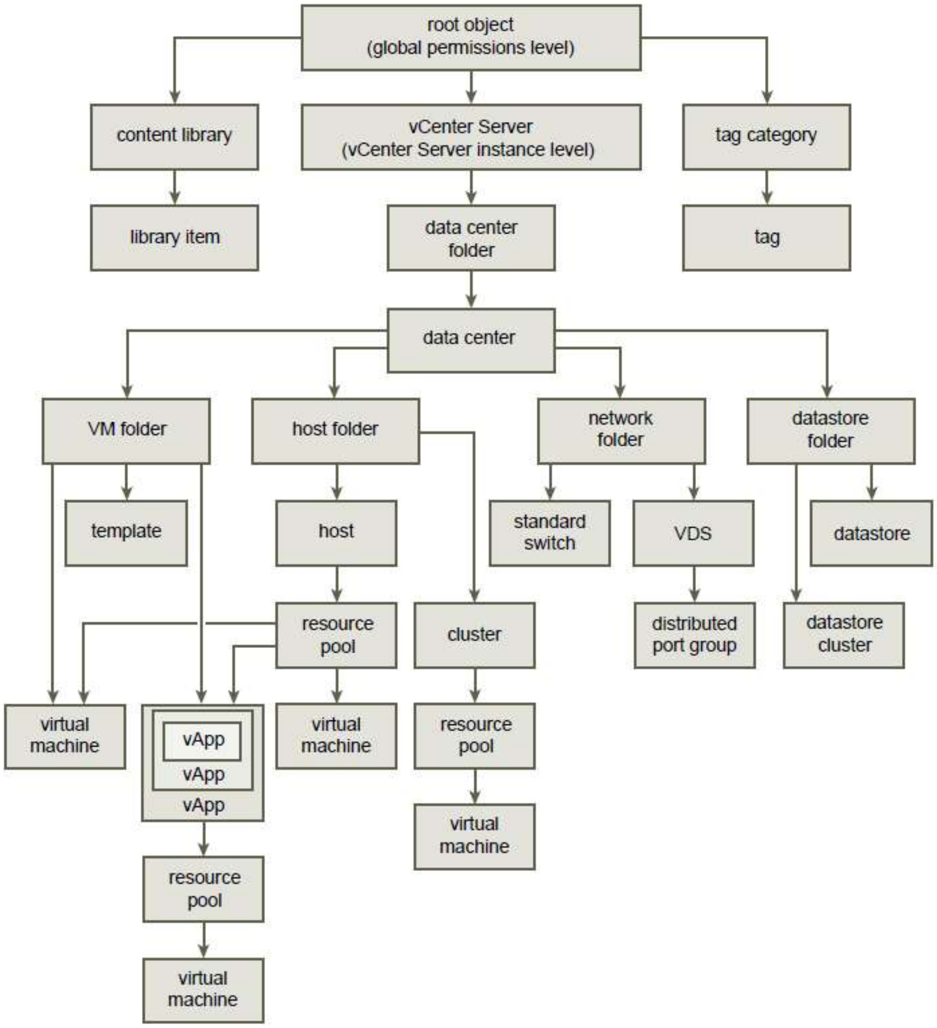
The permission model for vCenter Server systems relies on assigning permissions to objects in the object hierarchy. A permission is the assignment of a user (or group) and a role to an inventory object.

When you add a new identity source to SSO, all users can be authenticated, but will effectively have the `No Access` role to the vCenter Server inventory.

Inventory Hierarchy and Objects

You can assign permissions to objects at different levels of the inventory hierarchy, such as ESXi hosts, clusters, virtual machines, folders, resource pools, datastores and networks. You can also assign permissions to a global root object to apply the permissions to all object in all solutions.   You can apply permissions to container objects and optionally allow the permissions to propagate to its descendent objects. Most objects inherit permissions from its parents via a single path, but virtual machines inherit permissions from virtual machine folders, hosts, resource pools, etc., as you can see in Figure 17-1. If an object inherits permissions from two parent objects, then its inherited permissions are determined by the union of the permissions. Figure 17-1 is a diagram from the *VMware vSphere 6.7 Security Guide* that shows the vSphere Inventory Hierarchy.

Figure 17-1 vSphere Inventory Hierarchy



Objects might have multiple permissions, but only one permission for each user or group. In other words, you cannot assign two permissions on a specific object that specify the same group. If multiple permissions are applied to a specific object using multiple groups and if a specific user belongs to more than one of these groups, then the effective permissions for that user on that object is the union of the privileges in applicable roles.

Privileged users can define permissions on managed objects.

- Clusters
- Data centers

- Datastores
- Datastore clusters
- Folders
- Hosts
- Networks (except vSphere Distributed Switches)
- Distributed port groups
- Resource pools
- Templates
- Virtual machines
- vSphere vApps

Users and Groups

Users can log in to vCenter Server if they are in a domain that has been added as a SSO identity source. By default, immediately following installation, only the `localos` and `vSphere.local` identity sources are available. SSO administrators can add identity sources, such as a native Active Directory (Integrated Windows Authentication) domain or an OpenLDAP directory service. For backward compatibility, Active Directory as an LDAP Server is also available. If your vCenter Server Appliance (or vCenter Server Windows machine) is in an Active Directory domain, then you can use the following procedure to add an Active Directly identity source.

1. Use the vSphere Client to logon to the vCenter Server as an SSO administrator, such as `administrator@vsphere.local`
2. From the Home page, select **Administration**.
3. Under Single Sign On, click **Configuration**.
4. Click **Identity Sources** and click **Add Identity Source**.
5. From the list of available types, select Active **Directory (Integrated Windows Authentication)**

In some cases, you may wish to manage multiple, independent vSphere environments having similar, but separate SSO domains and users. In such scenarios, you can export SSO users using this procedure.

Step 1. Log onto the vSphere Web Client

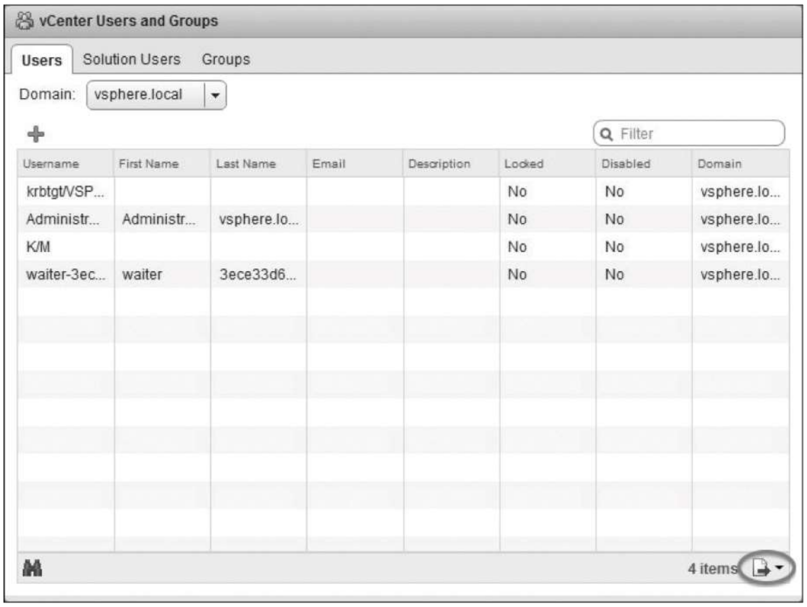
Step 2. Select **Home** > **Administration**.

Step 3. Select **Singe Sign On** > **Users and Groups**

Step 4. Select the **Users** tab

Step 5. click the **Export List** icon in lower right corner, as shown in Figure 17-3

Figure 17-2 Export SSO Users List



Domain: vsphere.local

Username	First Name	Last Name	Email	Description	Locked	Disabled	Domain
krbtgt/VSP...					No	No	vsphere.lo...
Administr...	Administr...	vsphere.lo...			No	No	vsphere.lo...
K/M					No	No	vsphere.lo...
waiter-3ec...	waiter	3ece33d6...			No	No	vsphere.lo...

You can use a similar procedure to export SSO groups, except that you choose **Groups** in Step 4.

Privileges and Roles

Privileges are the lowest-level access controls, which can be used to define the actions that a user can take on an object in the vSphere inventory. Table 17.2 describes a few of the available privilege categories and a few sample privileges in each category.

Table 17-2 Sample Privileges

Category	Sample privileges
Virtual Machine Configuration	Virtual machine.Configuration.Add existing disk
	Virtual machine.Configuration.Add new disk
	Virtual machine.Configuration.Change CPU count
Datastore	Datastore.Allocate space
	Datastore.Browse datastore
	Datastore.Remove file
Virtual machine Snapshot	Virtual machine .Snapshot management. Create snapshot
	Virtual machine .Snapshot management.Rename Snapshot
	Virtual machine .Snapshot management.Revert to snapshot

A role is a set of privileges. The vCenter Server provides many roles out of the box. You cannot modify the vCenter Server System Roles, which are Administrator, Read Only, No Access, and No Cryptography Administrator roles. You can modify the Sample Roles, but VMware recommends that you do not modify these roles directly, but instead clone the roles and modify the clones to suit your case.

> **NOTE:** Changes to roles take effect immediately even for users who are currently logged into vCenter Server. One exception is using searches where the change is not realized until the next time the user logs into vCenter Server.

To create a role in vCenter Server using the vSphere Web Client, you can use this procedure.

Step 1. Click **Administration** > **Roles**,

Step 2. Click **Manage** > **Permissions**

Step 3. Click the **Create role action** (+) button.

Step 4. Provide a name for the role.

Step 5. Select the desired privileges.

Step 6. Click **OK**.

After creating custom roles, you can assign the roles to permissions in the same manner as you assign the vCenter Server system roles and sample roles.

To clone a sample role or custom role in the vSphere Web Client, select the role at **Administration** > **Roles**, click the **Clone role action** icon and provide a name for the new role. To edit a sample role or custom role in the vSphere Web Client, select the role at **Administration** > **Roles**, click the **Edit role action** icon and modify the set of privileges in the role. The sample roles are:

- Resource Pool Administrator (sample)
- Virtual Machine User (sample)
- VMware Consolidated Backup User (sample)
- Datastore Consumer (sample)
- Network Administrator (sample)
- Virtual Machine Power User (sample)
- Content Library Administrator (sample).

Table 17.3 provides details on vCenter Server system roles

Table17-3 System Roles in vCenter Server

System Role	Details
Read Only	Allows the user to view the state of an object and details about the object. For example, users with this role can view virtual machine attributes, but cannot open the VM console.
Administrator	Includes all privileges of the Read Only role plus allows the user to view and perform all actions on the object. If you have the Administrator role on an object, you can assign privileges to individual users and groups. If you have the Administrator role in vCenter Server, you can assign privileges to users and groups in the default SSO identity source. By default, the `administrator@vsphere.local` user has the Administrator role on both vCenter Single Sign-On and vCenter Server.
No Access	Users with the No Access role for an object cannot view or change the object in any way. New users and groups are effectively assigned this role by default.
No Cryptography Administrator	Users with the No cryptography administrator role for an object have the same privileges as users with the Administrator role, except for Cryptographic operations privileges. This role allows administrators to designate other administrators that cannot encrypt or decrypt virtual machines or access encrypted data, but that can perform all other administrative tasks.

Roles that were added in vSphere 6.0 are `Tagging Admin` and `Content Library Administrator(sample).` Although the `Tagging Admin` role does not contain *sample* in its name, it appears to be a sample role, because it can be edited. The `Tagging Admin` role includes only a few privileges, all of which reside in the `Inventory Service` > `vSphere Tagging` category, but it does not include all the privileges in this category. It does not include privileges related to managing the tagging scope. Likewise, the `Content Library Administrator` role contains only privileges in the Content Library category, but does not include all the privileges in the category.

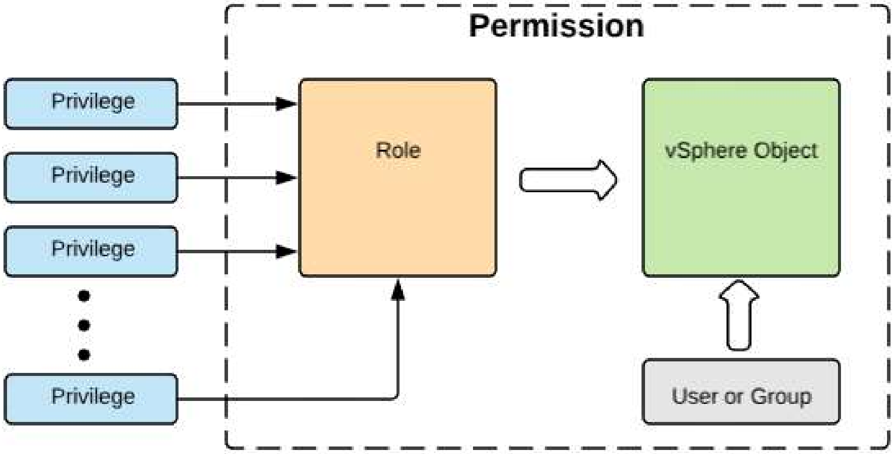
To get familiar with the privileges in a sample role, edit the role explore the privileges that are included in the role. For example, if you edit the `VMware Consolidated Backup User` role, you will see that it only includes some privileges in the Virtual Machine category and no other privileges. Specifically, it includes only these privileges:

- Virtual machine > Configuration > Disk lease
- Virtual machine > Provisioning > Allow read-only disk access
- Virtual machine > Provisioning > Allow virtual machine download
- Virtual machine > Snapshot management > Create snapshot
- Virtual machine > Snapshot management > Remove snapshot

Permissions

The permission model for vCenter Server systems relies on assigning permissions to objects in the object hierarchy. A permission is the assignment of a user (or group) and a role to an inventory object. A permission is set on an object in the vCenter object inventory. Each permission associates the object with a group (or user) and a role, as illustrated in Figure 17-3. For example, you can select a virtual machine object, add one permission that gives the `ReadOnly` role to `Group 1`, and add a second permission that gives the `Administrator` role to `User 2`.

Figure 17-3 vSphere Permissions



To set a permission using the vSphere Client, you can use the following steps.

1. Select the object in the inventory
2. Click the **Permissions** tab
3. Click the Add Permission icon
4. Select a user or group form the **User** drop-down menu
5. Select a role from the **Role** drop-down menu
6. Optionally, select **Propagate to children**
7. Click **OK**

By assigning a different role to a group of users on different objects, you control the tasks that those users can perform in your vSphere environment. For example, to allow a group to configure memory for the host, select that host and add a permission that grants a role to that group that includes the **Host.Configuration.Memory Configuration** privilege.

Global permissions.

Most entities that appear in the vCenter inventory are managed objects, whose access can be controlled using permissions. You cannot modify permissions on entities that derive permissions from the root vCenter Server system:

- Custom fields
- Licenses
- Roles
- Statistics intervals
- Sessions

The global root object is used to assign permissions across solutions. The vCenter Server is an example of a solution and it is attached as a child to the global root object in the hierarchy. The Content Library and Tag Category objects are also attached as children to the global root object. Global permissions are permissions that are applied to the global root object and span solutions. For example, a global permission can be applied to both vCenter Server and vRealize Orchestrator. Each solution has its own root object in the hierarchy, whose parent is the global root object. You can give a group of users Read permissions to all objects in both object hierarchies.

In some cases, you may assign a global permission and choose not to propagate to child objects. This may be useful to provide a global functionality, such as creating roles. To assign a global permission, you should use the vSphere Web Client with a user account that has the **Permission > Modify permission** privilege on the root object of all inventory hierarchies. Select **Administration** > **Global Permissions** > **Manage** and use the **Add Permission** icon (green plus sign) dialog to select the desired user group (or user) and role, as described previously in this chapter.

> **NOTE:** By default, the administrator account in the SSO domain, such as `administrator@vsphere.local`, can modify global permissions, but the vCenter Server appliance `root` account cannot.

NOTE: Be careful when applying global permission. Decide if you truly want the permission to apply to all solutions and to all objects in all inventory hierarchies.

Objective 7.6

VCP6-DCV Cert Guide:

- Chapter 13 – Objective 7.5
- Chapter 15 - Objective 9.1
- Chapter 15 - Objective 9.2

DRS cluster configuration / administration

HA cluster configuration / administration

VSAN cluster configuration / administration

Create a VSAN Cluster

To create a VSAN cluster, apply following characteristics, prerequisites, and procedure:

Characteristics:

- You can have multiple vSAN clusters for each vCenter Server instance. You can use a single vCenter Server to manage more than one vSAN cluster.
- vSAN consumes all devices, including flash cache and capacity devices, and does not share devices with other features.
- vSAN clusters can include hosts with or without capacity devices. The minimum requirement is three hosts with capacity devices. For best results, create a vSAN cluster with uniformly configured hosts.
- If a host contributes capacity, it must have at least one flash cache device and one capacity device.
- In hybrid clusters, the magnetic disks are used for capacity and flash devices for read and write cache. vSAN allocates 70% of all available cache for read cache and 30% of available cache for the write buffer. In a hybrid configuration, the flash devices serve as a read cache and a write buffer.
- In all-flash clusters, one designated flash device is used as a write cache and additional flash devices are used for capacity. In all-flash clusters, all read requests come directly from the flash pool capacity.
- Only local or direct-attached capacity devices can participate in a vSAN cluster. vSAN cannot consume other external storage, such as SAN or NAS, attached to cluster.

Prerequisites:

To use the full set of vSAN capabilities, the ESXi hosts that participate in vSAN clusters must be version 6.5 or later. During the vSAN upgrade from previous versions, you can keep the current on-disk format version, but you cannot use many of the new features. vSAN 6.6 and later software supports all on-disk formats. Prior to creating the VSAN cluster, verify the following:

- ESXi hosts
    - Use ESXi 6.5 or later. (Alternatively, if you do not need the latest VSAN features, ESXi 5.5 Update 1 or later hosts can join the VSAN cluster. All hosts in the VSAN cluster must have the same on-disk format.)
    - Prepare at least three hosts, and preferably four or more hosts.
    - A host does not have to contribute storage to the VSAN cluster. However, to be able to access a VSAN datastore, a host must be a member of the VSAN cluster.
- Memory
    - Configure each host with at least 8GB.
    - Configure each host for 32GB if you need larger configurations and better performance.
- Storage controller
    - Verify that the storage I/O controllers, drivers, and firmware versions are VSAN certified.
    - Configure the controller for passthrough or RAID 0 mode.
    - Disable the controller cache and advanced features. Alternatively, set the controller cache to 100%.
    - Use a controller with queue depth of at least 256.
- Cache and capacity
    - Ensure that each host that contributes storage to the VSAN cluster has at least one cache drive and one capacity drive. These devices must be dedicated to VSAN and not used for other purposes, such as Virtual Flash, VMFS, or boot partitions.
    - For best results, create the VSAN cluster with uniformly configured ESXi hosts.
- Network connectivity
    - Configure each host with at least one network adapter.
    - For hybrid configurations, ensure at least 1GbE is dedicated to VSAN
    - For all flash configurations, ensure at least 10GbE is dedicated to VSAN.
- vCenter Server
    - Use vCenter Server 6.5 or later.
- License key
    - Use a valid VSAN license key that supports the features you require, such as all flash, stretched clusters, deduplication, and compression.
    - Ensure the license capacity is equal to (or greater than) the total number of CPUs that participate in the VSAN cluster.

Procedure:

The following procedure can be used to create a VSAN cluster using the vSphere Web Client:

1. Configure a VMkernel network for VSAN.

    a. On each host that will participate in the cluster, create a VMkernel network adapter.
    b. In the settings of the VMkernel network adapter, select the **vSAN traffic** check box.

2. Create a VSAN cluster.
    a. Right-click a data center and select **New Cluster**.
    b. Provide a name for the cluster.
    c. Select **VSAN Turn ON** check box and click **OK**.
    d. Add hosts to the cluster by dragging and dropping existing hosts or by right-clicking the cluster and choosing **Add Host**.
    e. Select the vCenter Server, click the **Configure** tab, and then click **Storage Providers**. Verify that each host has a VSAN storage provider, but only one is enabled.
3. Configure a VSAN cluster.
    a. Select the VSAN cluster and click the **Configure** tab.
    b. Select the desired VSAN capabilities, such as deduplication, encryption, and fault tolerance mode.
    c. Click **Next**.
    d. Use the **Claim disks** page to select the cache and capacity disks to be used by the cluster. Click **Next**.
    e. If you selected **Configure two host vSAN cluster**, then choose a witness host and claim disks for the witness host.
    f. If you selected **Configure stretched cluster**, define fault domains for the cluster, choose a witness host, and claim disks for the witness host.
    g. If you selected **Configure fault domains**, define fault domains for the cluster.
    h. On the **Ready to Complete** page, click **Finish**.

Create Disk Groups

When you create disk groups, you must specify each host and each device to be used for the vSAN datastore. You organize cache and capacity devices into disk groups. To create a disk group, you define the disk group and individually select devices to include in the disk group. Each disk group contains one flash cache device and one or more capacity devices.

The vSAN cluster initially contains a single vSAN datastore with zero bytes consumed. As you create disk groups on each host and add cache and capacity devices, the size of the datastore increases according to the amount of physical capacity added by those devices. vSAN creates a single distributed vSAN datastore using the local empty capacity available from the hosts added to the cluster.

If the cluster requires multiple flash cache devices, you must create multiple disk groups, because a maximum of one flash cache device is allowed per disk group.

> **NOTE** When you add an ESXi host to a vSAN cluster, the local storage from that host is not added to the vSAN datastore automatically. You have to create a disk group using storage from the new ESXi host.

The following procedure can be used to create a disk group on a VSAN host using the vSphere Web Client:

1. Select the VSAN cluster and click the **Configure** tab.
2. Under **VSAN**, click **Disk Management**.
3. Select the host and click the **Create a new disk group** icon.
4. Select a flash device to be used for cache.
5. In the **Capacity type** drop-down menu, select the type of capacity disks to be used (HDD or Flash).
6. Select the capacity drives.

7. Click **OK**.

Alternatively, you claim storage devices for a VSAN cluster and allow VSAN to organize the devices into default disk groups. To do this, select the cluster and then select **Configure** > **Disk Management** > **Claim Disks**. For each host, select any available desired device and click **Claim for cache tier** or click **Claim for capacity tier**.

Monitor VSAN

You can use the vSphere Web Client to monitor the following items:

- **The VSAN cluster**: Select the VSAN cluster and click **Monitor** > **vSAN**. Select **Physical Disks** to review hosts, cache devices, and capacity devices. Select **Health** to review VSAN health categories. Select **Configure** > **General** to check cluster status, Internet connectivity, and on-disk format.
- **VSAN capacity**: Select the VSAN cluster and click **Monitor** > **vSAN**. Select **Capacity** to review provisioned and used capacity. Here, you can view the percentage of capacity used by object type, such as virtual disks, swap objects, file system overhead, and deduplication/compression overhead.
- **Virtual devices in the VSAN cluster**: Select the VSAN cluster and click **Monitor** > **vSAN**. Select **Virtual Disks** to review the virtual disks in the VSAN cluster, their physical disk placement, and compliance failures.
- **Resynchronization tasks in the VSAN cluster**: Select the VSAN cluster and click **Monitor** > **vSAN**. Select **Resyncing Components** to track the progress of resynchronization of virtual machine objects and the number of remaining bytes.
- **Devices that participate in the VSAN datastore**: Select the VSAN cluster, click **Configure** > **Device Backing**, and select a disk group. View the devices in the **Disks** table.
- **VSAN health**: Select the VSAN cluster, click **Configure** > **VSAN** > **Health and Performance**, and then click the **Health Services > Edit Settings** button. You can turn on periodic vSAN health checks covering hardware compatibility, network configuration and operation, advanced vSAN configuration options, storage device health, and virtual machine objects.
- **VSAN performance**: Select the VSAN cluster and click **Configure** > **VSAN** > **Health and Performance**. Click **Edit** and select the **Turn on vSAN performance service** check box. This allows you to monitor the performance of your vSAN environment and investigate potential problems. With this setting, the cluster summary displays an overview of vSAN performance statistics, including IOPS (input/output operations per second), throughput, and latency.
- **VSAN default alarms**: You can examine the configuration of the VSAN default alarms and respond to these alarms whenever they are triggered. You cannot modify these alarms, but you can create custom VSAN alarms. To view the default VSAN alarms, select the cluster, select **Configure** > **Alarm Definitions**, and search for “vSAN.”
- **Customer VSAN Alarms based on VMkernel Observations (VOBs)**: Select the vCenter Server and then select **Configure** > **Alarm Definitions** to create an event-based alarm. In the wizard, select **specific event occurring on this object** and use **Triggers** to add a vSAN event.

NOTE When a hardware device, host, or network fails, or if a host is placed into maintenance mode, vSAN initiates resynchronization in the vSAN cluster. The following events can trigger resynchronization: changing a virtual machine storage policy, restarting a host after a failure, recovering hosts from a failure, evacuating data by using the full data migration mode before placing a host in maintenance mode, and exceeding the capacity threshold (80% by default) of a capacity device.

SDRS cluster configuration / administration

VCP6-DCV Cert Guide: Chapter 11 - Monitor/Troubleshoot Storage Distributed Resource Scheduler (SDRS) Issues

Objective 7.7

Chapter 13 vCEnter and host Management guide

CPU compatability and EVC pp 143

vMotion. Pp 132

Storage vMotion pp 142

Migrated powered off or suspsended VM in vSphere Web Client. Pp 152

Migrate VM to new compute resource. Pp 155

Migrate VM to new storage. Pp 160

Limits on simultaneous migrations pp 1565

Compatability checks. Pp 167

Requirements for migration between vCenter Server instances. Pp 141

- The source and destination vCenter Server instances and ESXi hosts must be 6.0 or later.

- The cross vCenter Server and long-distance vMotion features require an Enterprise Plus license. For more information, see http://www.vmware.com/uk/products/vsphere/compare.html.

- Both vCenter Server instances must be time-synchronized with each other for correct vCenter Single Sign-On token verification.

- For migration of compute resources only, both vCenter Server instances must be connected to the shared virtual machine storage.

- When using the vSphere Web Client, both vCenter Server instances must be in Enhanced Linked Mode and must be in the same vCenter Single Sign-On domain. If the vCenter Server instances exist in separate vCenter Single Sign-On domains, you can use vSphere APIs/SDK to migrate virtual machines.

(Cross vCenter, Storage vMotion, vMotion etc)

Objective 7.8

VCP6-DCV Cert Guide:

- Chapter 8 – Objective 5.1
- Chapter 12 – Objective 7.4
- Chapter 15 - Objective 9.2

vSphere 6.7 Resource Management Guide

Resource allocation and admission control

Chapter 2

Resource pools and how to manage them

Chapter 10

DRS

Chaptesr 11 and 12

Datastore clusters, Storage DRS, Storage I/O Control

Chapters 9 and 13

Objective 7.9

VCP6-DCV Cert Guide: Chapter 16 – Objective 10.1

vSphere Virtual Machine Management: https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-esxi-vcenter-server-67-virtual-machine-admin-guide.pdf

New VM Wizard

Page 18

Deploy VM from a template

Page 27

Deploy VM from a template in the vSphere Web Client

Page 31

vSphere Web (Flex) Client wizard is similar to the vSphere Client (HTML5) wizard, but with fewer choices. Eg: only the HTML 5 client provide the option to: clone a VM that has vPMem hard disks

Clone an existing VM.

Page 35

Important: If the virtual machine that you clone has an NVDIMM device and virtual PMem hard disks, the destination host or cluster must have available PMem resource. Otherwise, you cannot proceed with the task

Important: You cannot change the storage policy if you clone an encrypted virtual machine.

Clone an existing VM in the vSphere Web Client.

Page 47

vSphere Web (Flex) Client wizard is similar to the vSphere Client (HTML5) wizard, but with fewer choices. Eg: only the HTML 5 client provide the option to: clone a VM that has vPMem hard disks

Clone a VM with Instant Clone

Page 43

In vSphere 6.7, you can Instant Clone a virtual machine only through the API calls.

Convert a template to a VM.

Page 50

Deploy OVF and OVA templates

Page 53

vRO workflows: Basic virtual machine management workflows

vCenter Server and Host Management Guide Update 1 (vsphere 6.7) page 191

Objective 7.10

VCP6-DCV Cert Guide: Chapter 16 – Objective 10.1

vSphere Virtual Machine Management: https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-esxi-vcenter-server-67-virtual-machine-admin-guide.pdf

You cannot modify templates after you create them. To alter an existing template, you must convert it to a virtual machine, make the required changes, and convert the virtual machine back to a template. To preserve the original state of a template, clone the template to a template.

Clone VM to a template

Page 21

Clone a VM to a template in the vSphere Web Client

Page 24

vSphere Web (Flex) Client wizard is similar to the vSphere Client (HTML5) wizard, but with fewer choices. Eg: only the HTML 5 client provide the option to: clone a VM that has vPMem hard disks

Clone a template to a template

Page 44

Clone a template to a template using the vSphere Web client

Page 47

vSphere Web (Flex) Client wizard is similar to the vSphere Client (HTML5) wizard, but with fewer choices. Eg: only the HTML 5 client provide the option to: clone a VM that has vPMem hard disks

Objective 7.11

VCP6-DCV Cert Guide: Chapter 8 – Objective 5.1

A large vSphere implementation might contain several virtual data centers with a complex arrangement of hosts, clusters, resource pools, and networks. It might involve multiple vCenter Server systems connected using Enhanced Linked Mode. Smaller implementations might require a single virtual data center with a much less complex topology. Regardless of the scale of your virtual environment, consider how the virtual machines it will support are going to be used and administered.

Here are the questions to answer as you create and organize an inventory of virtual objects:

- Will some virtual machines require dedicated resources?
- Will some virtual machines experience periodic spikes in workload?
- Will some virtual machines need to be administered as a group?
- Do you want to use multiple vSphere Standard Switches, or you want to have a single vSphere Distributed Switch per data center?

- Do you want to use vMotion and Distributed Resource Management with certain virtual machines but not others?

- Will some virtual objects require one set of system permissions, while other objects will require a different set of permissions?

The left pane of the vSphere Web Client displays your vSphere inventory. You can add and arrange objects in any way with the following restrictions:

- The name of an inventory object must be unique with its parent.

- vApp names must be unique within the Virtual Machines and Templates view.

- System permissions are inherited and cascade.

Create data centers.

A virtual data center is a container for all the inventory objects required to complete a fully functional environment for operating virtual machines. You can create multiple data centers to organize sets of environments. For example, you might create a data center for each organizational unit in your enterprise or create some data centers for high-performance environments and others for less demanding virtual machines.

Prerequisites

In the vSphere Client, verify that you have sufficient permissions to create a data center object.

Procedure

1. In the vSphere Client, navigate to the vCenter Server object.

2. Select **Actions > New Datacenter**.

3. Rename the data center and click **OK**.

Add a host to vCenter Inventory

You can add hosts under a data center object, folder object, or cluster object. If a host contains virtual machines, those virtual machines are added to the inventory together with the host

Prerequisites

- Verify that a data center, folder, or cluster exists in the inventory.
- Obtain the user name and password of the root user account for the host.
- Verify that hosts behind a firewall are able to communicate with the vCenter Server system and all other hosts through port 902 or other custom-configured port.
- Verify that all NFS mounts on the host are active.
- If you want to add a host with more than 512 LUNs and 2,048 paths to the vCenter Server inventory,

verify that the vCenter Server instance is suitable for a large or x-large environment.
Required privileges:
-Host.Inventory.Add host to cluster
-Resource.Assign virtual machine to resource pool

-System.View on the virtual machines folder where you want to place the virtual machines of the host.

Procedure

1. In the vSphere Client, navigate to a data center, cluster, or folder within a data center.
2. Right-click the data center, cluster, or folder and select **Add Host**.
3. Type the IP address or the name of the host and click **Next**.
4. Type administrator credentials and click **Next**.
5. Review the host summary and click **Next.**
6. License the host through one of the following methods: Assign an already existing license, Assign a new license.
7. In the Add Host wizard click **Next**.
8. (Optional) Select a lockdown mode option to disable the remote access for the administrator account after vCenter Server takes control of this host.
9. (Optional) If you add the host to a data center or a folder, select a location for the virtual machines that reside on the host and click **Next**.
10. Review the summary and click **Finish**.

A new task for adding the host appears in the Recent Tasks pane. It might take a few minutes for the task to complete.

Create a folder.

You can use folders to group objects of the same type for easier management. For example, permissions can be applied to folders, allowing you to use folders to group objects that should have a common set of permissions.

A folder can contain other folders, or a group of objects of the same type. For example, a single folder can contain virtual machines and another folder containing virtual machines, but it cannot contain hosts and a folder containing virtual machine

You can create these types of folders: Host and Cluster folders, Network folders, Storage folders, and VM and Template folders

If the parent object is a data center, then right-click it and select one of the following:

- Select New Folder > New Host and Cluster Folder.
- Select New Folder > New Network Folder.
- Select New Folder > New Storage Folder.
- Select New Folder > New VM and Template Folder.

Create clusters.

A cluster is a group of hosts. When a host is added to a cluster, the resources of the host become part of the resources of the cluster. The cluster manages the resources of all hosts within it.

Clusters enable vSphere High Availability (HA), vSphere Distributed Resource Scheduler (DRS), and the VMware vSAN features.

Starting with vSphere 6.7, you can create and configure a cluster that is hyper-converged. The hyper-converged infrastructure collapses compute, storage, and networking on a single software layer that runs on industry standard x86 servers.

You create and configure a cluster by either using the vSphere Web Client or through the simplified Quickstart workflow in the vSphere Client. In the **Cluster quickstart** page, there are three cards for configuring your new cluster.

1. Cluster basics
2. Add hosts
3. Configure cluster

The **Skip Quickstart** button prompts you to continue configuring the cluster and its hosts manually. You click **Continue** to confirm exiting the simplified configuration workflow. Once dismissed, there is no option to restore the **Cluster quickstart** workflow for the current cluster.

vCenter Server and Host Management Guide Update 1 (vsphere 6.7)

Configure a cluster. Pp 76

Extend a cluster pp 78

Objective 7.12

VCP6-DCV Cert Guide: Chapter 1 – Objective 1.1

Best practices for roles and permissions

VMware recommends the following best practices when configuring roles and permissions in your vCenter Server environment:

- Where possible, assign roles to groups rather than to individual users.
- Grant permissions to users (groups) only on the objects where they are required. Use the minimum number of permissions to meet the required functionality.
- If you assign a restrictive role to a group, check that the group does not contain the Administrator user or other users who require administrative privileges.
- Use folders to group objects. For example, to grant modify permission on one set of hosts and view permission on another set of hosts, place each set of hosts in a folder.
- Use caution when adding a permission to the root vCenter Server objects. Users with privileges at the root level have access to global data on vCenter Server, such as roles, custom attributes, and vCenter Server settings.
- Consider enabling propagation when you assign permissions to an object. Propagation ensures that new objects in the object hierarchy inherit permissions. For example, you can assign a permission to a virtual machine folder and enable propagation to ensure the permission applies to all VMs in the folder.
- Use the `No Access` role to mask specific areas of the hierarchy. The `No Access` role restricts access for the users or groups with that role.

NOTE: License propagation happens even if the user does not have privileges on all vCenter Server systems. Changes to licenses propagate to all vCenter Server systems that are linked to the same Platform Services Controller and to Platform Services Controller instances in the same SSO domain.

.

Required privileges for common tasks

Many tasks require permissions on multiple objects in the inventory. Consider the following:

- To perform any operation that consumes storage space, such taking a snapshot, you must have the **Datastore.Allocate Space** privilege on the target datastore in addition to having the directly required privileges on the major object.
- Moving an object in the inventory hierarchy requires appropriate privileges on the object itself, the source parent object (such as a folder or cluster), and the destination parent object.
- Deploying a virtual machine directly to a host or cluster requires the **Resource.Assign Virtual Machine to Resource Pool** privilege, because each host or cluster has its own implicit resource pool.

Table 17-4 Required permissions for common tasks

Task	Required Privileges
Create a virtual machine	On the destination folder or datacenter: **Virtual Machine.Inventory.Raw Create new** **Virtual Machine.Configuration.Add New Disk** **Virtual Machine .Configuration.Add Existing Disk** **Virtual Machine.Configuration.Raw Device** On the destination host, cluster, or resource pool: **Resource. Assign Virtual Machine to Resource Pool** On the destination datastore or datastore folder: **Datastore.Allocate Space** On the network **Network.Assign Network**
Deploy a virtual machine from a template	On the destination folder or datacenter: **Virtual Machine.Inventory.Create from existing** **Virtual Machine.Configuration.Add New Disk** On a template or template folder: **Virtual Machine.Provisioning.Deploy Template** On the destination host, cluster or resource pool: **Resource.Assign Virtual Machine to Resource Pool** On the destination datastore or folder of datastores: **Datastore.Allocate Space**

	On the network that the virtual machine will be assigned to: **Network.Assign Network**
Take a virtual machine snapshot	On the virtual machine or a folder of virtual machines: **Virtual Machine.Snapshot Management.Create Snapshot** On the destination datastore or folder of datastores: **Datastore.Allocate Space**
Move a virtual machine into a resource pool	On the virtual machine or folder of virtual machines: **Resource.Assign Virtual Machine to Resource Pool** **Virtual Machine.Inventory.Move** On the destination resource pool: **Resource.Assign Virtual Machine to Resource Pool**
Install a guest operating system on a virtual machine	On the virtual machine or folder of virtual machines: **Virtual Machine.Interaction.Answer Question** **Virtual Machine.Interaction.Console Interaction** **Virtual Machine.Interaction.Device Connection** **Virtual Machine.Interaction.Power Off** **Virtual Machine.Interaction.Power On** **Virtual Machine.Interaction.Reset** **Virtual Machine.Interaction.Configure CD Media** **Virtual Machine.Interaction.Configure Floppy Media** **Virtual Machine.Interaction.Tools Install** On a datastore containing the installation media ISO image: **Datastore.Browse Datastore** On the datastore to which you upload the installation media ISO image: **Datastore.Browse Datastore** **Datastore.Low Level File Operations**
Migrate a virtual machine with vMotion	On the virtual machine or folder of virtual machines: **Resource.Migrate Powered on Virtual Machine** **Resource.Assign Virtual Machine to Resource Pool** On the destination host, cluster, or resource pool: **Resource.Assign Virtual Machine to Resource Pool**

Cold migrate (relocate) a virtual machine	On the virtual machine or folder of virtual machines: **Resource.Migrate Powered Off Virtual Machine** **Resource.Assign Virtual Machine to Resource Pool** On the destination host, cluster, or resource pool: **Resource.Assign Virtual Machine to Resource Pool** On the destination datastore: **Datastore.Allocate Space**
Migrate a Virtual Machine with Storage vMotion	On the virtual machine or folder of virtual machines: **Resource.Migrate Powered On Virtual Machine** On the destination datastore: **Datastore.Allocate Space**
Move a host into a cluster	On the host: **Host.Inventory.Add Host to Cluster** On the destination cluster: **Host.Inventory.Add Host to Cluster**

If you wish to modify an existing permission, you can edit the permission and change role assignment. You cannot change the object, user or user group in the permission, but you can change the role. If this is not adequate, then remove the permission and create a new permission with the correct settings. This work must be done as a user with sufficient privileges to change permissions on the associated object.

The biggest challenge in editing permissions may be locating the permission, so it can be modified. If you know the object on which the permission was created, then you can select the object in the vSphere Web Client inventory, select **Manage** > **Permissions**, right-click the permission and choose **Change Role**. Select the appropriate role and click **OK**.

If you do not already know which permission to modify or on which object the permission is assigned, you may need to investigate. Begin by selecting an object in the inventory on which you know the applied user permissions are incorrect. Select **Manage** > **Permissions** to discover all the permissions that apply to the object. Use the **Defined in** column to identify were each applied permission is defined. Some of the permissions may be assigned directly on the object and some may be assigned to ancestor objects. Determine which permissions are related to the issue and where they are assigned. For example, in the next section, where permission inheritance is explained, review the provided scenario. In the scenario, if you want User-E to have the ability to view the host-02 object, then modify the permission on host-02 that assigns the `No Access` role to assign `Read Only` to Group-04. In many cases, you will decide that you do not want to change any existing permission, but instead add a new permission. For example, in the same scenario, if you want User-C to have administrator control on host-02, you may consider changing the existing permission on host-02 that assigns `Read Only` to Group-02. But, if you change that permission it also impacts other users, such as User-A. So, the solution may be to add a new permission on host-02 that assigns the `Administrator` role to just User-C. For more details, read the following section.

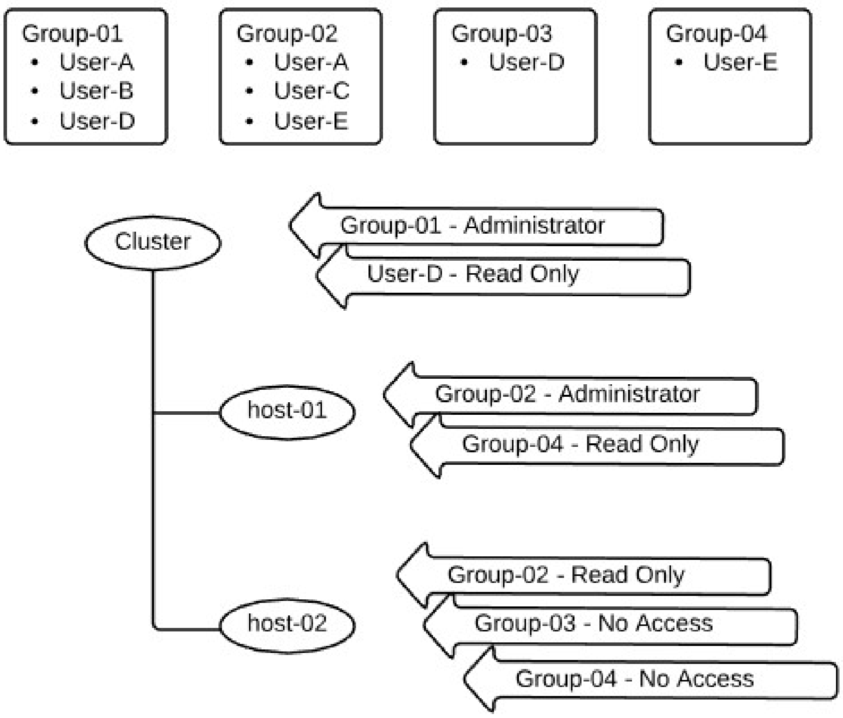
How Permissions are Applied by vCenter Server.

As you assign each permission, you can choose whether to allow the permission to propagate to child objects. This setting is made per permission. It cannot be universally applied. The default setting is to allow propagation to child objects. The propagation is applied to the vSphere Inventory Hierarchy as shown in Figure 17-1.

In the case where conflicting permissions are applied to an object and to its ancestors, the permissions that are assigned at a lower level object in the inventory hierarchy override permissions assigned at a higher level object. In the case where multiple permissions are assigned to the same object to different groups that contain a specific user, then that user's effective permissions are the union of the associated privileges. Permissions assigned to a user override permissions assigned to groups containing the user, when the permissions are applied to the same object. The `No Access` permission is given precedence over all other roles. To illustrate this, consider the following scenario, which is illustrated in Figure 17-4.

- One cluster exists in the inventory, which contains host-01 and host-02.
- The user account User-A is a member of groups Group-01 and Group-02
- The user account User-B is a member of group Group-01
- The user account User-C is a member of group Group-02
- The user account User-D is a member of groups Group-01 and Group-03
- The user account User-E is a member of groups Group-02 and Group-04
- **Propagate to Child Objects** is enabled for each of the following permissions
- A permission assigns Group-01 the Administrator role on the Cluster
- A permission assigns Group-02 the Administrator role on host-01
- A permission assigns Group-02 the Read Only role on host-02
- A permission assigns User-D the Read Only role on the cluster
- A permission assigns Group-03 the No Access role on host-02
- A permission assigns Group-04 the Read Only role on host-01
- A permission assigns Group-04 the No Access role on host-02

Figure 17-4 Permission Scenario



In this scenario, the following effective permissions apply

- User-A:
  - Can perform all tasks on the cluster object
  - Can perform all tasks on the host-01 object
  - Can only view the host-02 object
- User-B:
  - Can perform all tasks on the cluster object
  - Can perform all tasks on the host-01 object
  - Can perform all tasks on the host-02 object
- User-C:
  - Cannot view or perform any task on the cluster object
  - Can perform all tasks on the host-01 object

    - Can only view the host-02 object

- User-D
    - Can only view the cluster object
    - Can only view the host-01 object
    - Cannot view or perform any task on the host-02 object
- User-E
    - Cannot view or perform any task on the cluster object
    - Can perform all tasks on the host-01 object
    - Cannot view or perform any task on the host-02 object

Objective 7.13

VCP6-DCV Cert Guide: Chapter 15 – Objective 9.2

A VM-Host affinity rule specifies whether or not the members of a selected virtual machine DRS group can run on the members of a specific host DRS group.

Unlike a VM-VM affinity rule, which specifies affinity (or anti-affinity) between individual virtual machines, a VM-Host affinity rule specifies an affinity relationship between a group of virtual machines and a group of hosts. There are 'required' rules (designated by "must") and 'preferential' rules (designated by "should".)

A VM-Host affinity rule includes the following components.

- One virtual machine DRS group.
- One host DRS group.
- A designation of whether the rule is a requirement ("must") or a preference ("should") and whether it is affinity ("run on") or anti-affinity ("not run on").

A VM-VM affinity rule specifies whether selected individual virtual machines should run on the same host or be kept on separate hosts. This type of rule is used to create affinity or anti-affinity between individual virtual machines that you select.

When an affinity rule is created, DRS tries to keep the specified virtual machines together on the same host. You might want to do this, for example, for performance reasons.

With an anti-affinity rule, DRS tries to keep the specified virtual machines apart. You could use such a rule if you want to guarantee that certain virtual machines are always on different physical hosts. In that case, if a problem occurs with one host, not all virtual machines would be placed at risk.

You can create VM-VM affinity rules to specify whether selected individual virtual machines should run on the same host or be kept on separate hosts.

You can create and use multiple VM-VM affinity rules, however, this might lead to situations where the rules conflict with one another.

If two VM-VM affinity rules are in conflict, you cannot enable both. For example, if one rule keeps two virtual machines together and another rule keeps the same two virtual machines apart, you cannot enable both rules. Select one of the rules to apply and disable or remove the conflicting rule.

When two VM-VM affinity rules conflict, the older one takes precedence and the newer rule is disabled. DRS only tries to satisfy enabled rules and disabled rules are ignored. DRS gives higher precedence to preventing violations of anti-affinity rules than violations of affinity rules.

Create a VM-VM Affinity Rule

Procedure

1. Browse to the cluster in the vSphere Client.
2. Click the **Configure** tab.
3. Under **Configuration**, click **VM/Host Rules**.
4. Click **Add**.
5. In the **Create VM/Host Rule** dialog box, type a name for the rule.
6. From the **Type** drop-down menu, select either **Keep Virtual Machines Together** or **Separate Virtual Machines**.
7. Click **Add**.
8. Select at least two virtual machines to which the rule will apply and click **OK**.
9. Click **OK**.

VM-VM Affinity Rule Conflicts

You can create and use multiple VM-VM affinity rules, however, this might lead to situations where the rules conflict with one another.

If two VM-VM affinity rules are in conflict, you cannot enable both. For example, if one rule keeps two virtual machines together and another rule keeps the same two virtual machines apart, you cannot enable both rules. Select one of the rules to apply and disable or remove the conflicting rule.

When two VM-VM affinity rules conflict, the older one takes precedence and the newer rule is disabled. DRS only tries to satisfy enabled rules and disabled rules are ignored. DRS gives higher precedence to preventing violations of anti-affinity rules than violations of affinity rules.

Objective 7.14

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.monitoring.doc/GUID-9272E3B2-6A7F-427B-994C-B15FF8CADC25.html

Alarm overview

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.monitoring.doc/GUID-9272E3B2-6A7F-427B-994C-B15FF8CADC25.html

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.monitoring.doc/GUID-A3816E63-62BB-433F-8AAE-17CECA238874.html

Setting an Alarm

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.monitoring.doc/GUID-E30ED662-D851-4230-9AFE-1BBBC55C98D6.html

Acknowledge Triggered Alarms

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.monitoring.doc/GUID-2A9C547B-9D33-487D-8C1C-14F39D37ED1E.html

Preconfigured Alarms:

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.monitoring.doc/GUID-82933270-1D72-4CF3-A1AF-E5A1343F62DE.html

Set Alarm Rules

https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.monitoring.doc/GUID-16757D5E-94CD-4224-A65C-3050329B591D.html

Objective 7.15

VCP6-DCV Cert Guide: Chapter 7 – Objective 4.1

create baselines

pp18 vSphere 6.7 Install admin guide

apply baselines

pp20 vSphere 6.7 Install admin guide

notifications

pp70 vSphere 6.7 Install admin guide

download

vSphere 6.7 Install admin guide

download updates and related media pp 16

import ESXi images pp 17

remediate

pp22 vSphere 6.7 Install admin guide