NAKIVO®

The Complete Guide to VMware Clustering

Table of Contents

Overview ... 4
Cluster: Definition and Types ... 4
- What Is High Availability Cluster? ... 5
- What Is Distributed Resource Scheduler Cluster? ... 5

How to Create a VMware Cluster? ... 6
Installing VMware ESXi Server ... 7
- Requirements ... 8
- ESXi Installation Process ... 8
- Creating a New Datasatore ... 17

Installing Active Directory Domain Controller ... 21
- Installing a Server Role ... 21
- Post-Deployment Configuration ... 27

Installing and Setting up vCenter Server ... 35
- Requirements ... 35
- Methods to Install VCSA ... 36
- How to Install vCenter ... 36
- ESXi Hosts and vCenter Connection Scheme ... 52
- Adding Items to the vCenter Inventory ... 54
  - Adding a new datacenter ... 54
  - Adding ESXi hosts to the datacenter ... 56

ESXi Networking and Storage Configuration ... 61
- Using Shared Storage in vSphere ... 63
- Shared Storage Configuration ... 63
  - Adding a virtual switch ... 63
  - Adding an iSCSI adapter ... 71
  - Creating a datastore ... 75
  - Adding a shared datastore on other ESXi hosts ... 79
  - Datastore Configuration in vCenter ... 81
  - Troubleshooting ... 83
  - Heartbeat network ... 84
  - Fault Tolerance logging network ... 84

How to Create and Configure a Cluster 89
Creating a New Cluster 89
Adding Hosts to the Cluster 91
How to Create a DRS Cluster 94
Automation 95
Additional options 96
Power management 97
Advanced options 98
Affinity rules 100
How to Configure a HA Cluster 109
Failures and responses 111
Admission Control 118
Heartbeat Datastores 119
Advanced Options 120
Proactive HA 121
Fault Tolerance: Purpose and Setup 123
Fault Tolerance requirements 123
Fault Tolerance limitations 123
How to enable Fault Tolerance 124
How to turn off and disable Fault Tolerance 130
Removing Hosts from the Cluster 130
Conclusion 133

Overview

Information technologies are evolving at high speed and are used in a growing number of industries. While IT solutions provide automation and ensure rational usage of natural and human resources, their hardware requirements gradually increase. Even a powerful server can be overloaded with multiple computing tasks. For better performance and reliability, servers can now be connected to each other over the network. For this purpose, clustering technologies are widely used. This e-book explains what a cluster is, what issues you can resolve by using clustering and how to create and deploy clusters in a VMware environment.

What Is a Cluster: Definition and Types

A **cluster** is a group of independent servers that communicate with each other over a network and can act as a single system. The servers forming a cluster are called nodes or members, and they are fine-tuned to perform the same tasks under the control of a specific piece of software. Any cluster consists of at least two nodes.

There are three common types of clusters:

- High-Performance Computing clusters
- High Availability clusters
- Load Balancing clusters

High-Performance Computing (HPC) clusters are also called parallel clusters. This type of cluster provides a single system image. This means that an application can be executed on any of the servers within the cluster. HPC clusters are used to execute compute-intensive and data-intensive tasks by running a job on multiple nodes simultaneously, thus enhancing application performance.

High Availability (HA) clusters are also called failover clusters. This type of cluster provides software operation with minimum downtime. Redundant storage, software instances, and networking provide continued service when a system component fails. HA clusters usually use a heartbeat private network connection to monitor the health and status of each node in the cluster.

Load Balancing (LB) clusters ensure better performance. In LB clusters, tasks are distributed between nodes to load hardware more rationally and avoid overloading individual servers if enough computing resources are available.

In VMware vSphere, you can deploy two of the above types of clusters that work on the virtual machine layer: HA cluster and LB cluster, which is called Distributed Resource Scheduler (DRS) in the context of VMware vSphere.

What Is a High Availability Cluster?

A High Availability (HA) cluster supports the migration of virtual machines from one ESXi host to another in case of a failure. Two or more ESXi servers of the same network with shared storage are united into a logical group called a pool. When one of the ESXi servers fails, the virtual machines that were running on this host are started on another ESXi server within the cluster. Powering on and loading these virtual machines may take some time. Hence, the idle time of a virtual machine is equal to the time required for loading this virtual machine.

After an ESXi server is added to a cluster, a special agent called the Fault Domain Manager (FDM) is automatically installed. This utility monitors signals called heartbeats from other ESXi hosts in the cluster and, by default, communicates with the vCenter Server once every second. If only one virtual machine fails, this VM is restarted on the same ESXi server. The type of action depends on the type of failure detected and may be set in the preferences.

The first five hosts added to the cluster are primary, and all subsequent hosts are secondary. If one of the primary hosts is removed from the cluster, a secondary host becomes one of the primary ones.

The main HA cluster features are:

- **Host monitoring** helps monitor each ESXi host in the cluster and ensure that this server is running. ESXi hosts exchange network heartbeats in the cluster. If an ESXi host fails, virtual machines can be restarted on another host.
- **Admission control** controls the policy used by an HA cluster for reserving resources to ensure failover capacity within the cluster.
- **VM monitoring** monitor each virtual machine in the cluster with VMware Tools heartbeats to ensure this VM is running. A virtual machine that fails can be restarted.
- **Datastore heartbeating** uses datastores to monitor hosts and virtual machines when the management network fails. This feature also reduces the probability of false restart and false migration.
- **Fault tolerance** allows avoiding VM downtime by running a VM replica on another ESXi host within the cluster.

We will explain these features and their configuration in detail below.

What Is Distributed Resource Scheduler Cluster?

A Distributed Resource Scheduler (DRS) cluster supports distributing computing resources between hosts, depending on the performance required by virtual machines and the availability of free ESXi host resources. DRS checks the performance of your virtual machines and makes placement decisions. This means that DRS decides to which host within the cluster to migrate the particular VM automatically or manually after the notification.

Some virtual machines can be idle and “wake up” when they are required to execute an important task, using the host’s CPU, memory, and network. This can influence the DRS decision about moving these VMs to another host with more free resources available. The feature helps reduce administration effort spent on monitoring and maintaining the infrastructure.

The main DRS cluster features are:

- **Load balancing** allows performing or recommending VM migrations between ESXi hosts to balance the load depending on the settings.
- **Power management** supports VM migration from one ESXi host to another if there are enough free resources and sets the standby power mode for the source ESXi server.
- **Affinity rules** allows controlling the placement of virtual machines to hosts by assigning rules.

How to Create a VMware Cluster

The following are hardware and software requirements for a VMware cluster:

- Availability of at least two ESXi servers with unique host names, static IP addresses, and processors of the same vendor and family using the equivalent instructions to reach the maximum compatibility.
- All hosts within the cluster should be attached to shared storage, such as Network Attached Storage (NAS) or Storage Area Network (SAN) via Fibre Channel, Internet Small Computer System Interface (iSCSI), or Network File System (NFS) protocols. Virtual Machine File System (VMFS) volumes must be used for block-based shared storage. Also, there must be enough free storage space.

> **Note**
> Choose the NAS or SAN from authorized vendors that meet your requirements for the production environment. Set up the storage according to the manufacturer’s documentation. You can use more than one NAS or SAN to create a VMware cluster.

- All volumes on the ESXi hosts must use the same volume names.
- A machine with a DNS server has to be installed.
- A machine with vCenter has to be installed.

Creating a VMware cluster requires taking the following steps:

1. Installing VMware ESXi Server
2. Installing DNS server and Active Directory Domain Controller (recommended)
3. Installing and setting up vCenter Server

4. Setting up a shared datastore
5. Connecting hosts in clusters
6. Configuring the network for the cluster
7. Configuring HA and DRS clusters
8. Configuring Fault Tolerance (optional)

We use the following configuration to explain VMware vSphere cluster installation and configuration in our environment:

- Domain name: *domain1.net*
- ESXi1 IP address: *10.10.10.46*
- ESXi1 hostname: *ESXi7-1*
- ESXi2 IP address: *10.10.10.82*
- ESXi2 hostname: *ESXi7-2*
- vCenter IP address: *10.10.10.18*
- vCenter hostname: *vCenter7*
- vCenter VM name: *vCenter7*
- Network: *10.10.10.0/24*
- Gateway/DNS server *10.10.10.2*
- Secondary DNS server: *(optional)*
- NAS IP address: *192.168.105.228*
- Storage network: *192.168.105.0/24*
- Management network: *10.10.10.0/24* or *192.168.101.0/24*. Later we can configure different IP addresses for the management network and VM network.

Installing VMware ESXi Server

VMware ESXi server is an enterprise-class type-1 hypervisor with its own kernel (VMkernel). It runs directly on server hardware and does not require the installation of an additional underlying operating system.

ESXi is highly reliable and includes an ultra-thin architecture, which is not dependent on a general-purpose operating system. A smaller code-base presents a smaller vulnerability area and performs quick installation and booting.

As the virtualization server, ESXi is the most important component of a vSphere environment. All virtual machines run on the ESXi server. But the virtual disks of virtual machines can be stored on internal data storage located directly on the ESXi server (such as hardware SAS

RAID) or a shared external data storage. You can access the ESXi server remotely via the Secure Shell (SSH) client, VMware Command Line Interface (CLI), or vSphere Client. SSH access is disabled by default.

Requirements

The minimum hardware requirements for ESXi are as follows:

- A 64-bit processor with at least two cores.
- At least 8 GB of RAM to take full advantage of ESXi features and run virtual machines in typical production environments.
- Support for hardware virtualization (Intel VT-x or AMD-V).
- Two or more 1-Gigabit or 10-Gbit Ethernet controllers to provide network redundancy for the cluster.
- Small Computer System Interface (SCSI) disk, Serial Attached SCSI (SAS) disk, or a local, non-network, Redundant Array of Independent Disks (RAID) Logical Unit Number (LUN) with unpartitioned space.
- Serial ATA (SATA) disks connected through supported SAS controllers or supported on-board SATA controllers that are considered remote and not local. These disks are not used as a scratch partition by default because they are seen as remote.
- At least 32 GB of free persistent disk space. Unlike previous ESXi versions, ESXi 7 must not be installed on a USB flash drive or SD memory card.

ESXi installation process

To install the ESXi server, do the following:

1. Insert the installation disc into the optical drive, select this disc as the first boot device in BIOS, and boot from this disc.

   You may also write the installation image to a USB flash drive and use this USB drive as the boot device.

2. Select the Installer in the boot menu if this option has not been applied automatically (see *Figure 1.1*).

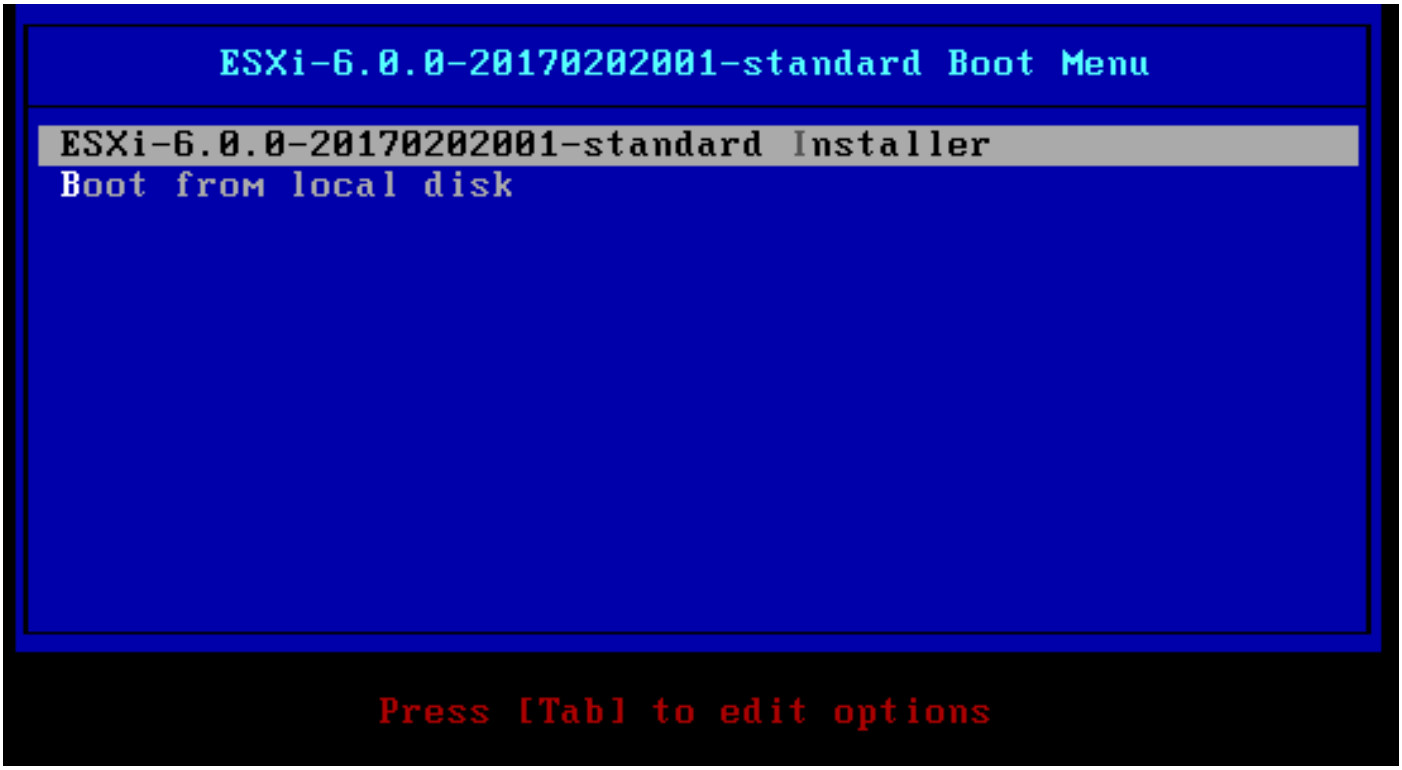


Figure 1.1

Note
If your system hangs up at the user *loaded successfully* stage (see *Figure 1.2*), this may be due to insufficient RAM. Check the amount of memory available. Press **Alt+F12** to view the details.

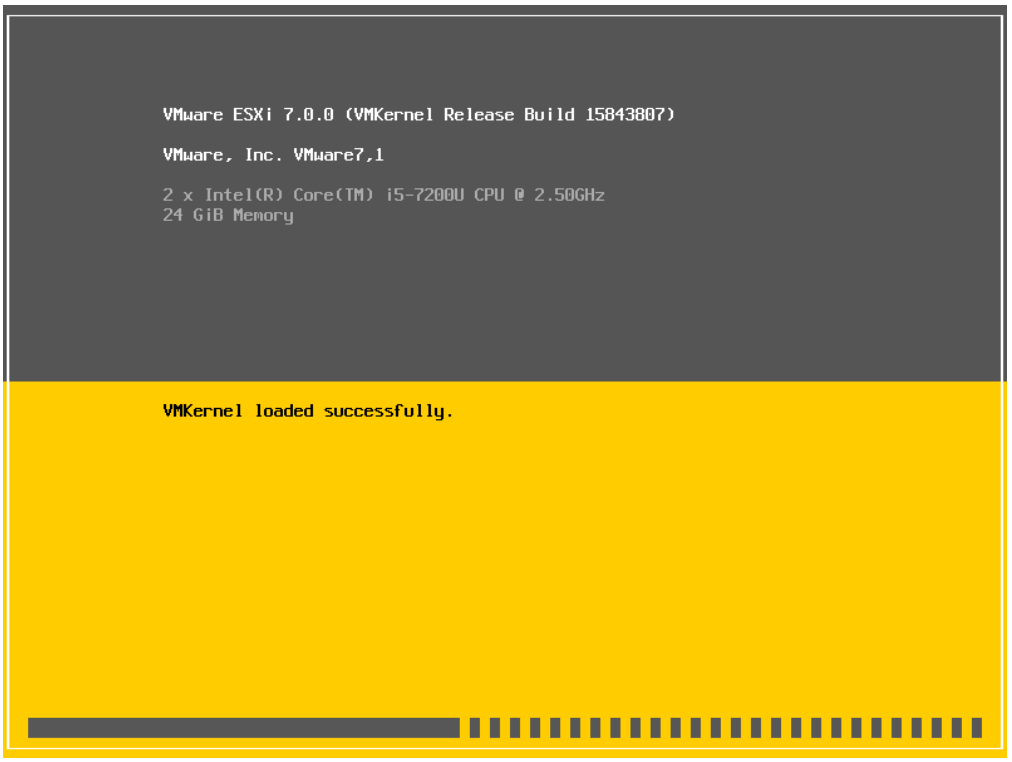


Figure 1.2

3. If everything is OK, the welcome installation screen appears (see *Figure 1.3*). Hit **Enter** to continue.

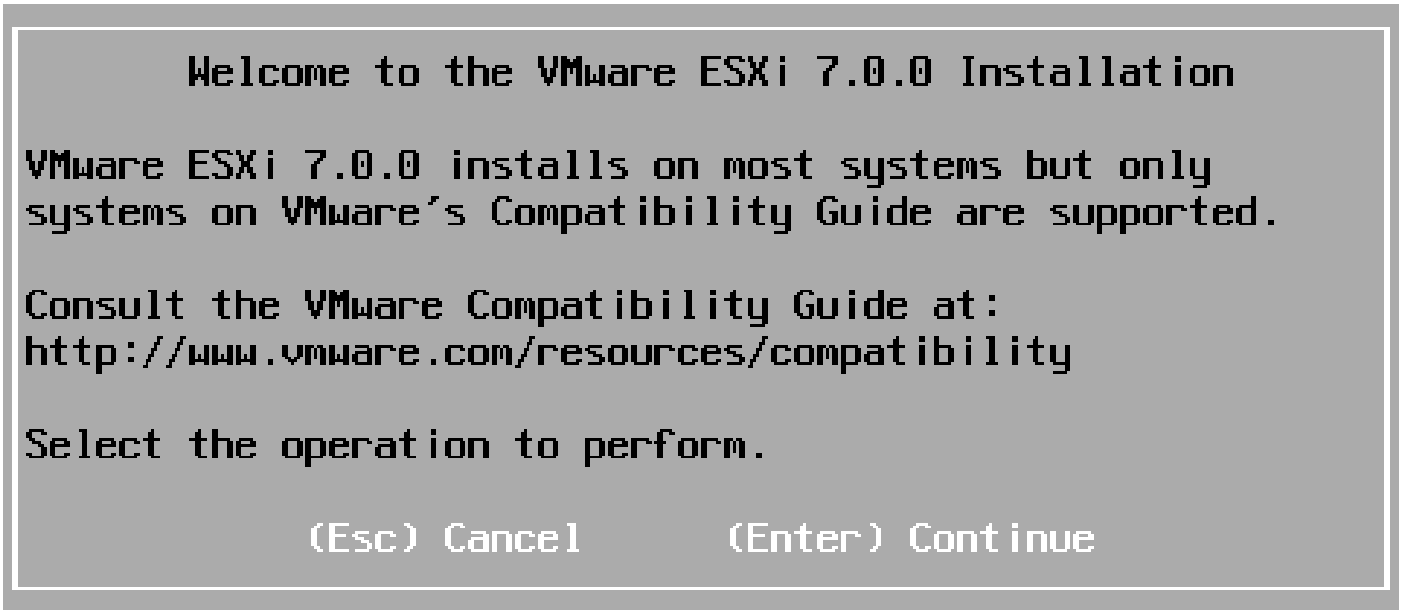


Figure 1.3

4. Select a disk to install ESXi (see *Figure 1.4*).

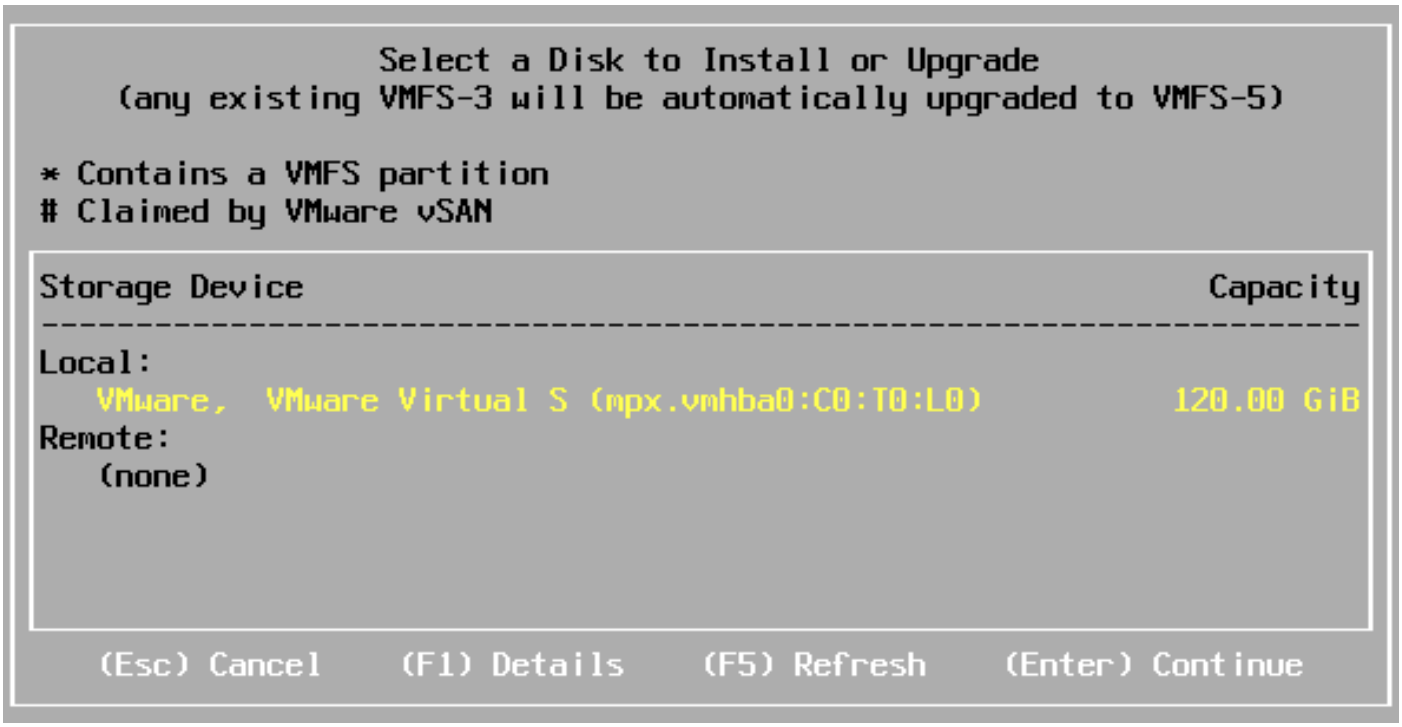


Figure 1.4

5. Choose your keyboard layout.
6. Enter a root password (see *Figure 1.5*).

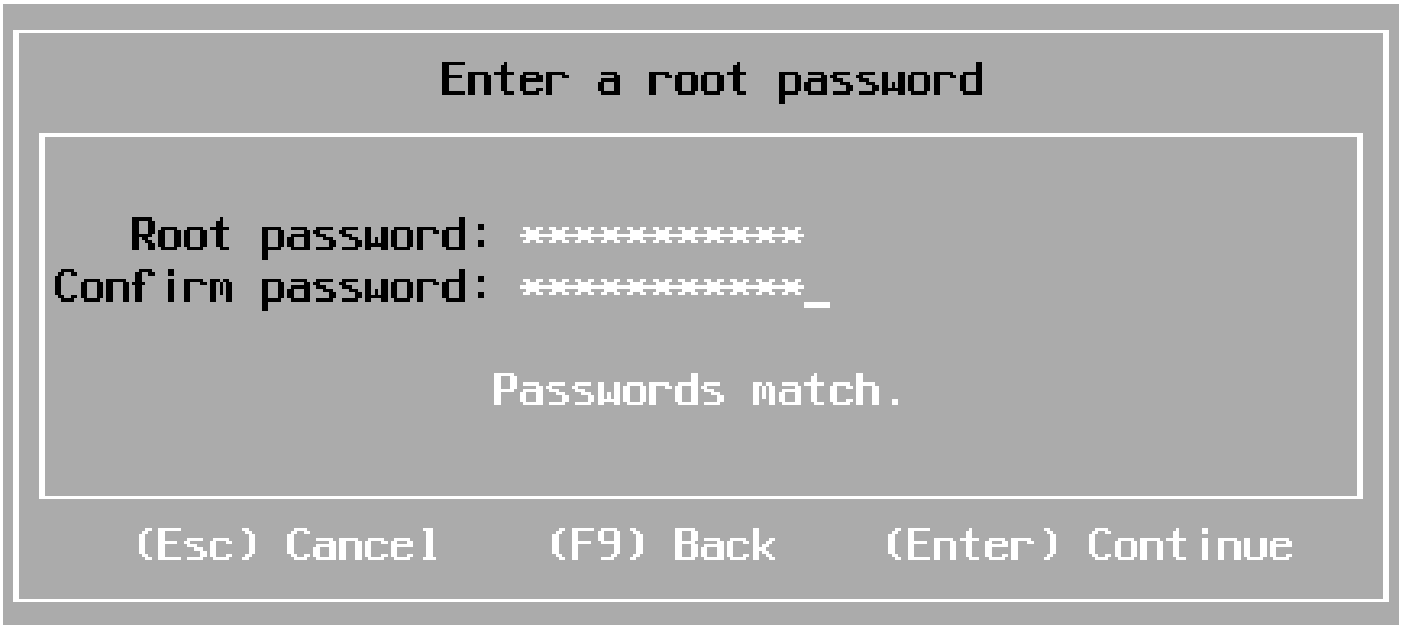


Figure 1.5

7. Then wait for system scanning to complete (see *Figure 1.6*).

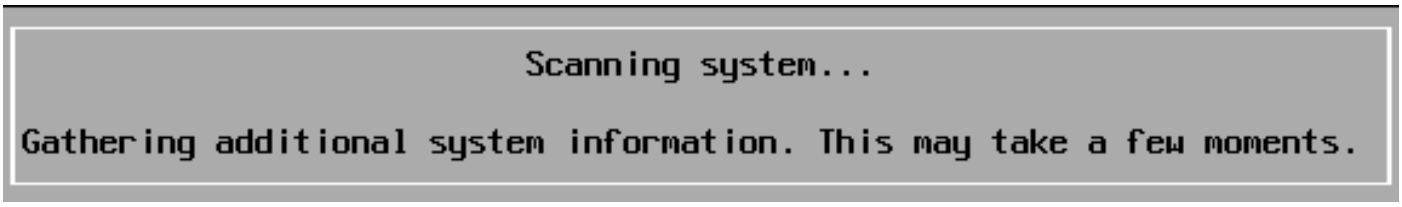


Figure 1.6

Note
If there are fewer than 2 CPU cores, an error message appears (see *Figure 1.7*).

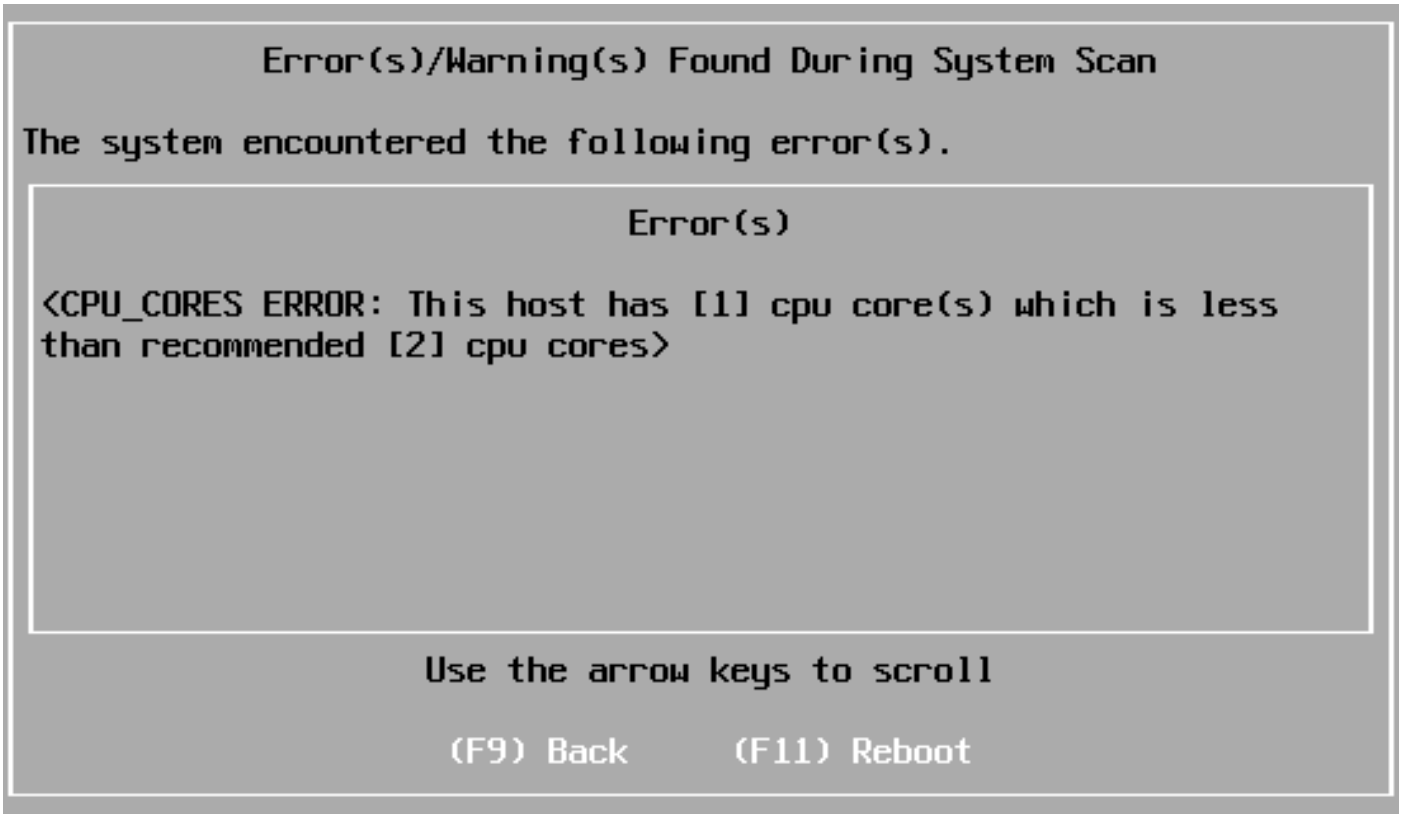


Figure 1.7

If everything is OK, the confirmation message is displayed (see *Figure 1.8*).

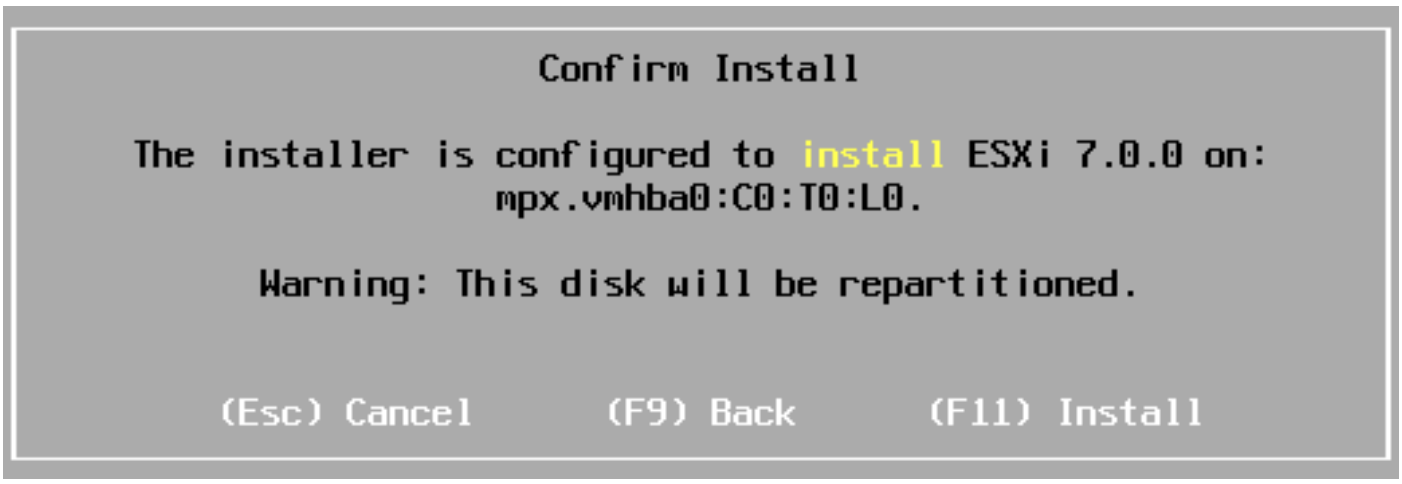


Figure 1.8

8. Press **F11** to install and wait for the installation progress to complete (see *Figure 1.9*).

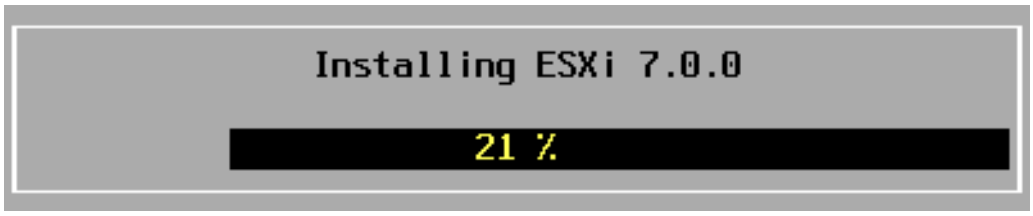


Figure 1.9

9. When you see the *Installation Complete* message, press **Enter** (see *Figure 1.10*).

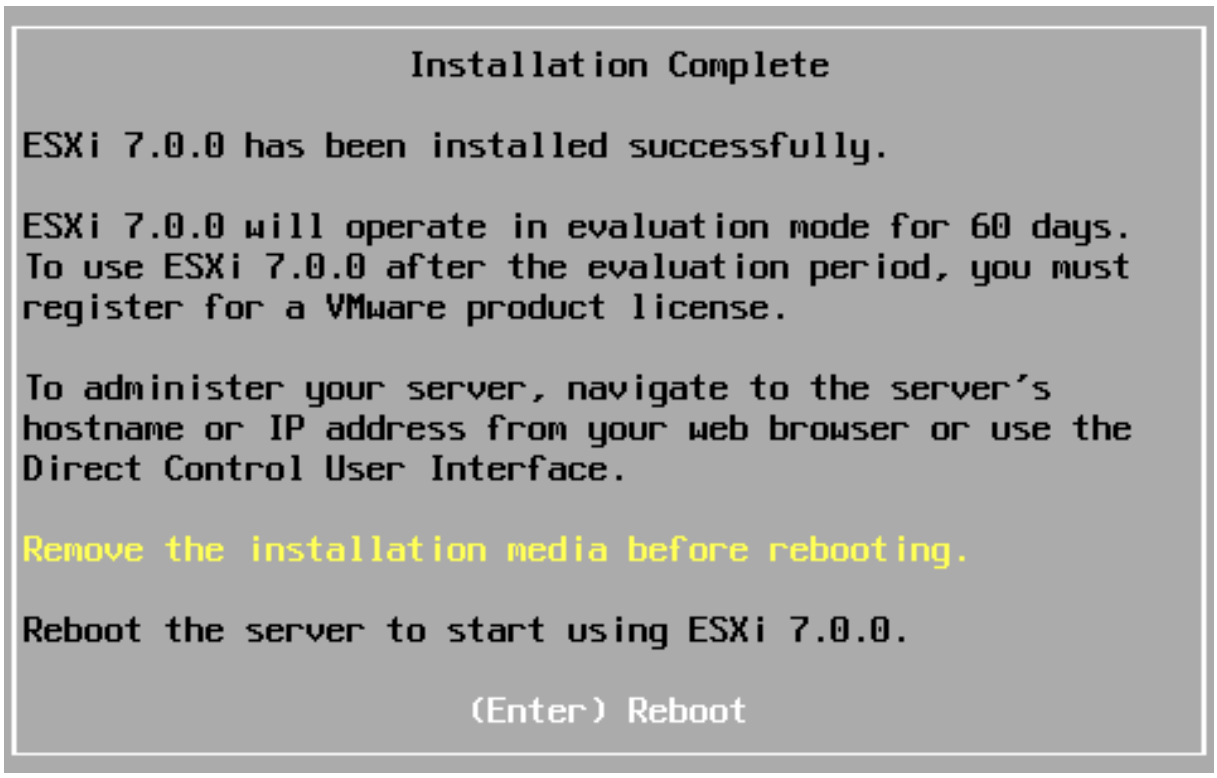


Figure 1.10

Wait for the server to reboot (see *Figure 1.11*).

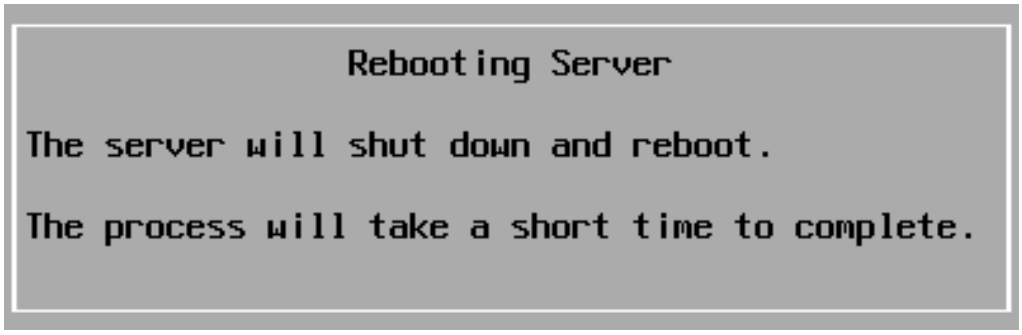


Figure 1.11

10. After rebooting the server, log in to ESXi (see *Figure 1.12*).
11. Press **F2** to customize settings in the ESXi DCUI (Direct Console User Interface).

Figure 1.12

12. Select **Configure Management Network** in the menu, and set up a hostname along with the IP address manually, for example, *10.10.10.46* (see *Figure 1.13*). Set a hostname, for example, *ESXi7-1*. Disable IPv6 if you don't use this protocol.

Figure 1.13

13. Press **Y** (Yes) to confirm network changes (see *Figure 1.14*).

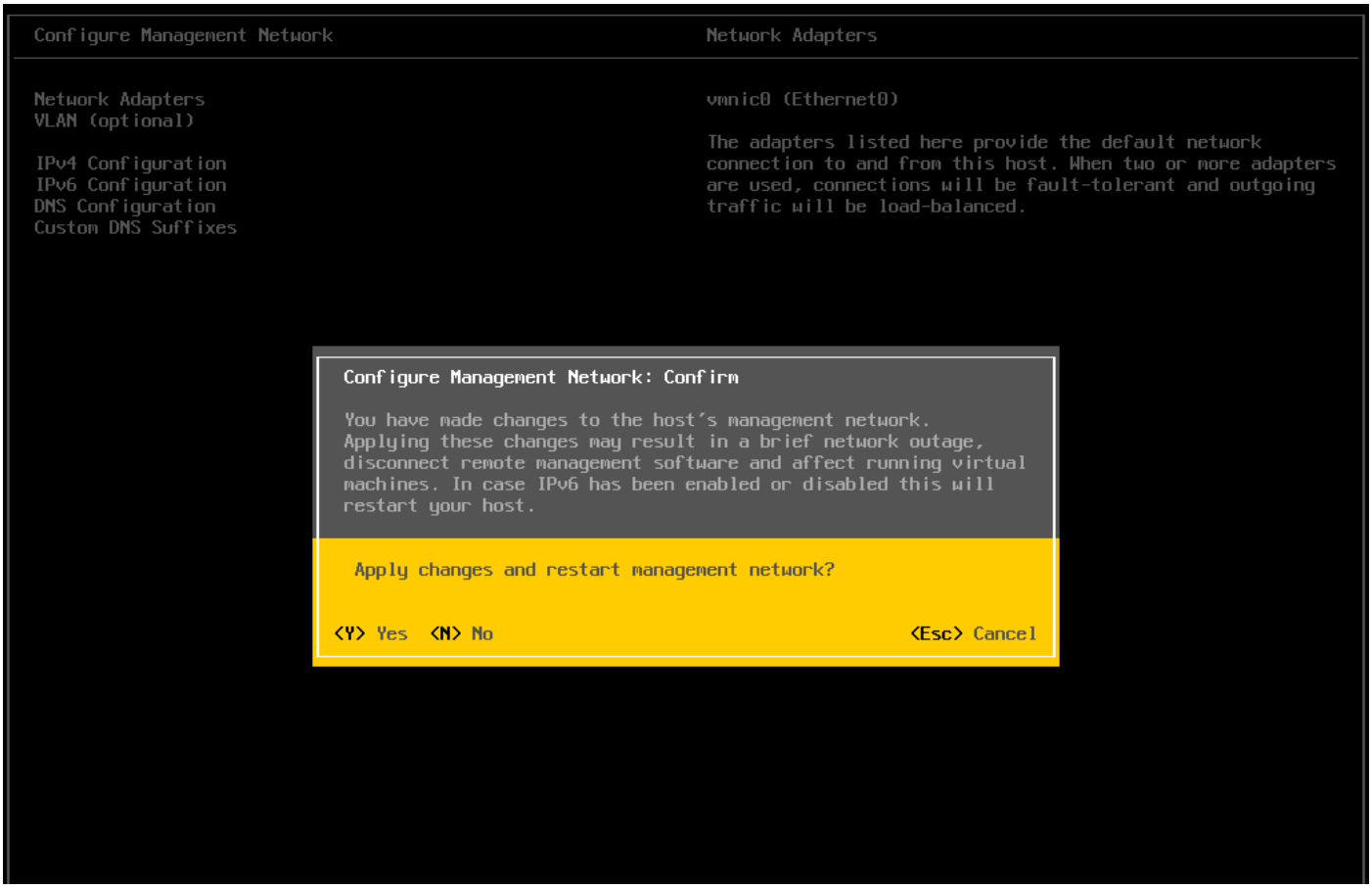


Figure 1.14

14. Now, you can open VMware Host Client in a web browser to manage your ESXi host settings. Note that a standalone vSphere Client that is installed in Windows is no longer supported. Open a web browser and enter the URL of your ESXi server (see *Figure 1.15*).

    Ignore the certificate warning.
15. Enter the username and password you have set during the ESXi installation to log in to vSphere Client.

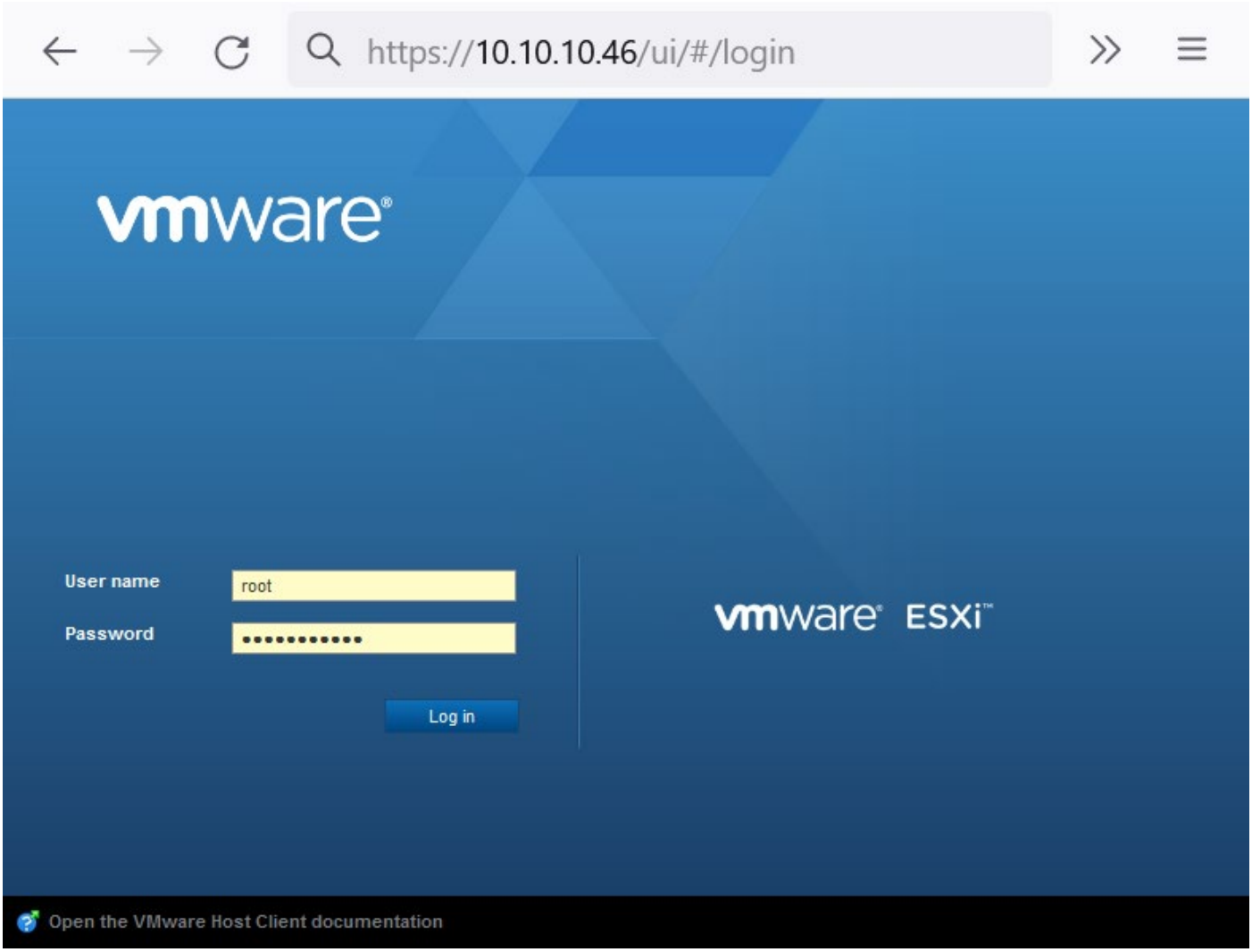


Figure 1.15

This is how the vSphere Client interface looks like (see *Figure 1.16*):

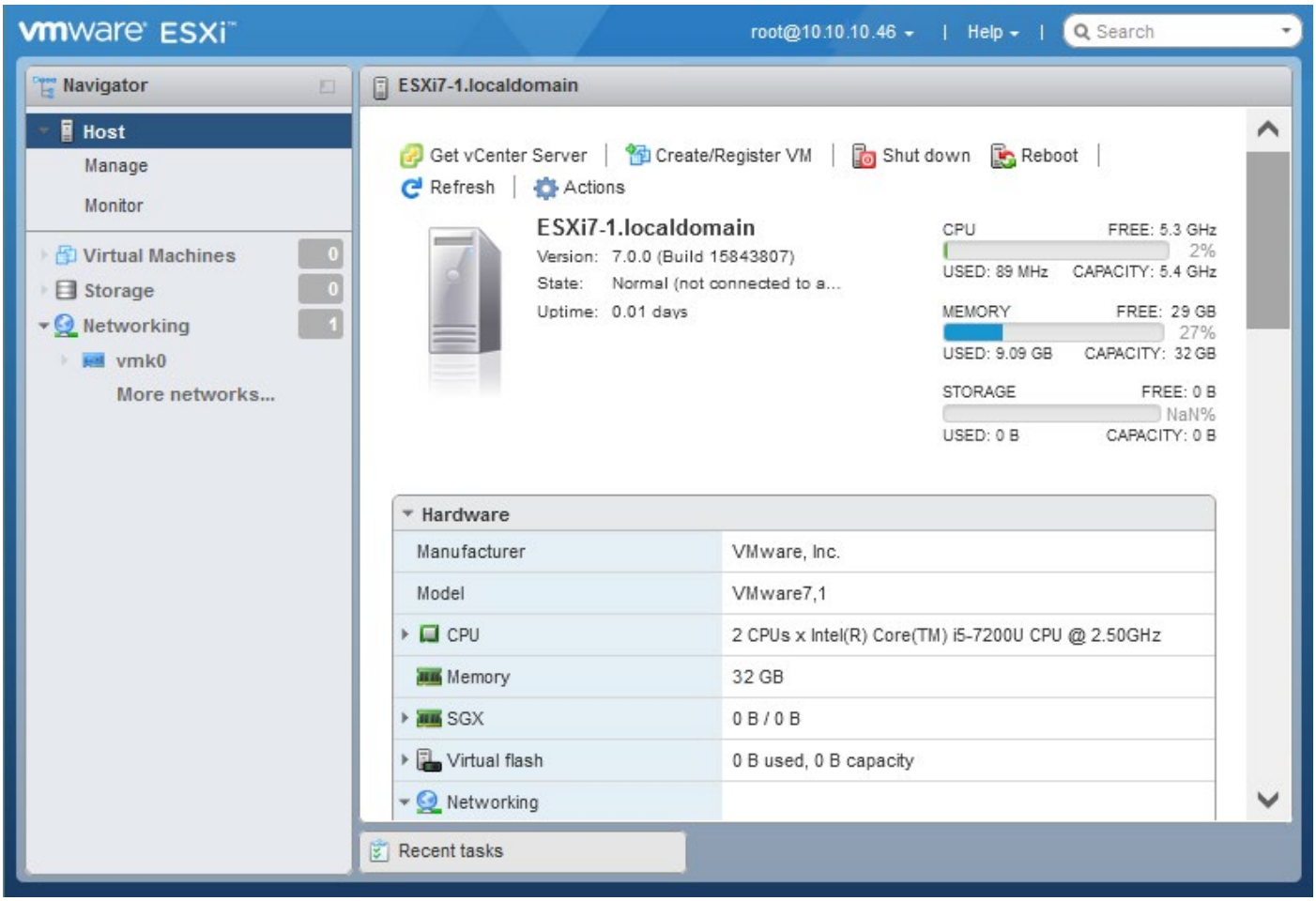


Figure 1.16

Creating a new datasatore

You can now create a new datastore. You can attach additional disk drives (storage arrays to create new datastores). Datastores are needed to store VMs, including vCenter, installation images, and other files.

> **Note**
> If you use a large disk to install ESXi (larger than 142 GB), one datastore is created automatically with an ESXi partition layout.

1. Go to **Storage** in the *Navigator* pane of VMware Host Client.
2. Click **New datastore** in the **Datastores** tab (see *Figure 1.17*).

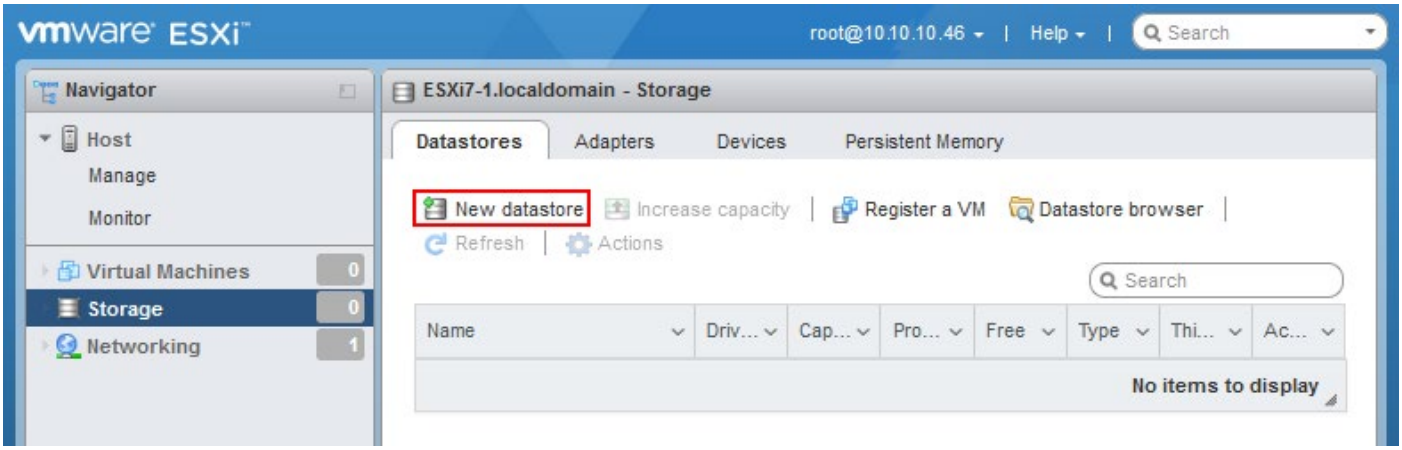


Figure 1.17

3. The *New datastore* wizard opens.

   1. Select creation type. Click **Create a new VMFS datastore**. Hit **Next** at each step to continue (see *Figure 1.18*).

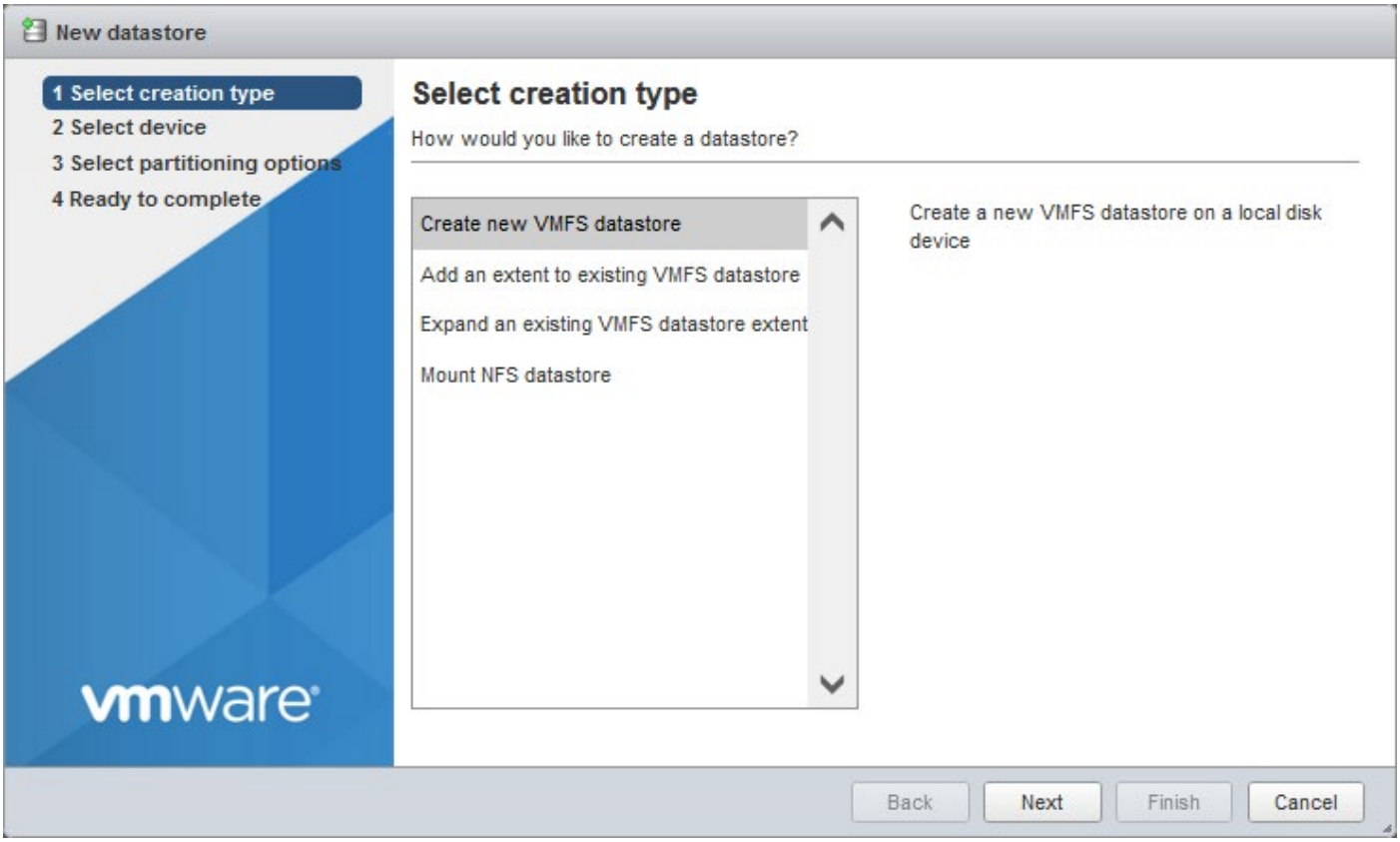


Figure 1.18

   2. Select device. Select a LUN, disk, or disk array to create a new datastore. Enter a datastore name (see *Figure 1.19*).

Figure 1.19

3. Select partitioning options. We use the full disk in this example and VMFS 6 (see *Figure 1.20*).

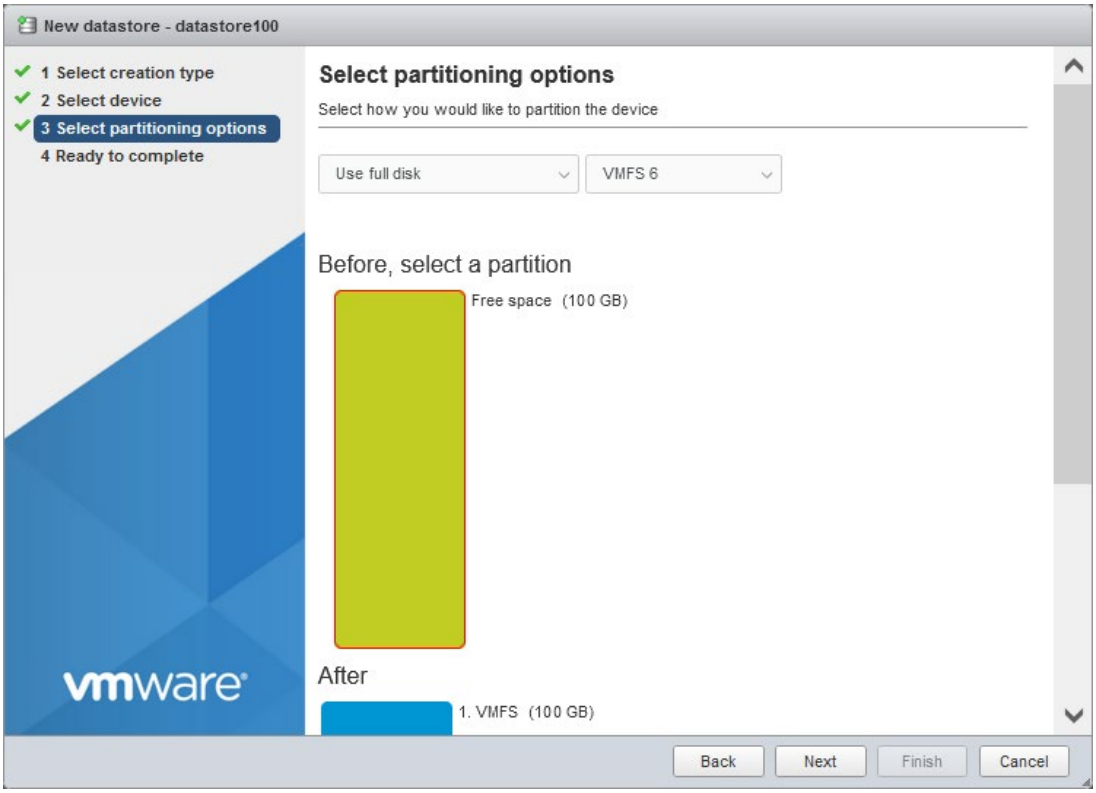


Figure 1.20

4. Ready to complete. Check your datastore configuration options and hit **Finish** (see *Figure 1.21*).

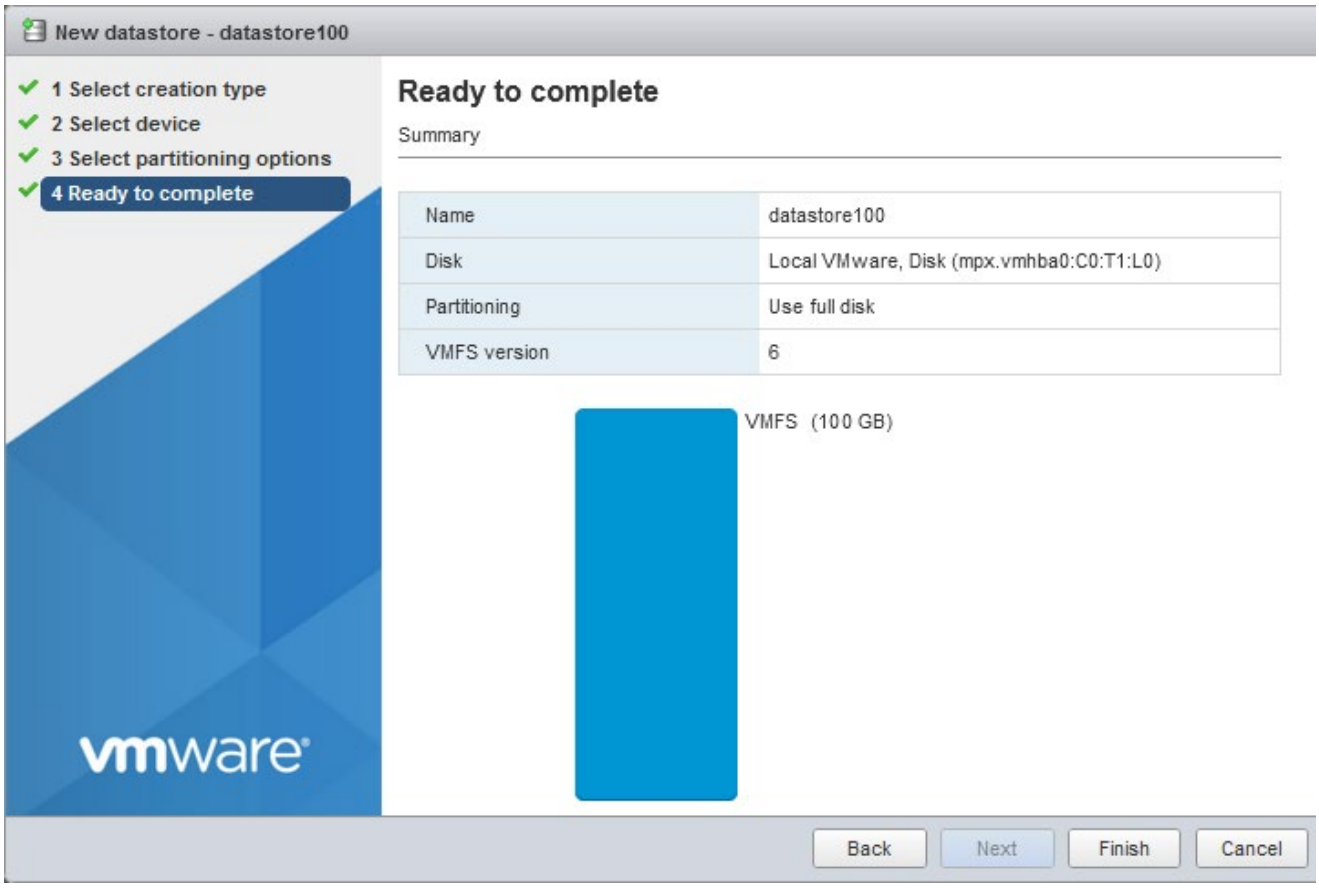


Figure 1.21

A confirmation message is displayed. If you use an empty disk for the new datastore, click **Yes** without any concern (see *Figure 1.22*).

Figure 1.22

Now you have one ESXi host installed. Use the 60-day full-featured trial mode or enter a license key. Install the second ESXi host similarly.

Below we will explain how to configure a shared datastore that is required for a cluster.

Installing an Active Directory Domain Controller

The recommendation is to have an Active Directory Domain Controller (AD DC) for integration with vCenter Server. A Domain Controller is not a requirement to deploy a vSphere environment, but a Domain Name System (DNS) Server is required to run vCenter and resolve hostnames to IP addresses and vice versa.

A Domain Controller is used for centralized management of Windows environments and other integrated environments, including authentication. A Domain Controller is the main component of the Active Directory Domain Services server role, which needs to be installed in this case.

Note 1
Go to the official VMware website and check the compatibility table of Microsoft Windows Server versions and domain functional level with the vCenter version you are using. In the example below, we use Windows Server 2019 and vCenter 7.0.

Note 2
You can install a DNS server and domain controller on a physical server or a virtual machine. After configuring a cluster, you can migrate a VM running Active Directory Domain Services to the cluster for high availability.

Windows Server 2019 and the same domain level are supported by the latest vCenter versions (6.7 and 7.0). You can use the latest Windows Server version to deploy a domain controller and select an older domain functional level. The Windows Server 2012 R2 domain functional level is supported by most vCenter versions.

- Windows Server to be used as domain controller can be installed as a physical server or virtual machine.
- You need to set a hostname and static IP address on Windows Server before deploying AD DC and DNS server roles.

Installing a Server Role

To install Active Directory Domain Services, do the following:

1. Open Server Manager in Windows Server 2019. Click **Start > Server Manage**r.
2. Click **Add roles and features** in the *Dashboard* of the *Server Manager* window (see *Figure 2.1*).

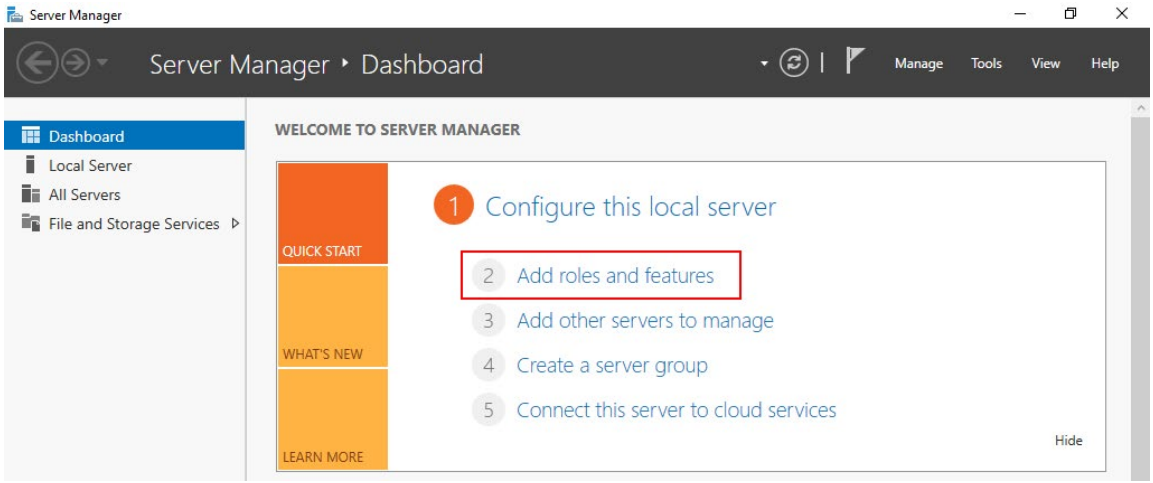


Figure 2.1

The *Add Roles and Features Wizard* opens.

3. **Installation Type.** At this step of the wizard, you need to select the **Role-based or feature-based installation** option. Click **Next** at each step of the wizard to continue (see *Figure 2.2*).

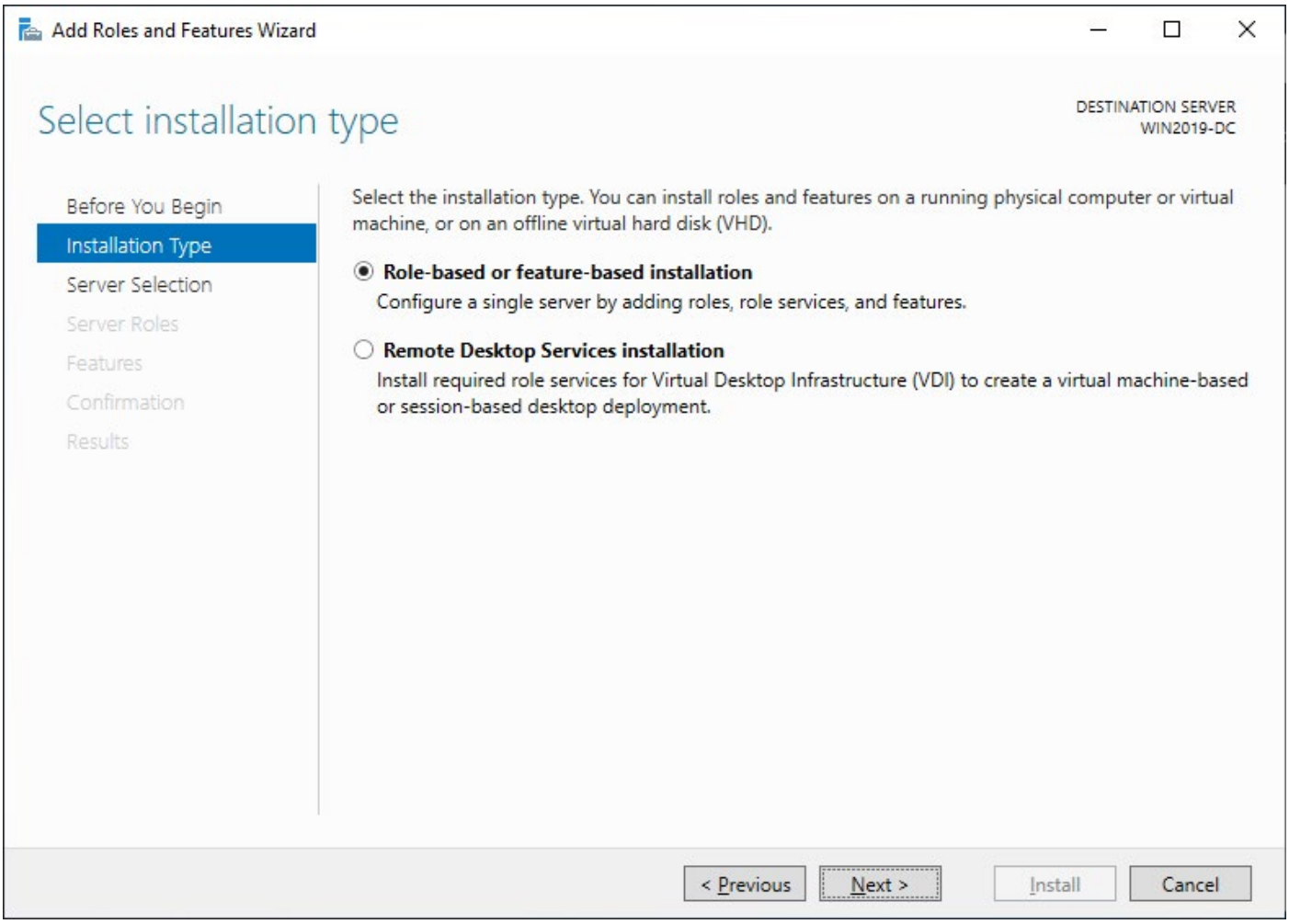


Figure 2.2

4. **Server selection**. Click **Select a server from the server pool** (this option is selected by default). Select your server (see *Figure 2.3*).

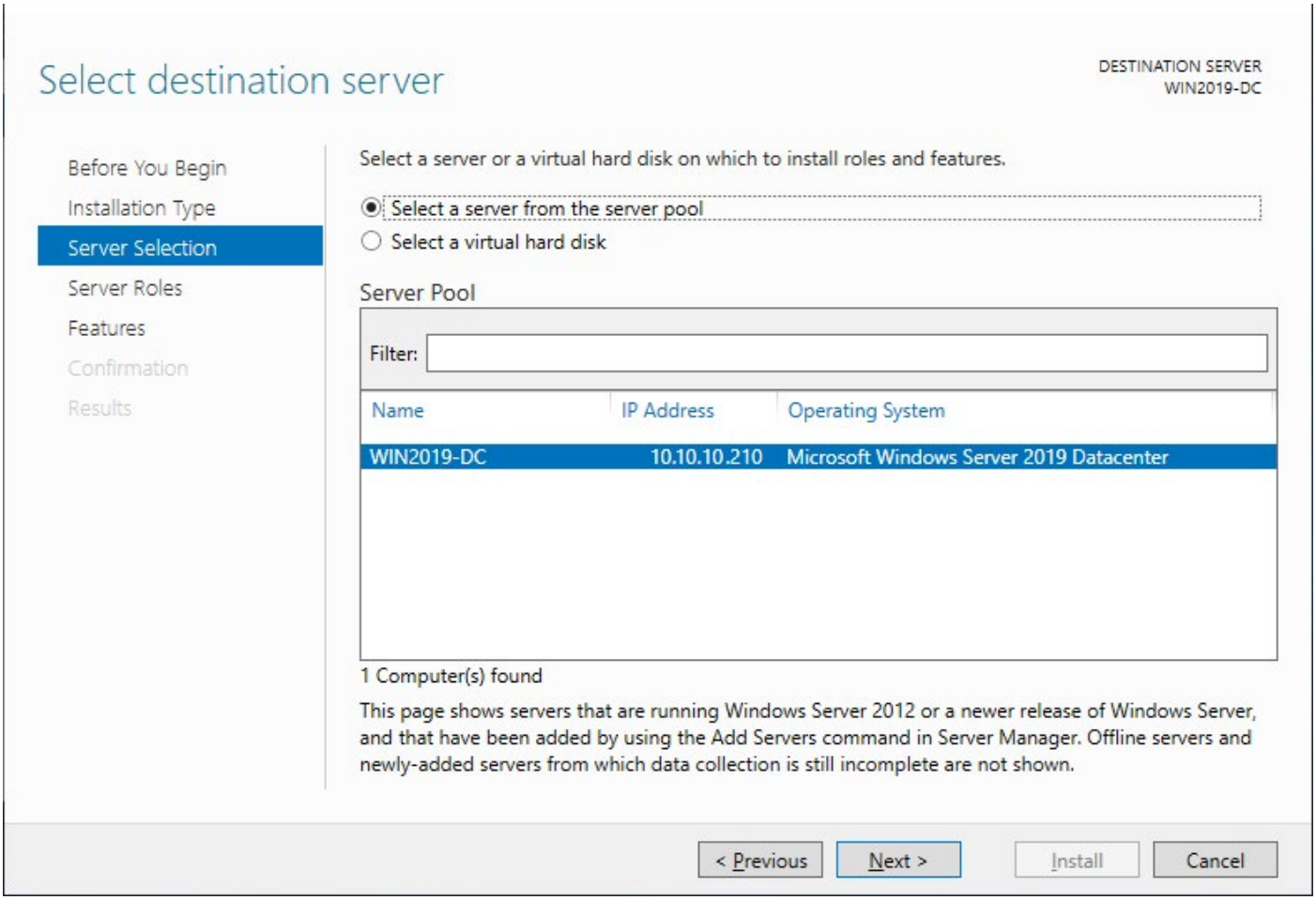


Figure 2.3

5. **Server Roles.** Select the two roles:

- Active Directory Domain Services
- DNS Server

If you don't select DNS Server, Windows will ask you to install this component with a Domain Controller (see *Figure 2.4*). In this walkthrough, we are installing a DNS server with Active Directory Domain Services.

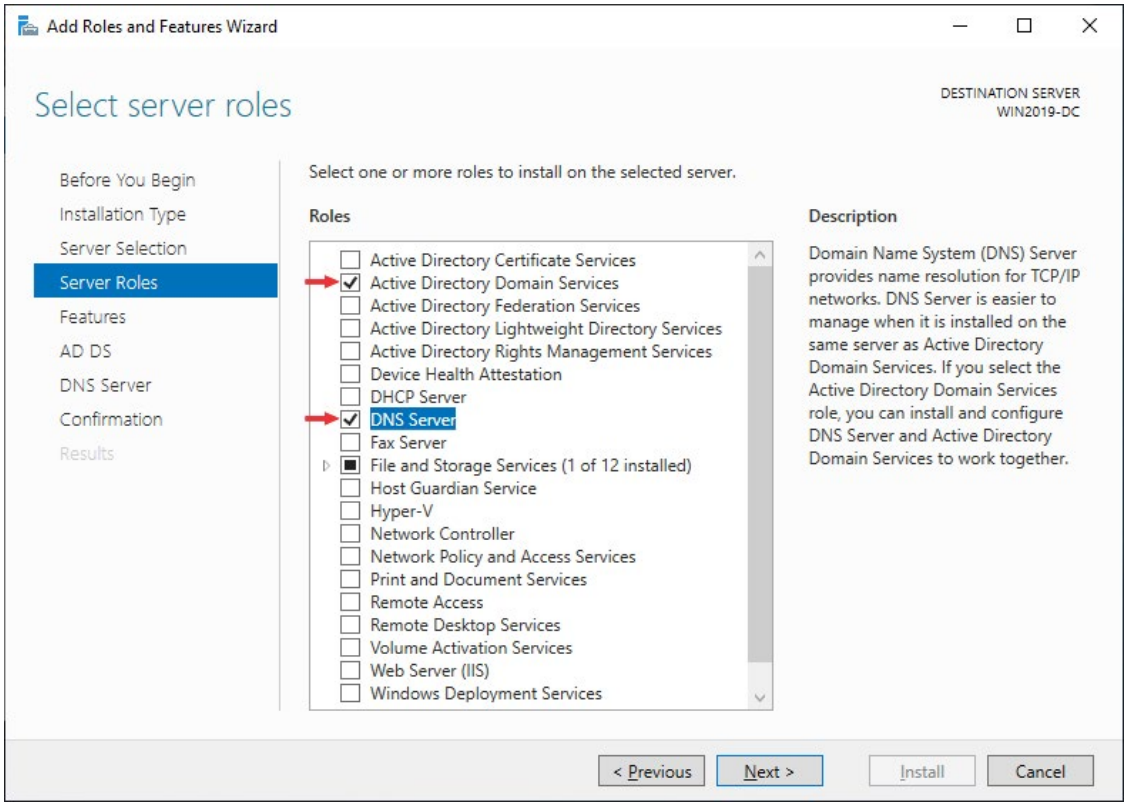


Figure 2.4

6. **Features**. You can leave features as is and don't make any changes at this step (see *Figure 2.5*).

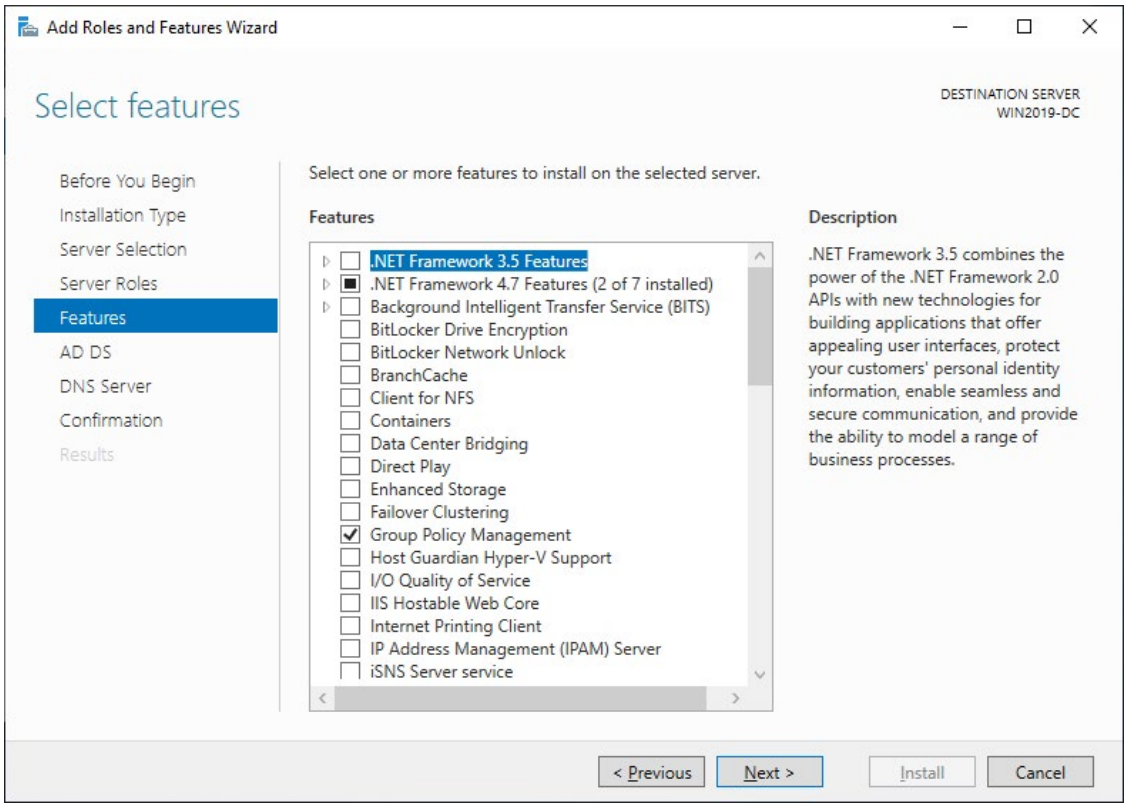


Figure 2.5

7. **AD DS**. There is nothing to configure on this screen. You can read the information about Active Directory Domain Services (see *Figure 2.6*).

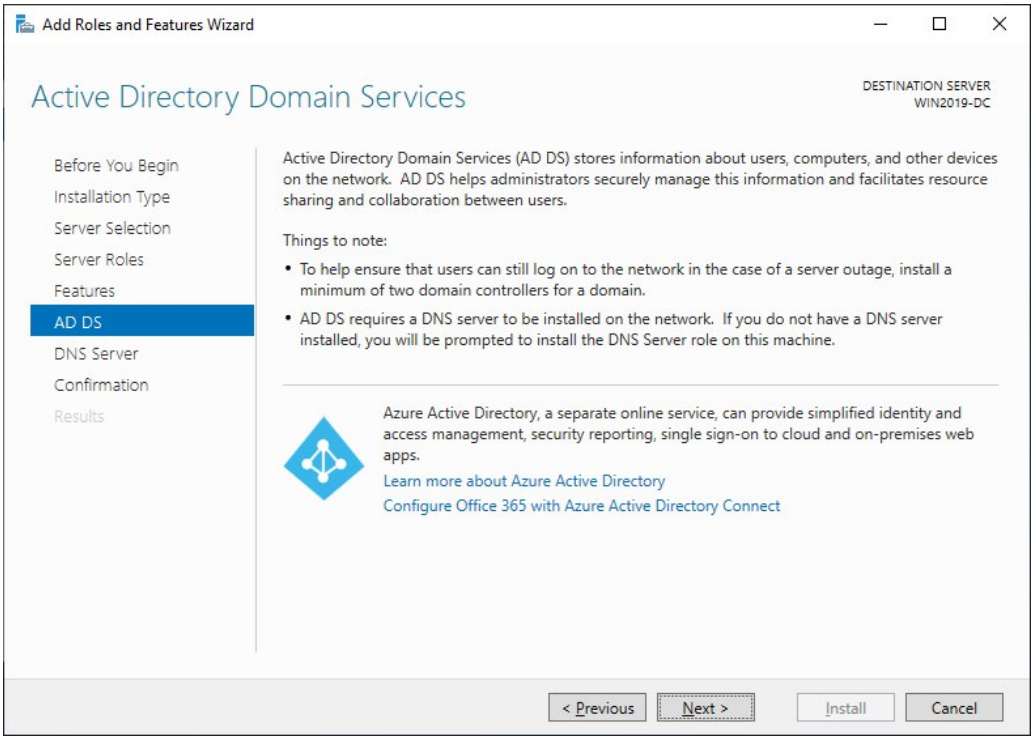


Figure 2.6

8. **DNS Server**. Information about DNS is displayed on this screen (see *Figure 2.7*).

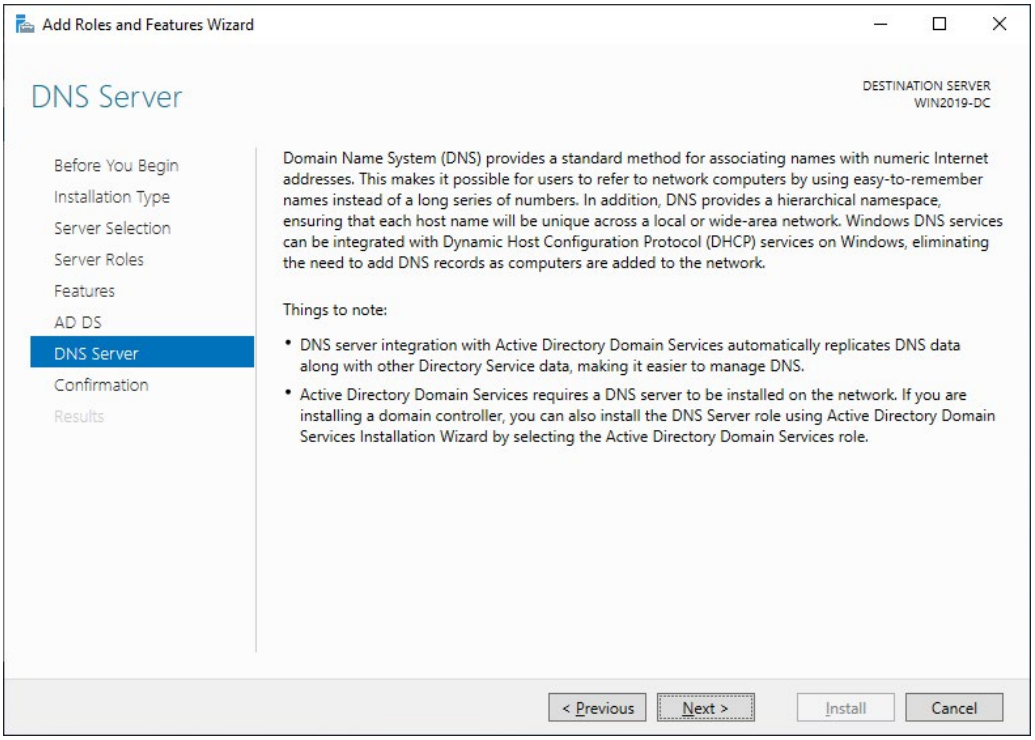


Figure 2.7

9. **Confirmation**. Check your installation selections. You can select the **Restart the destination server automatically if required** checkbox. Click **Install** to confirm your selections and start the installation (see *Figure 2.8*).

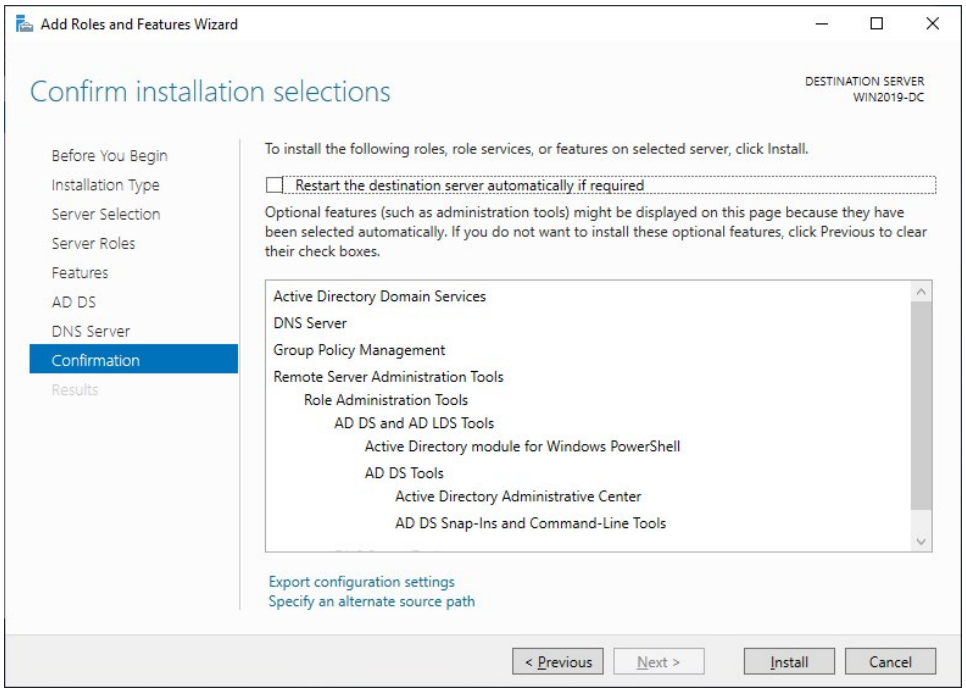


Figure 2.8

10. **Results**. Wait until the installation process of the selected roles is finished (see *Figure 2.9*) and then reboot the server.

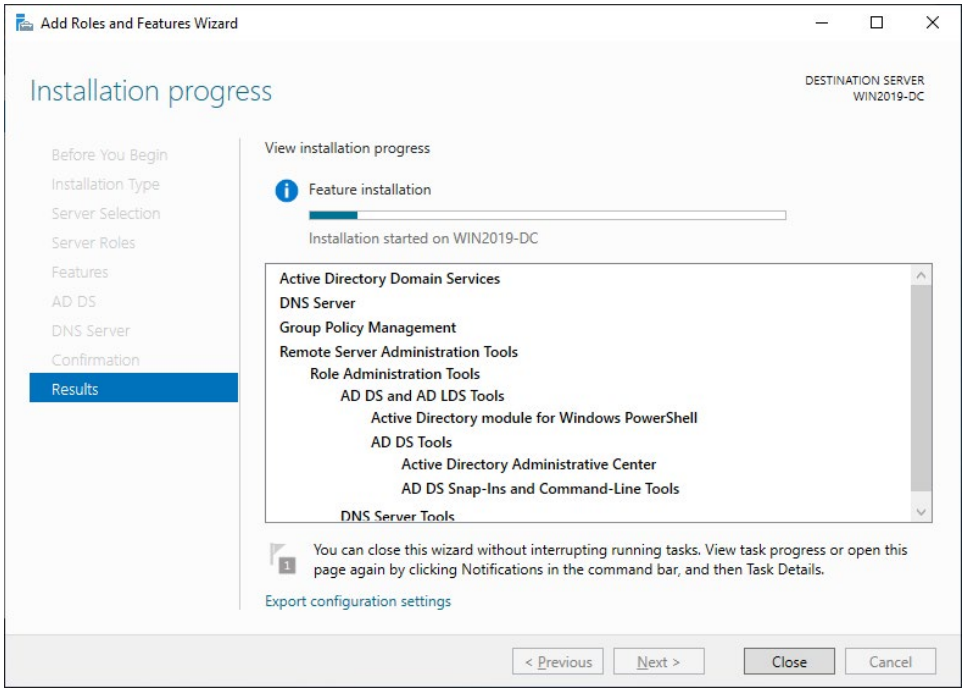


Figure 2.9

Post-Deployment Configuration

Open **Server Manager** and click the yellow triangle icon near the flag (see *Figure 2.10*).

Post-deployment configuration is required after installing the server roles. Click **Promote this server to a domain controller**.

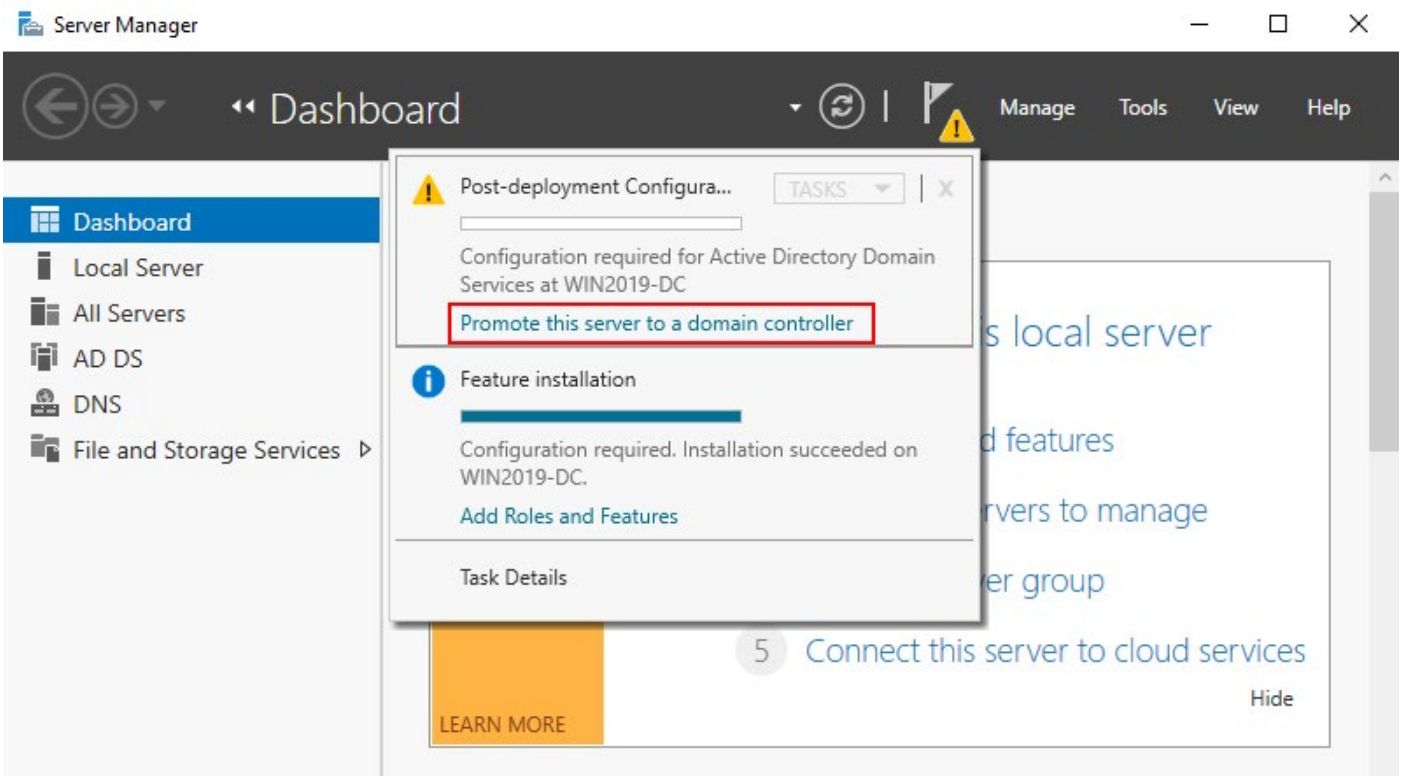


Figure 2.10

The *Active Directory Domain Services Configuration Wizard* opens.

1. **Deployment Configuration**. Select the deployment operation **Add a new forest**.

   Specify the root domain name. We use *domain1.net* in this example.

   Click **Next** at each step of the wizard to continue (see *Figure 2.11*).

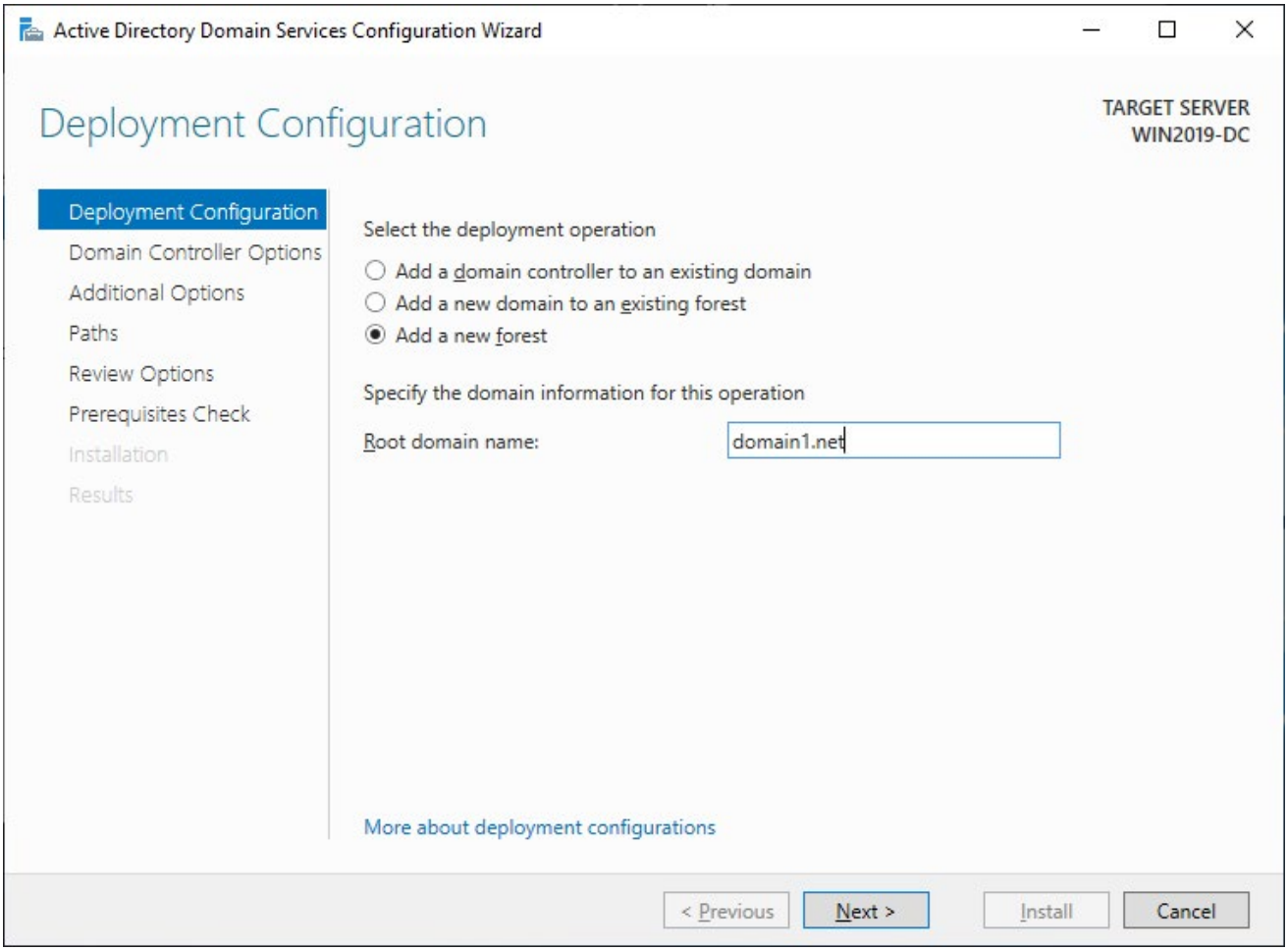


Figure 2.11

2. **Domain Controller Options.** Select the functional level of the new forest and root domain in the drop-down list:

   - Windows Server 2008
   - Windows Server 2008 R2
   - Windows Server 2012
   - Windows Server 2012 R2
   - Windows Server 2016

   We select **Windows Server 2016** in our tutorial.

   Specify domain controller capabilities:

   - Domain Name System (DNS) server
   - Global Catalog (GC)

   Type the Directory Services Restore Mode (DSRM) password and confirm this password (see *Figure 2.12*).

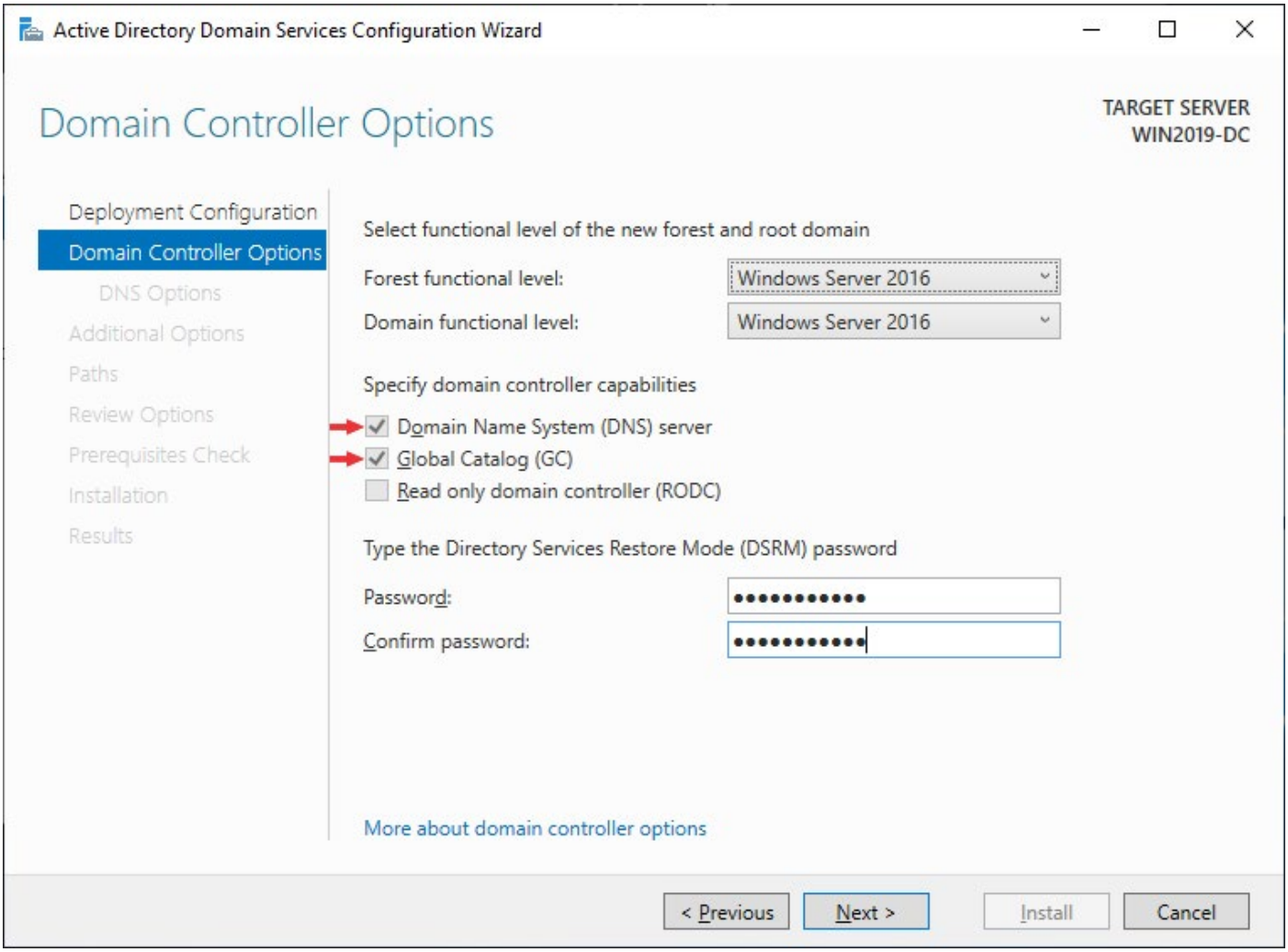


Figure 2.12

This is a list of available options for a forest functional level (see *Figure 2.13*).

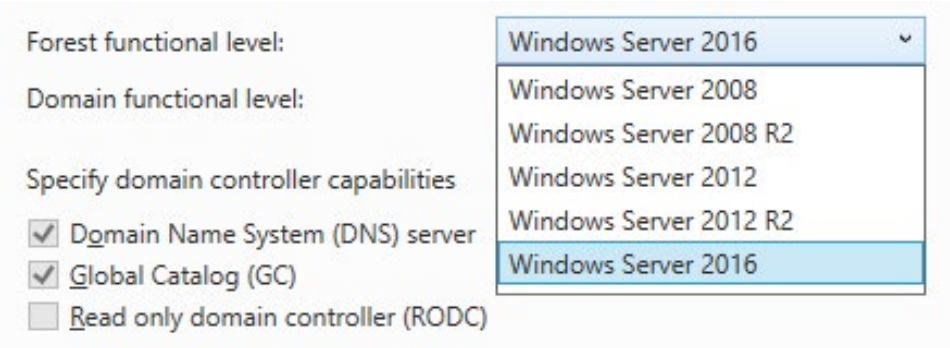


Figure 2.13

3. **DNS Options**. Select the **Create DNS delegation** checkbox. If you cannot select this checkbox, it means that you have previously selected and installed the DNS server role manually. You can ignore the warning and continue (see *Figure 2.14*).

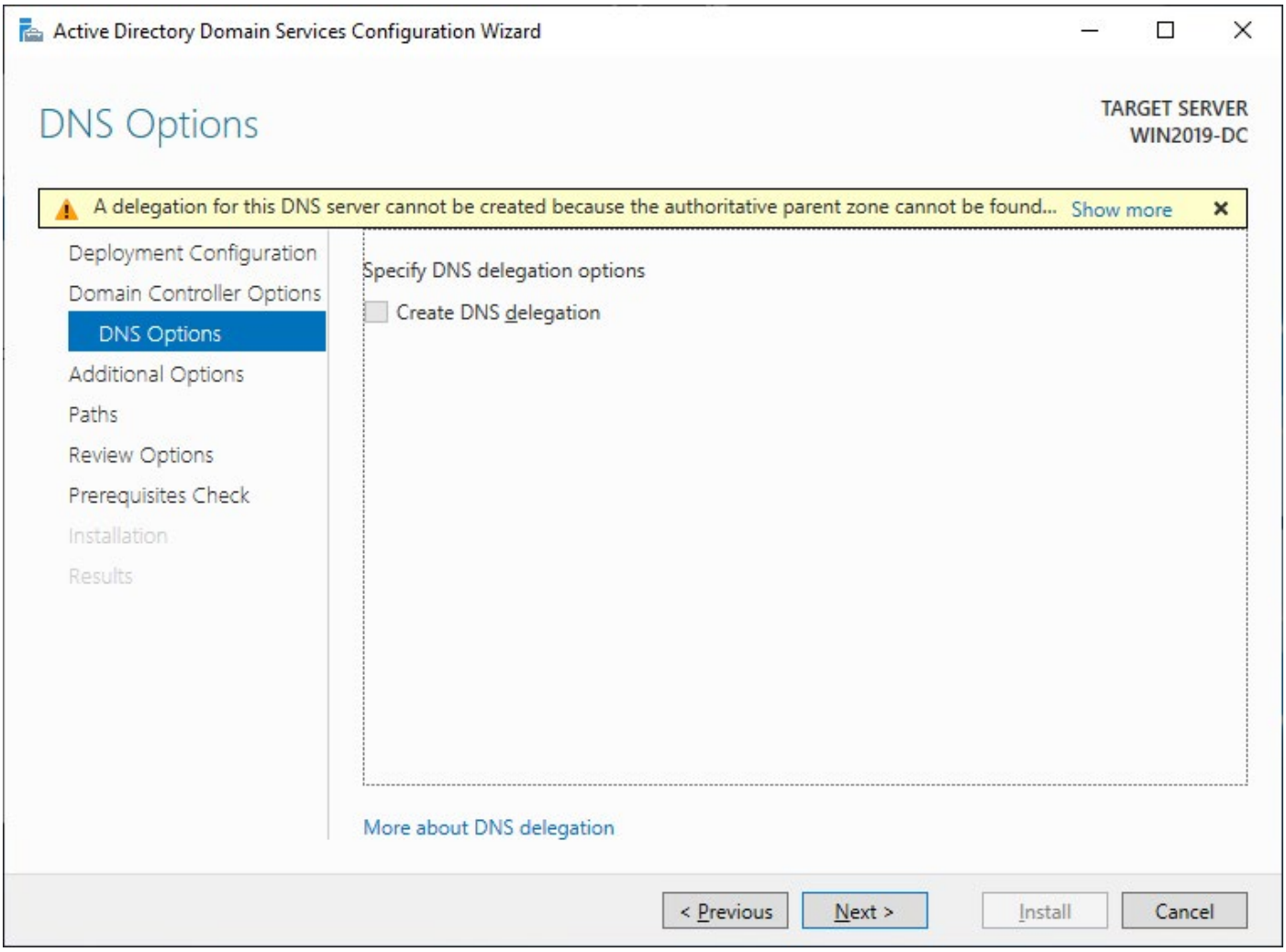


Figure 2.14

4. **Additional Options**. Verify the NetBIOS name assigned to the domain. In our case, the NetBIOS domain name is the same as the domain name and is correct (see *Figure 2.15*).

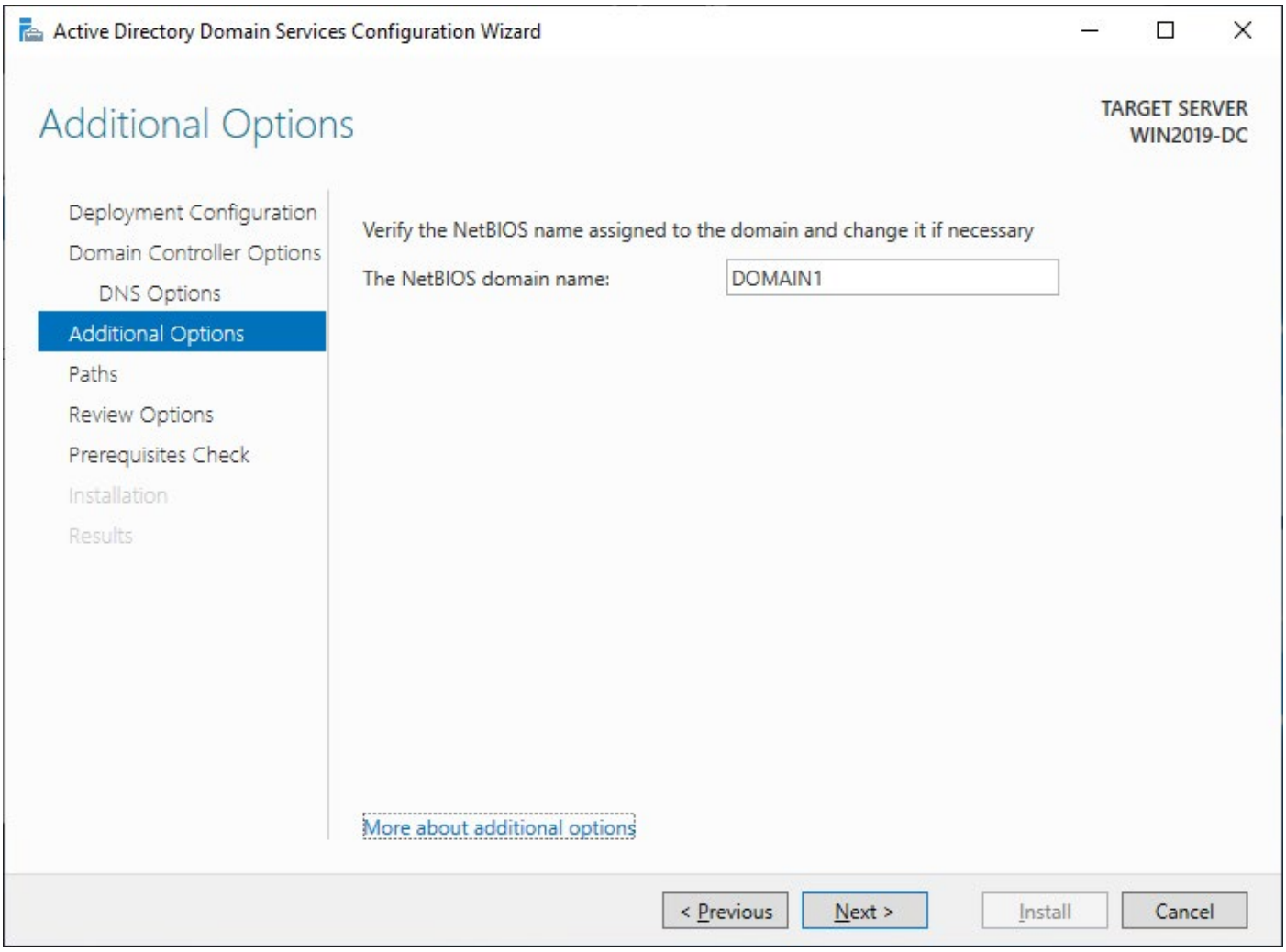


Figure 2.15

5. **Paths**. Specify the location of the AD DS database, log files, and SYSVOL (see *Figure 2.16*). You can use the default parameters.

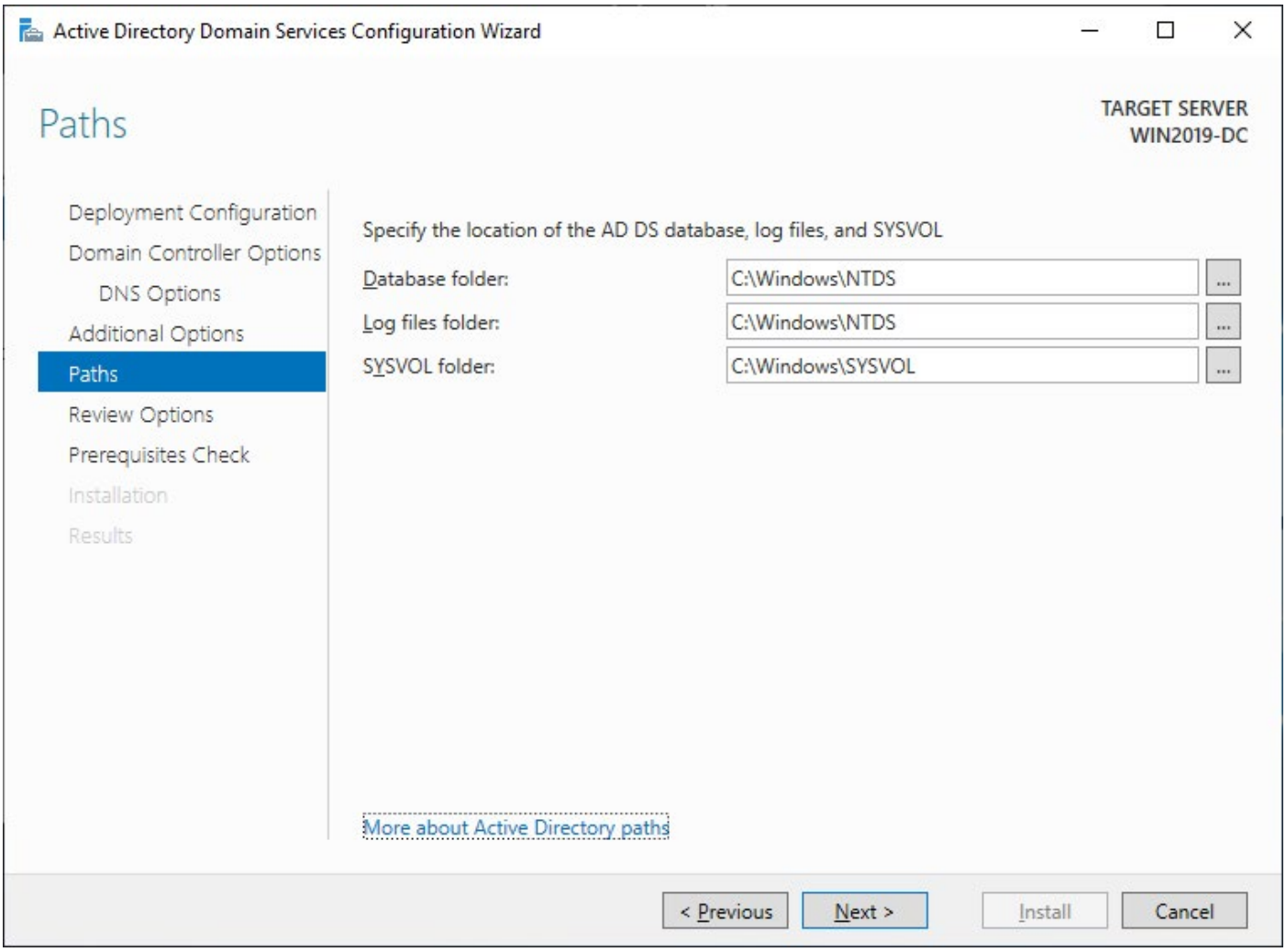


Figure 2.16

6. **Review options**. Review your selected configuration (see *Figure 2.17*). You can click the **View script** button to view a PowerShell script to automate the deployment and install AD DS in PowerShell next time. Click **Next** to check your configuration.

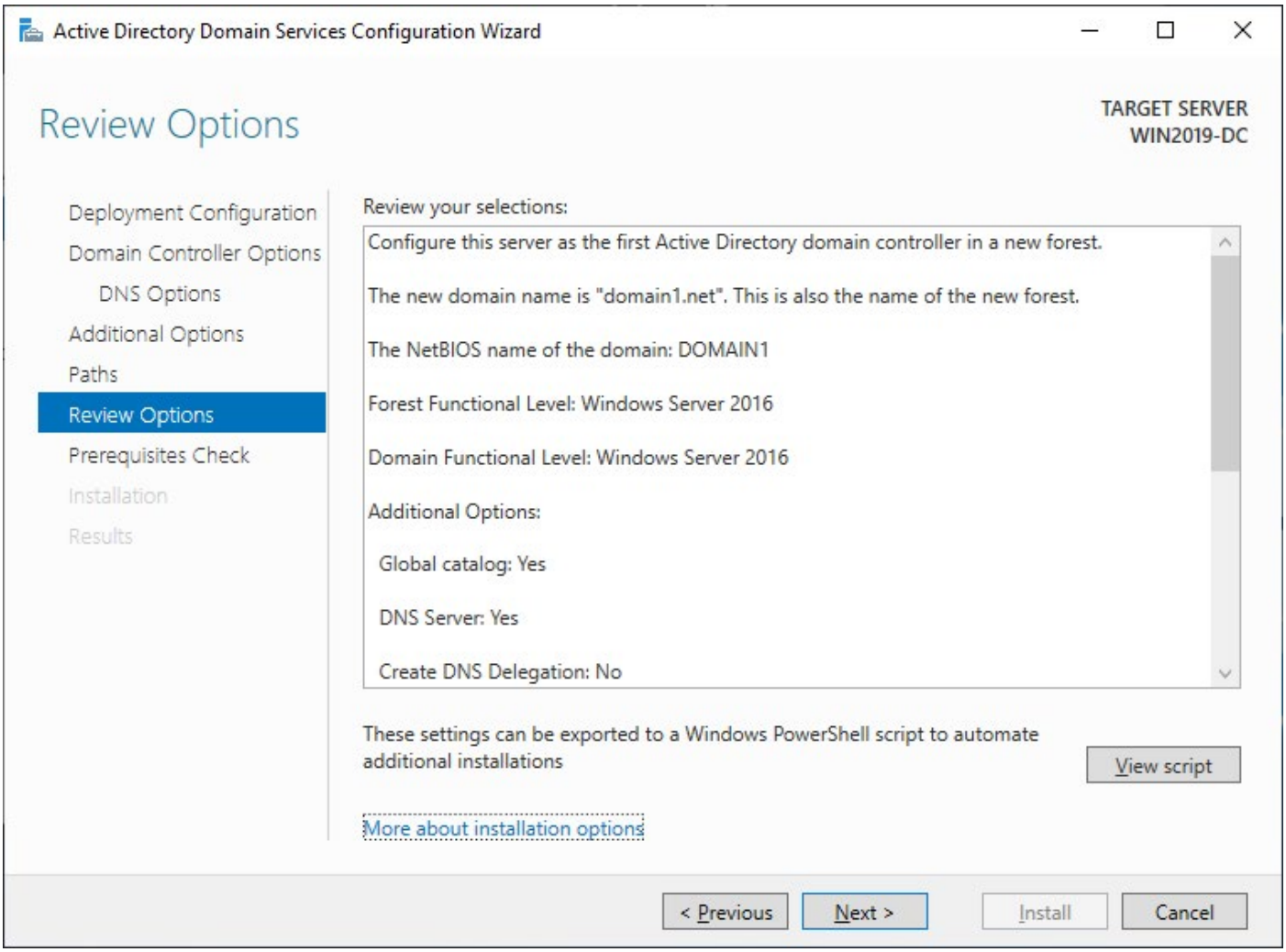


Figure 2.17

7. **Prerequisites Check**. Wait until checks are finished. If your configuration is correct, you should see the message: *All prerequisite checks passed successfully*.

   Click **Install** to begin the installation (see *Figure 2.18*).

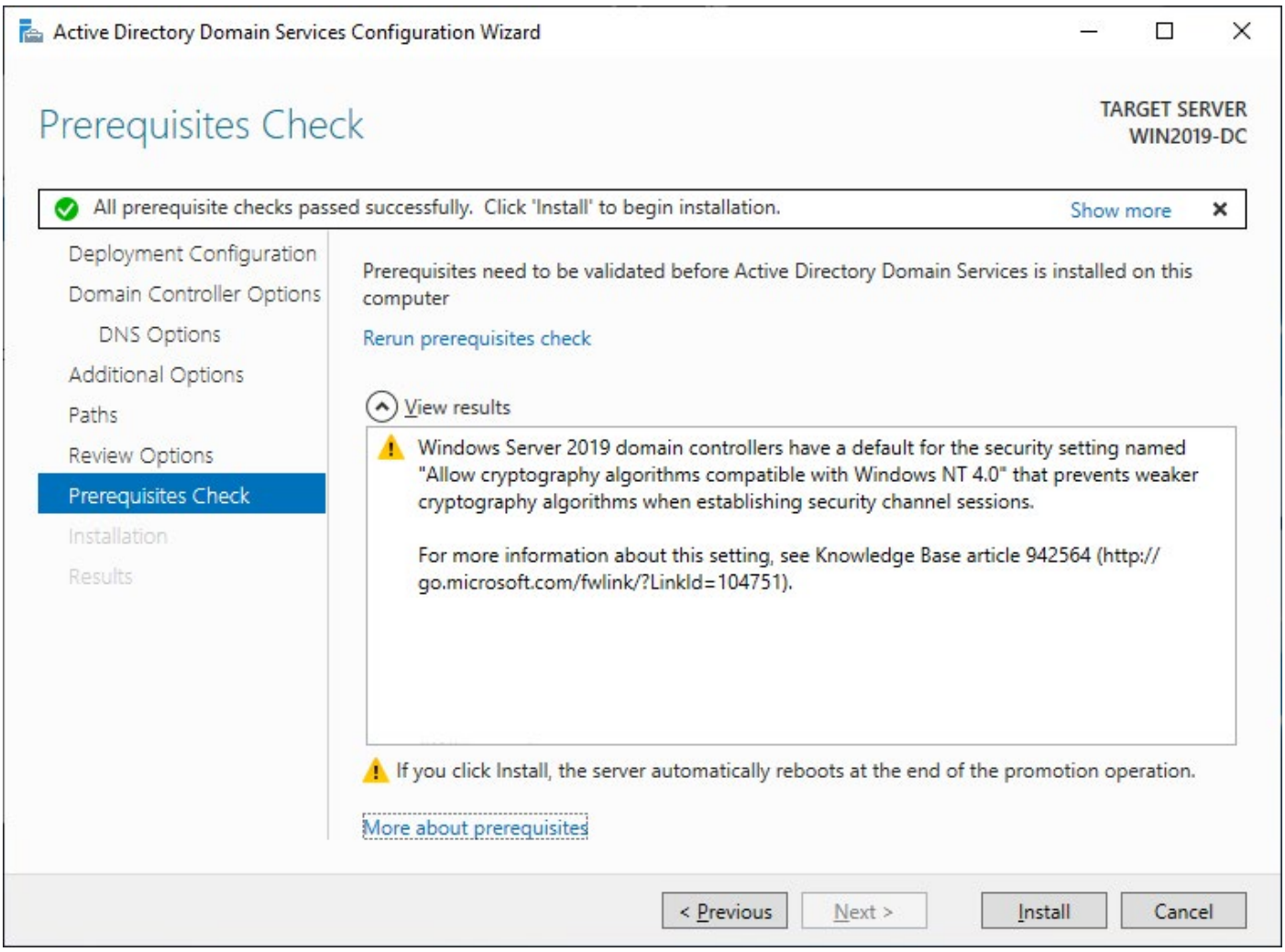


Figure 2.18

8. **Installation**. Wait until the domain services are installed.

9. **Results**. When the installation process is finished, you should see the message: *This server was successfully configured as a domain controller*.

   Reboot is required.

Log in to your Windows Server you have configured as a domain controller. Your local administrator account is now transformed into a domain administrator account. Use the appropriate credentials to log in.

Installing and Setting up vCenter Server

vCenter is a VMware centralized management server application that controls all virtual infrastructure and provides centralized management and operation, resource provisioning, and performance evaluation of virtual machines in vSphere virtual environment.

vCenter 6.7 is the latest version for Windows. Starting from vCenter 7.0, vCenter Server cannot be installed manually on Windows..

VMware vCenter Server Appliance (VCSA) is a preconfigured Linux-based virtual machine image with all necessary software installed. The operating system used for VCSA is called Photon OS.

Installing vCenter on a virtual machine has a number of advantages:

- No need for dedicating a separate server
- Snapshots usage and ease of backup
- Easy migration of VM from one host to another
- High availability for the vCenter Server system by using vSphere HA

Requirements

The following are the minimum requirements for vCenter Server installation:

- 2 virtual CPUs (vCPUs)
- 12 GB of RAM
- Disk storage: 40 GB minimum (depends on the database type and the number of VMs used); 412 GB recommended.
- The vCenter version must be compatible with versions of ESXi used in your vSphere environment.
- The vCenter version must not be lower than the version of the ESXi hosts you want to manage in vCenter.
- Fully Qualified Domain Names (FQDN) must be resolved in forward and reverse directions. DNS must be configured properly to resolve A records. Hostnames must be resolved between your local machine, ESXi hosts, and vCenter.
- The appropriate TCP and UDP ports must be allowed for network communications. ICMP connections must be allowed.

Methods to Install VCSA

The VMware vCenter Server Appliance installation image is provided as an ISO file.

There are two methods to deploy vCenter:

- Running the appliance setup by using a windows installer
- Installing from an OVA template

Both Windows installer and the OVA template are located in the ISO installation image of vCenter 7. The OVA virtual machine template file is located in the **vcsa** subdirectory. The vCenter installer you can run from Windows is located in the **vcsa-ui-installer\win32** directory in the ISO installation disk.

If you want to install VCSA from the OVA template, do the following:

1. Open *VMware Host Client*.
2. Click **Virtual Machines** in the navigation pane
3. Click **Create/Register VM**.
4. Select **Deploy a virtual machine from an OVF or OVA file**.
5. Browse the vCenter OVA file from your local computer.
6. Follow the steps in the wizard and finish the vCenter deployment process.

We use the first method in our walkthrough and run the vCenter installer from Windows. Read the detailed step-by-step explanation below.

How to Install vCenter

To install and set up vCenter Server, do the following:

1. Mount the vCenter installation ISO image to a virtual drive on a Windows machine. In this example, the virtual DVD drive is *F:*
2. Go to the directory where the installer EXE file is stored (see *Figure 3.1*) and run the file:

F:\vcsa-ui-installer\win32\installer.exe

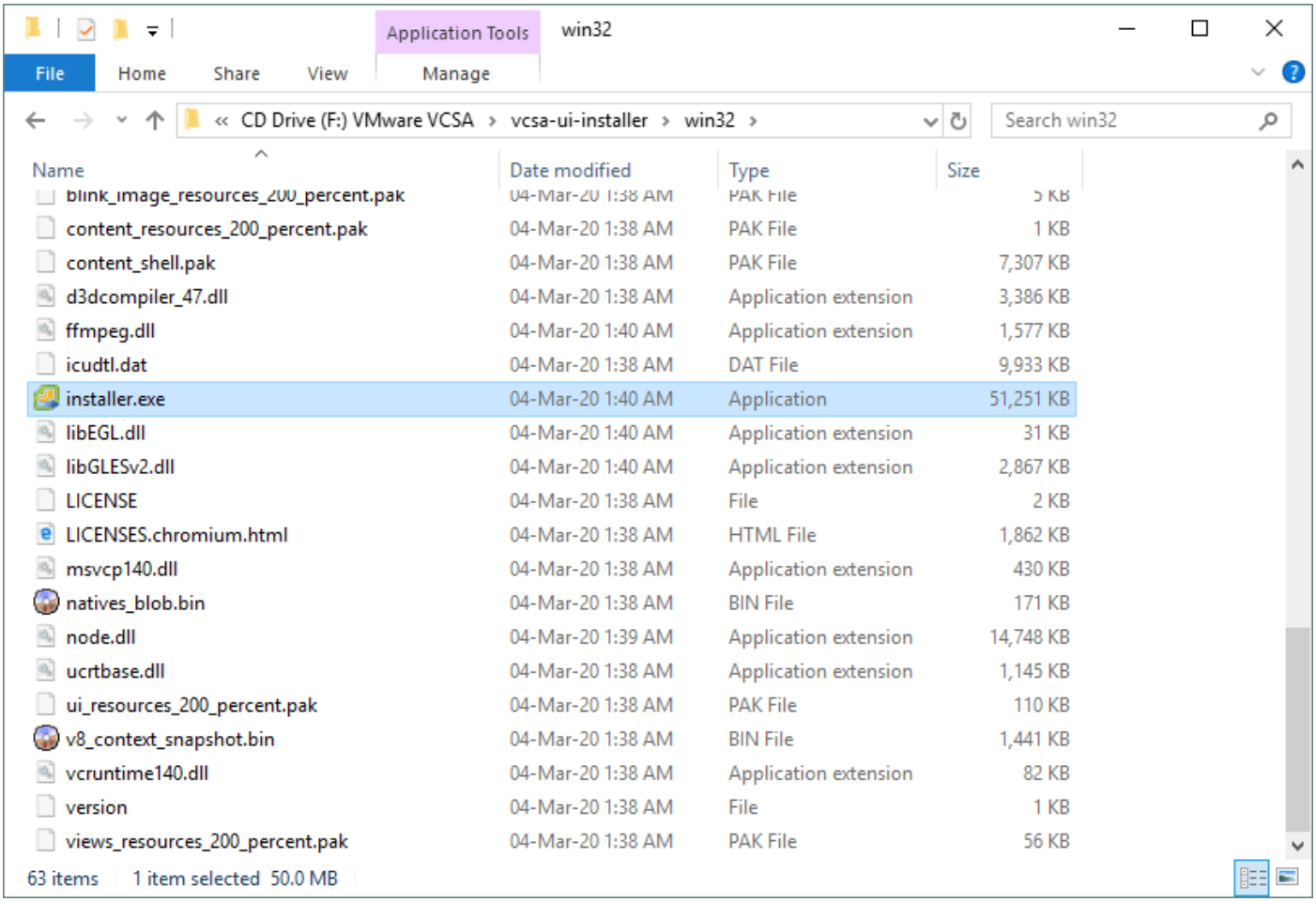


Figure 3.1

3. Click **Install** to install a new vCenter Server instance (see *Figure 3.2*).

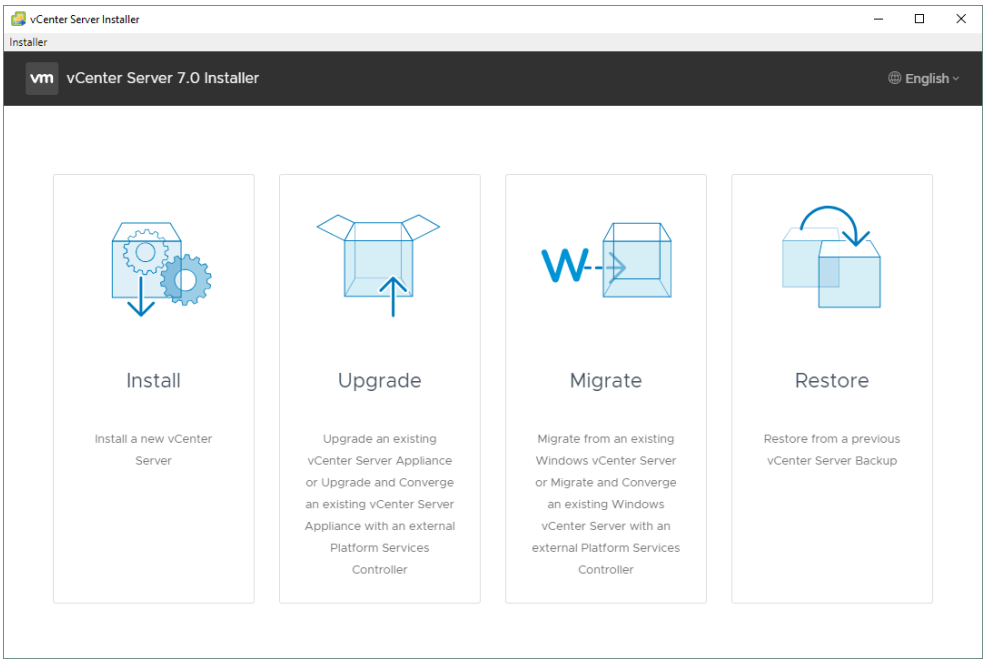


Figure 3.2

Stage 1: Deploy vCenter Server

Stage 1 of the vCenter installation wizard opens.

1. **Introduction.** Read notes about vCenter installation (see *Figure 3.3*). Click **Next** at each step of the wizard to continue.

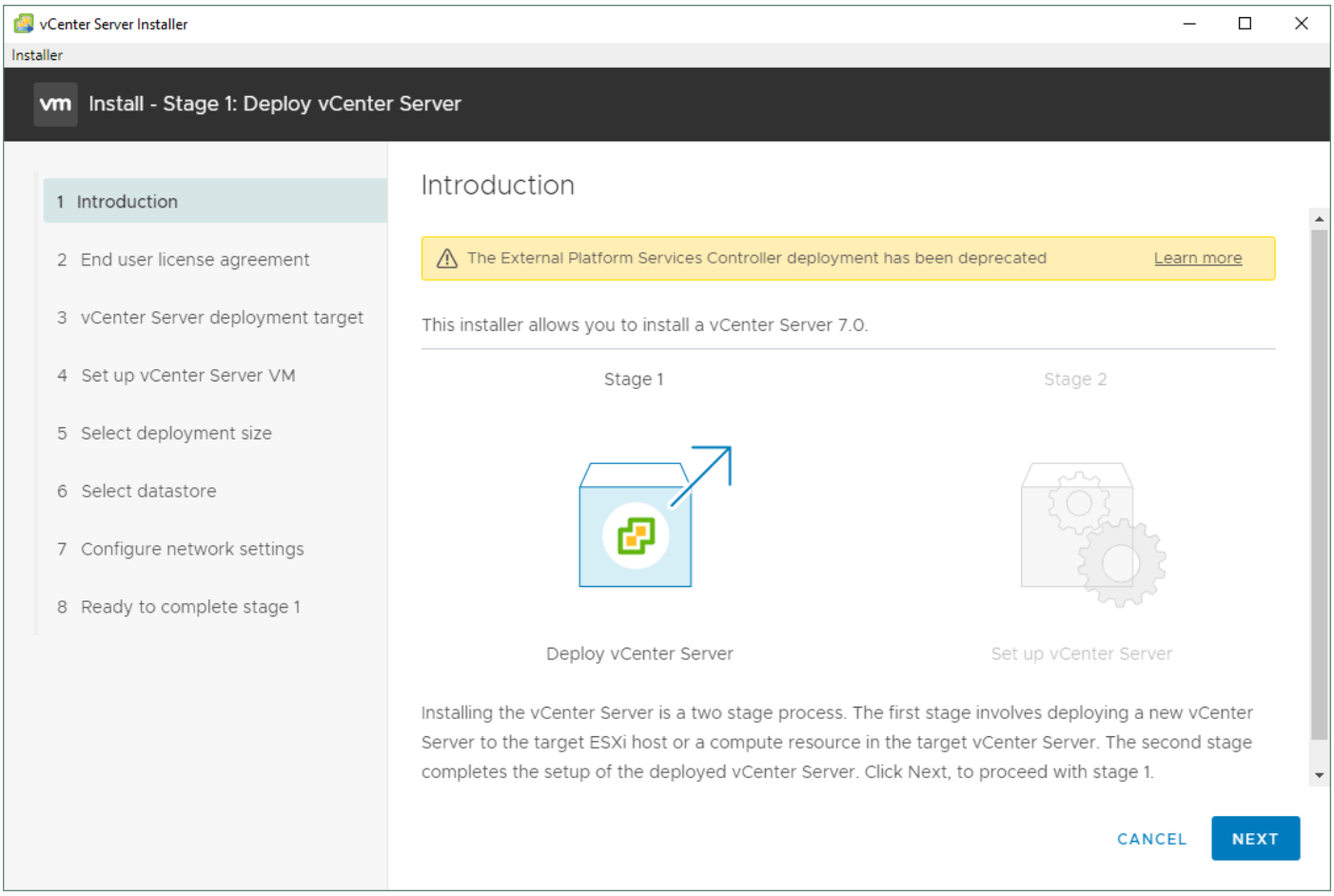


Figure 3.3

2. **End user license agreement**. Read the end-user license agreement (EULA) and select the **I accept the terms of the license agreement** checkbox (see *Figure 3.4*).

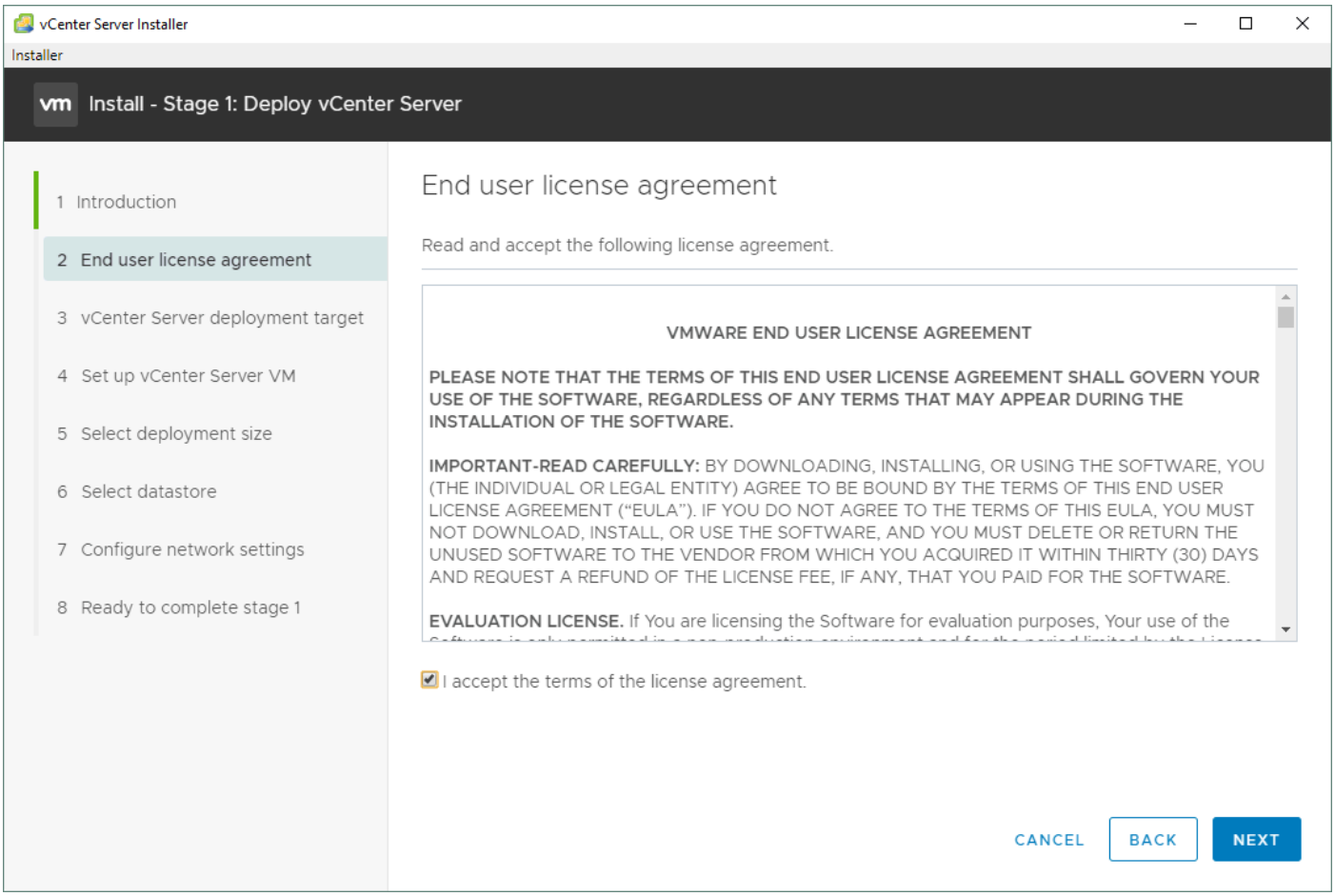


Figure 3.4

3. **vCenter Server deployment target**. Enter the IP address of the ESXi host on which you want to install the vCenter VM; enter the HTTPS port, username, and password to access the target ESXi host (see *Figure 3.5*).

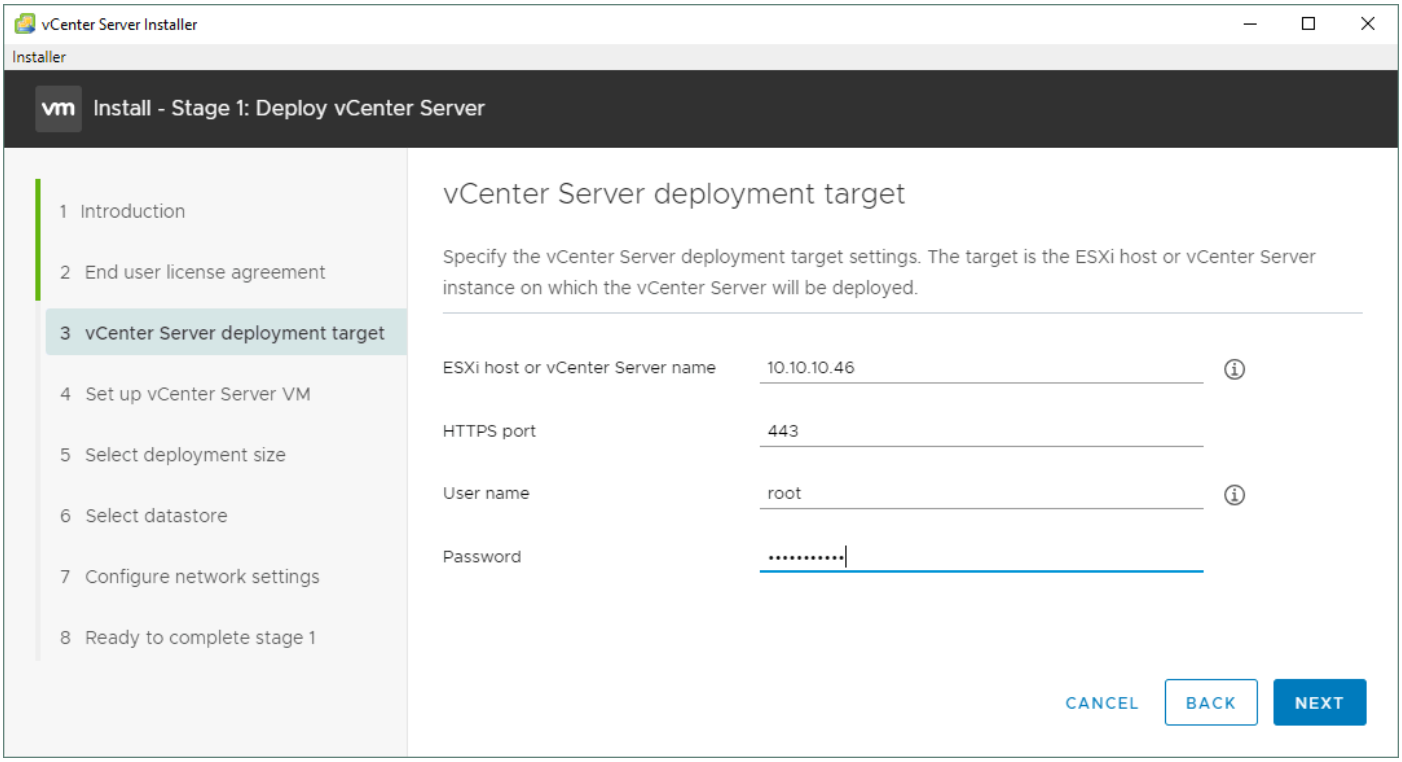


Figure 3.5

If you see a certificate warning, ignore the warning and click **Yes** to accept the certificate (see *Figure 3.6*).

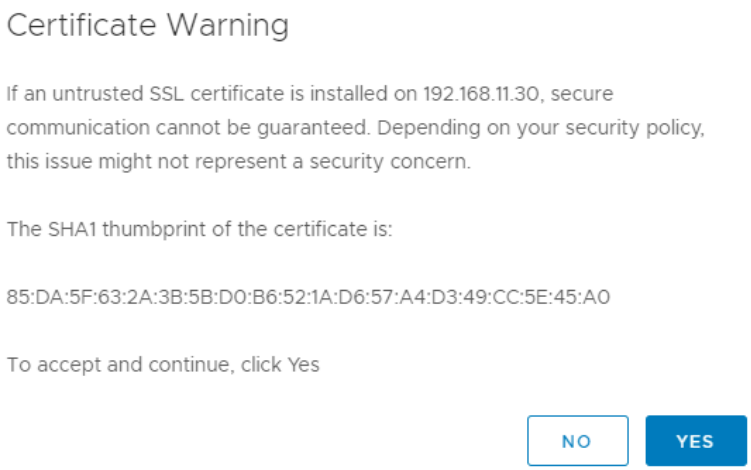


Figure 3.6

4. **Set up vCenter Server VM**. Specify the VM settings for the vCenter Server to be deployed, including the VM name and root password for VCSA.

   In this example, we set *vCenter7* as the VM name (see *Figure 3.7*).

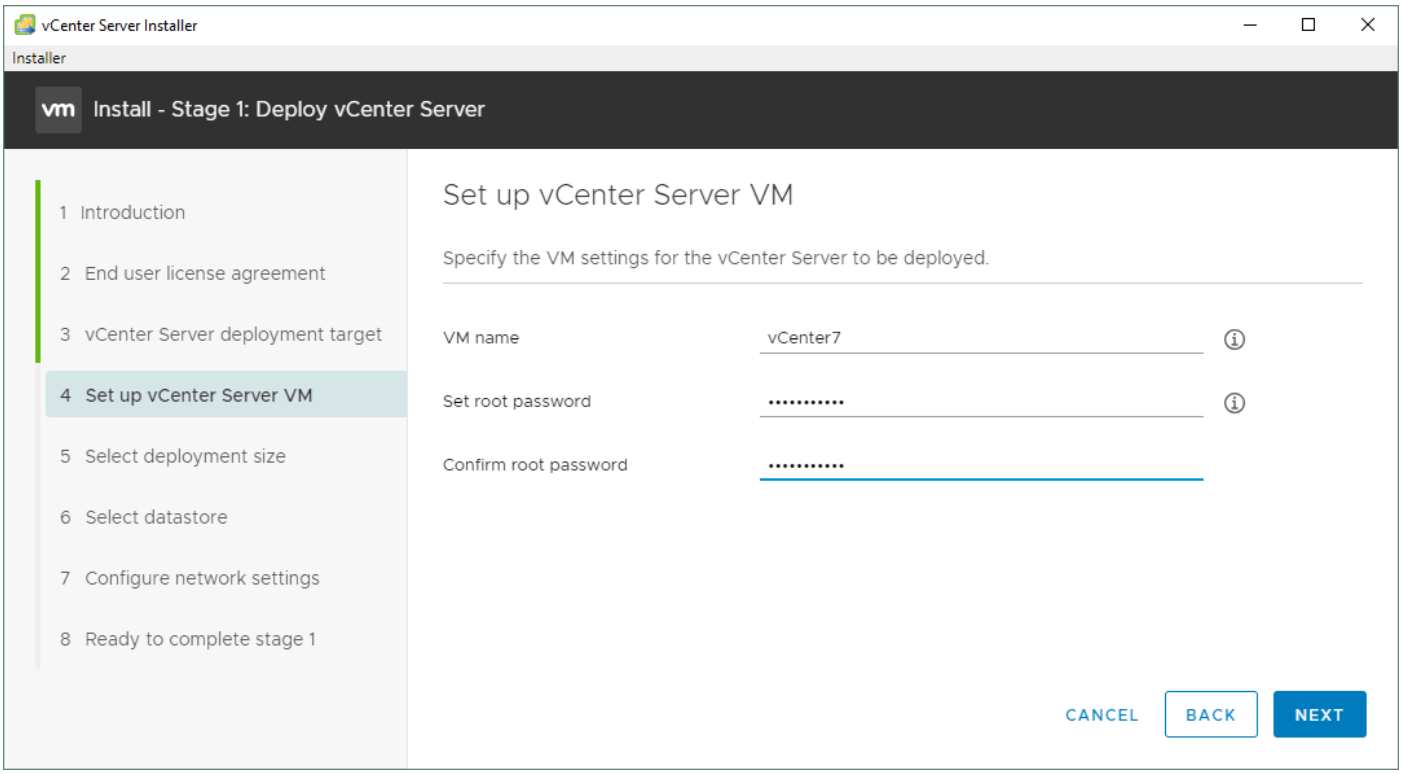


Figure 3.7

5. **Select deployment size**. The deployment size depends on the number of ESXi hosts and VMs you need to manage in vCenter. We select the **Tiny** deployment size (see *Figure 3.8*).

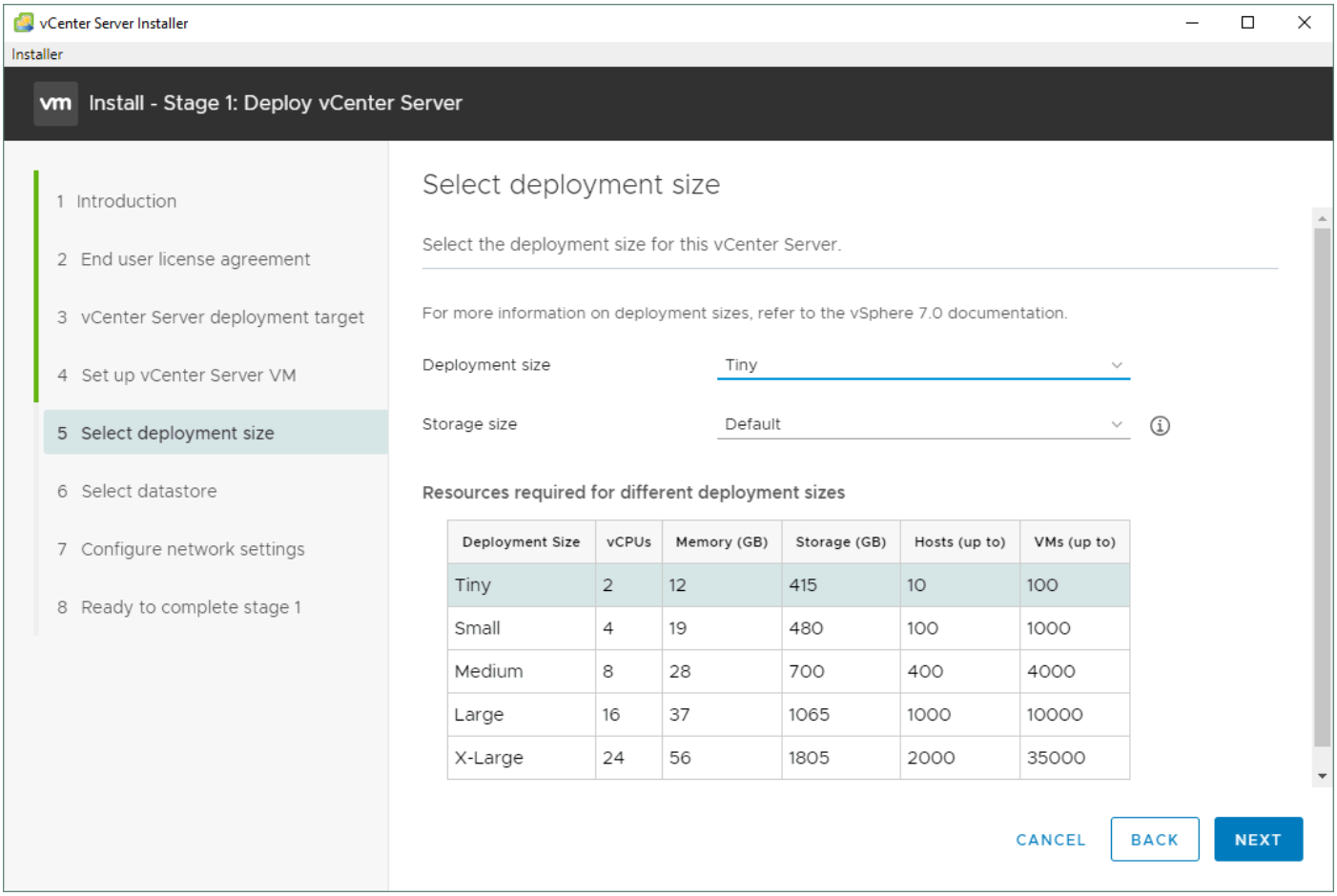


Figure 3.8

6. **Select datastore**. Select the storage location for this vCenter Server.
   - Select **Install on an existing datastore accessible from the target host**.
   - Select **Enable Thin Disk** mode to save storage space on the datastore (see *Figure 3.9*).

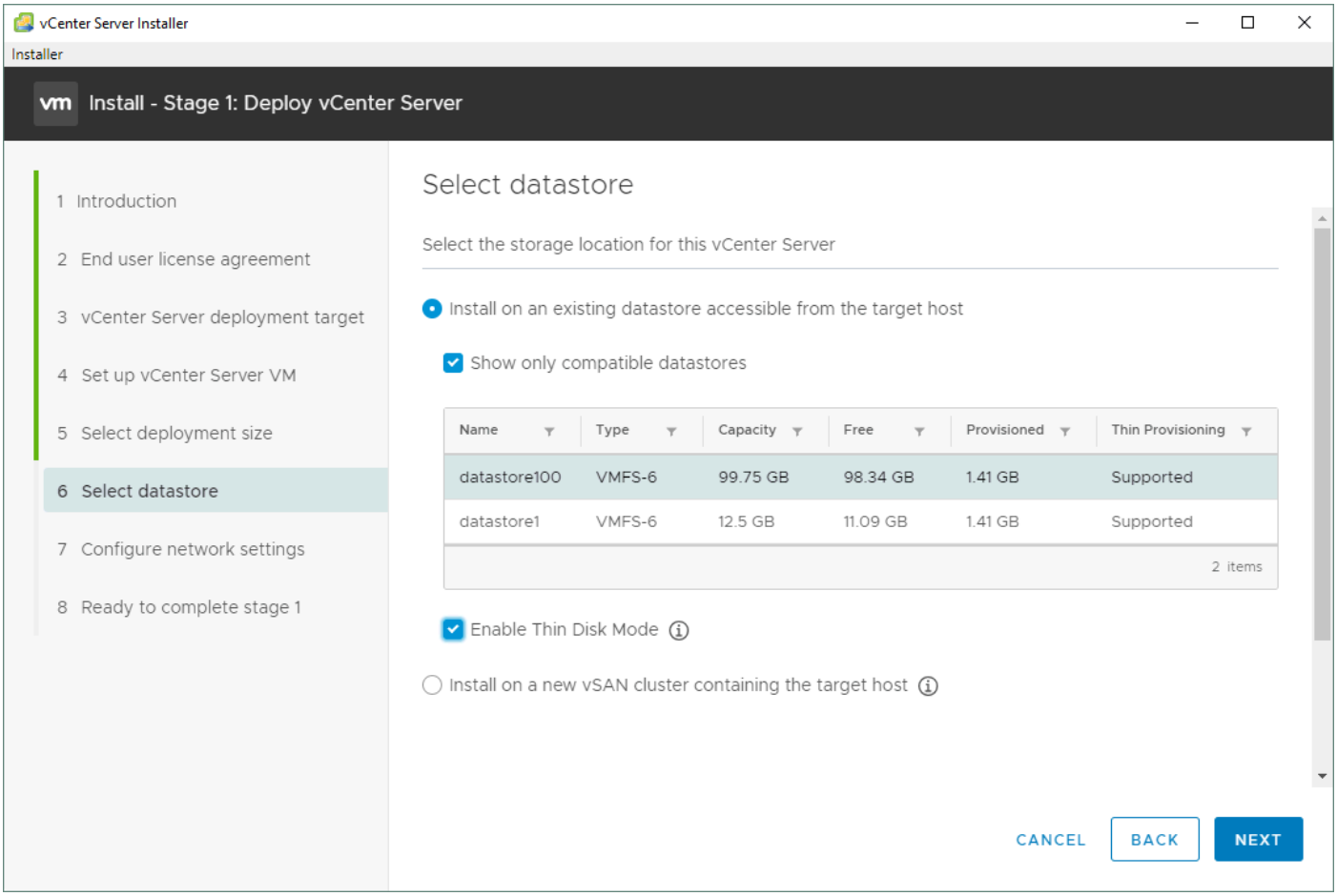


Figure 3.9

7. **Configure network settings.** This step is important for successful vCenter installation (see *Figure 3.10*). We use the configuration as follows, but you should take into account the network configuration in your environment.

   Network: *VM Network*

   IP version: *IPv4*

   IP assignment: *static*

   FQDN: *(optional)*

   IP address: *10.10.10.18*

   Subnet mask or prefix length: *255.255.255.0*

   Default gateway: *10.10.10.2*

   DNS servers: *10.10.10.2*

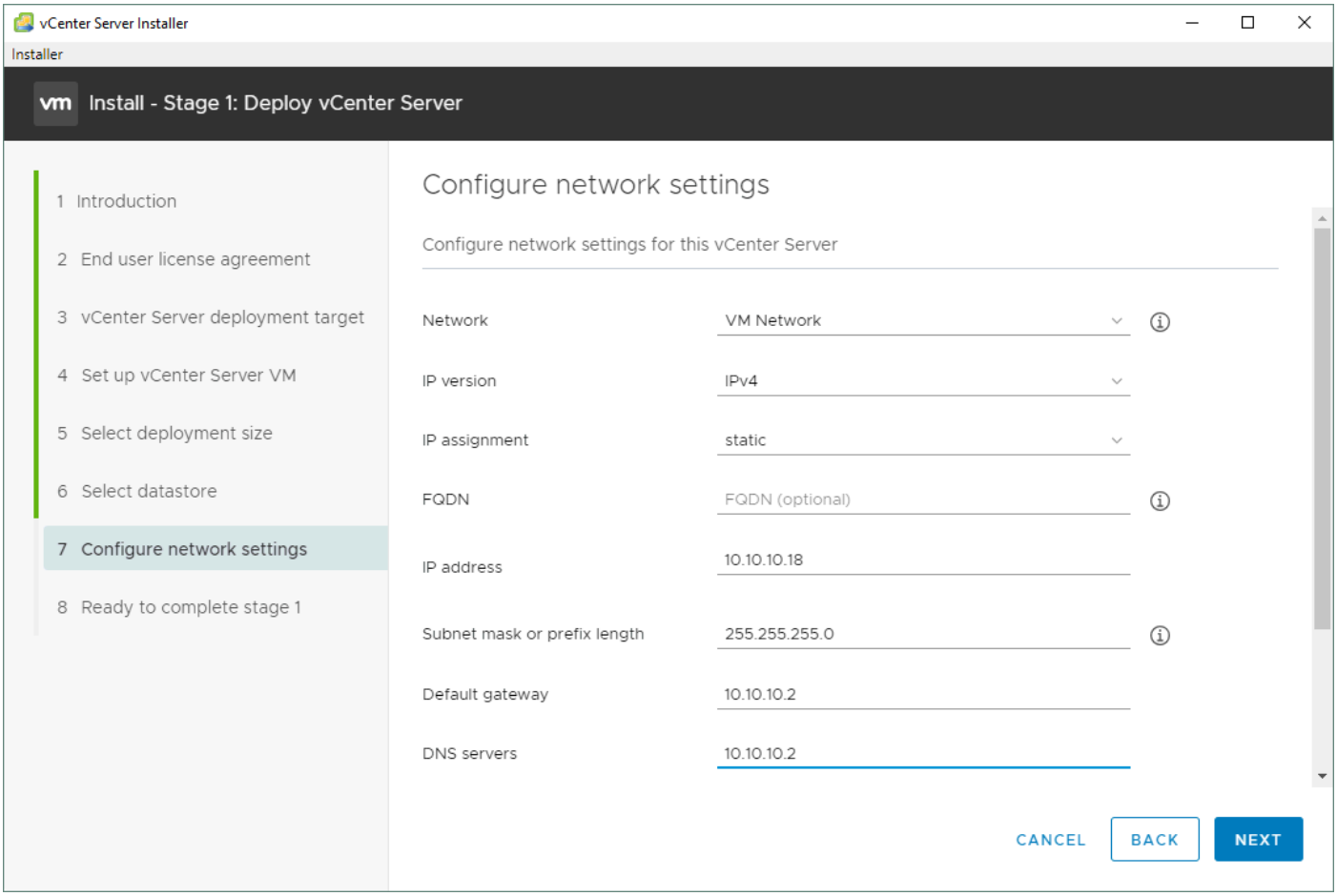


Figure 3.10

8. **Ready to complete Stage 1.** Review your settings (see *Figure 3.11*). If everything is correct, hit **Finish**.

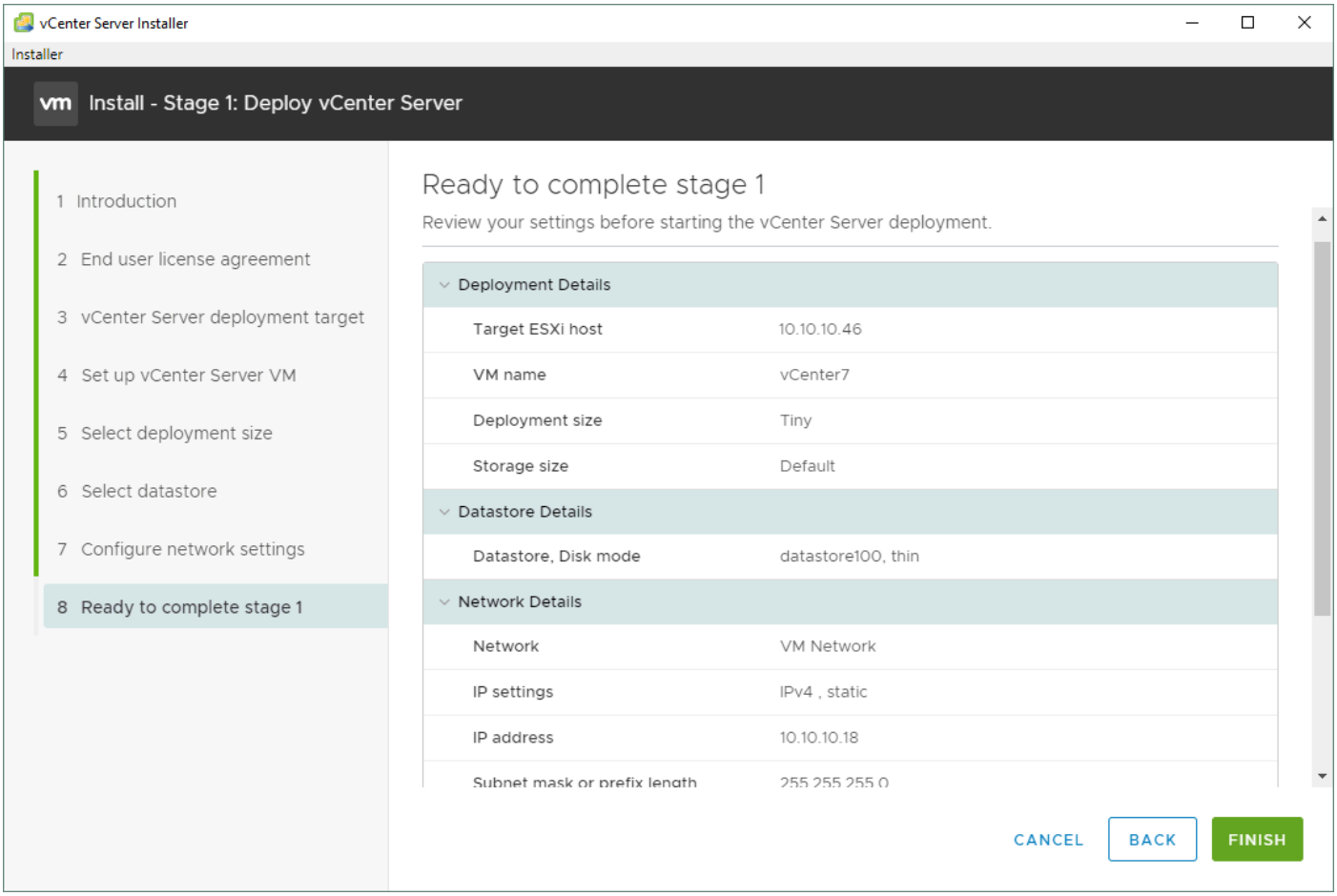


Figure 3.11

Wait until deploying the vCenter Server is finished for Stage 1 (see *Figure 3.12*).

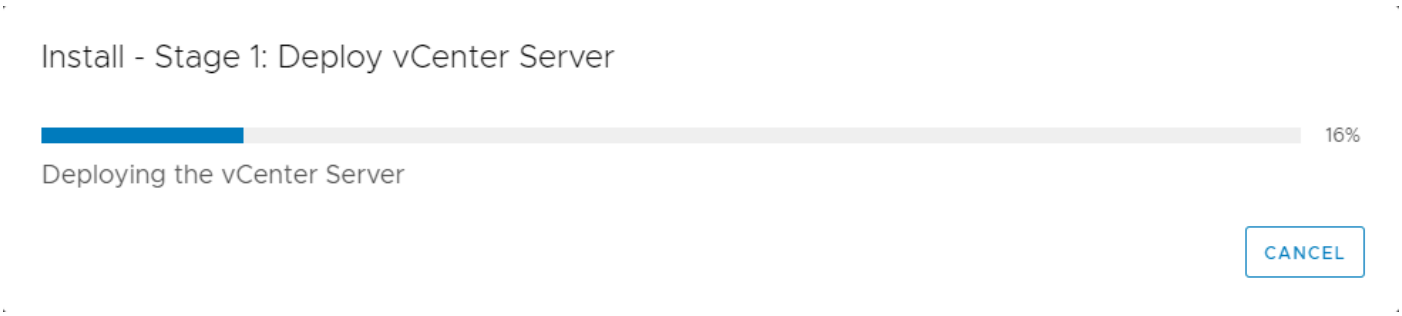


Figure 3.12

You have successfully deployed vCenter and finished Stage 1 (see *Figure 3.13*).

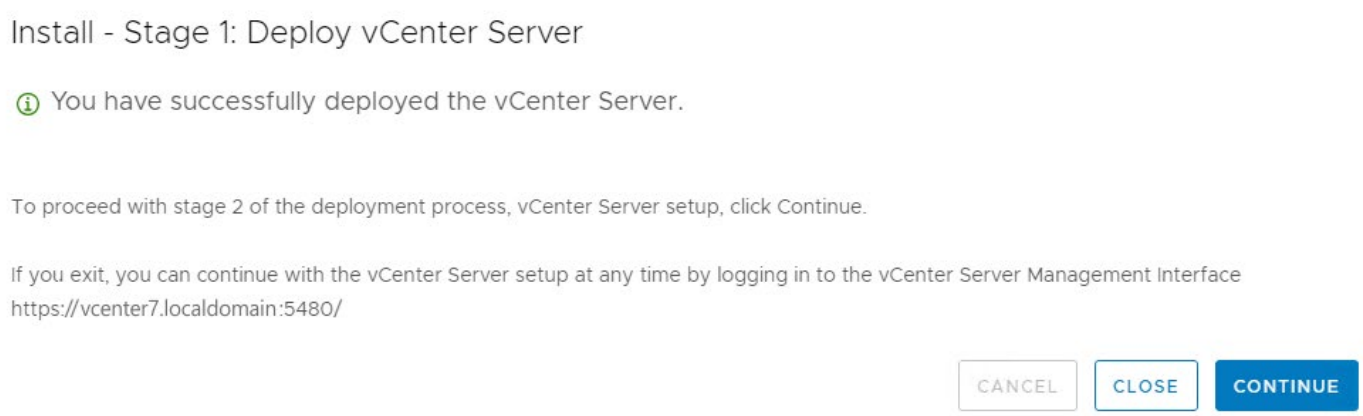


Figure 3.13

Now you are ready to go with *Stage 2* of vCenter deployment.

Stage 2: Set Up vCenter Server

1. Introduction. Read the notes and click **Next** to continue (see *Figure 3.14*).

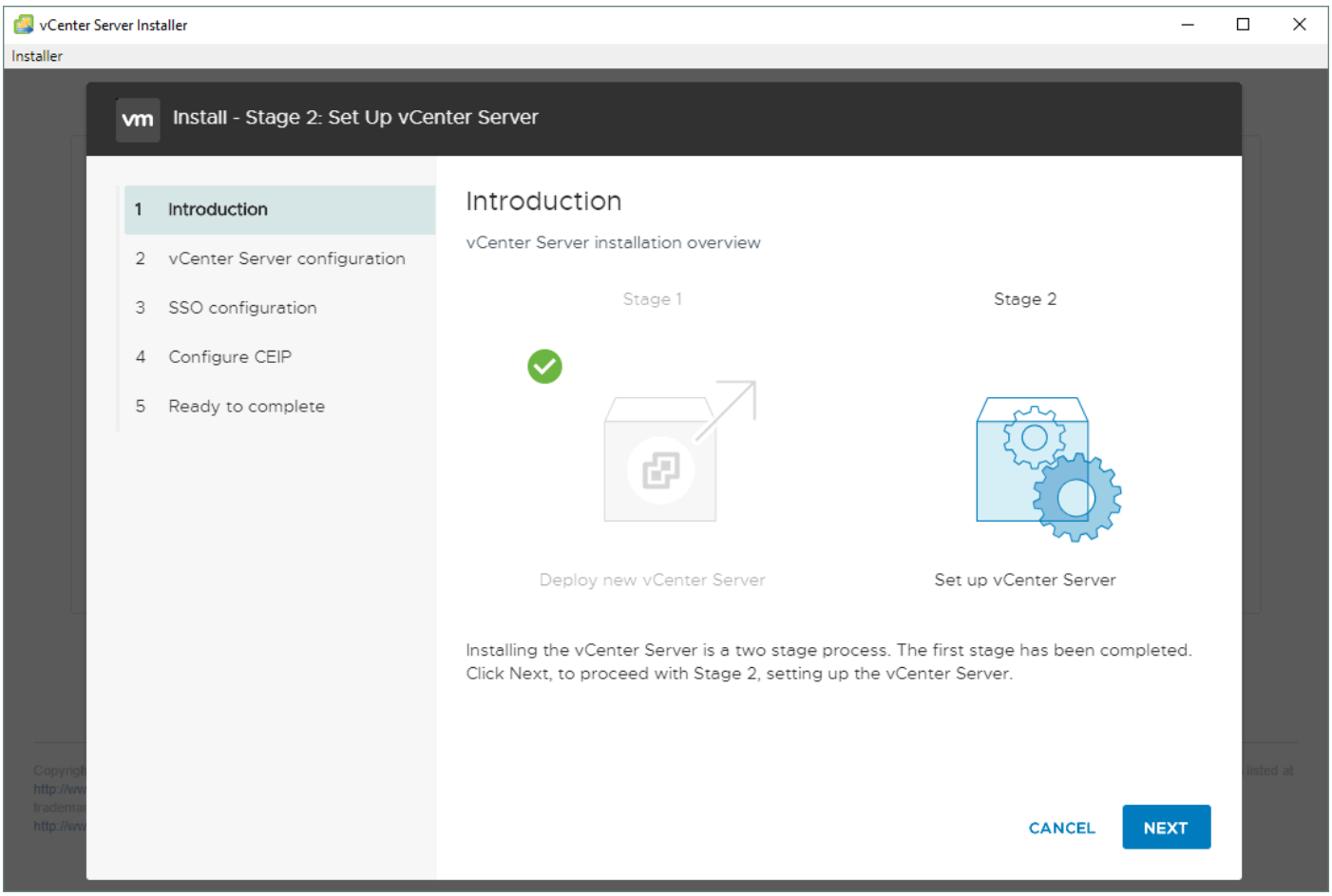


Figure 3.14

2. **vCenter Server configuration**. Select the following parameters (see *Figure 3.15*).
   - Time synchronization mode: *Synchronize time with the ESXi host*
   - SSH access: *Enabled*

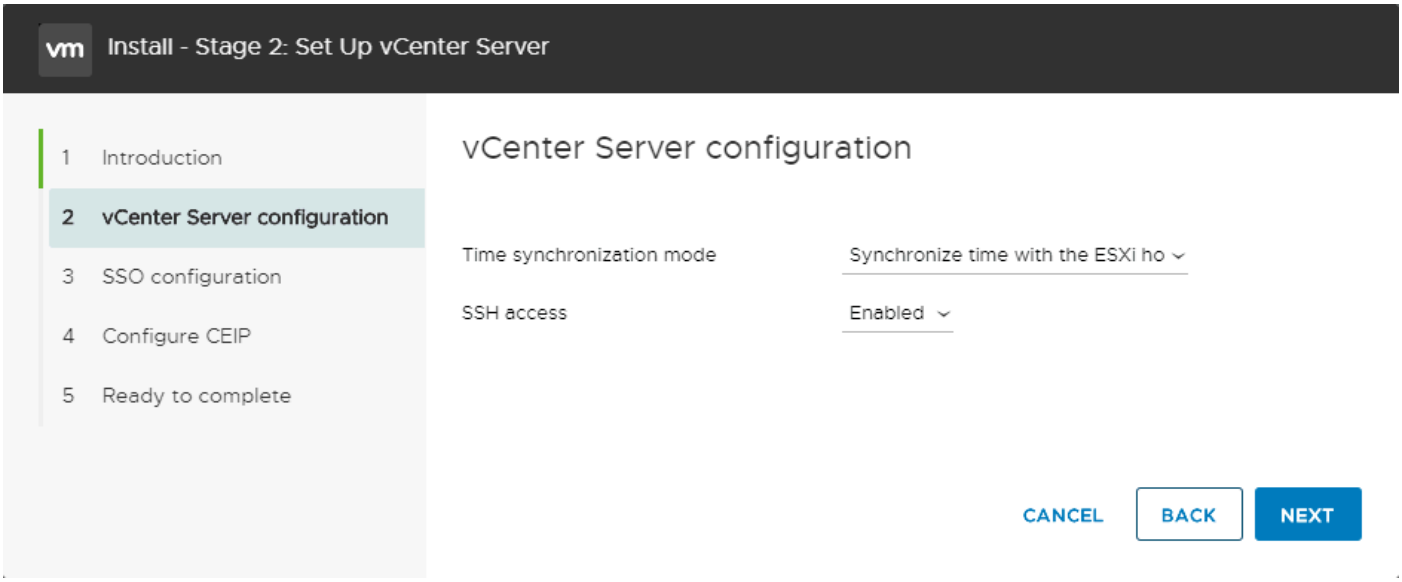


Figure 3.15

3. **SSO configuration**. Configure vCenter single sign-on (SSO). You can create a new SSO domain or select an existing SSO domain. We select the first option (see Figure 3.16).

   Single Sign-On domain name: *vsphere.local*

   Single Sign-On user name: *administrator*

   Single Sign-On password: **********

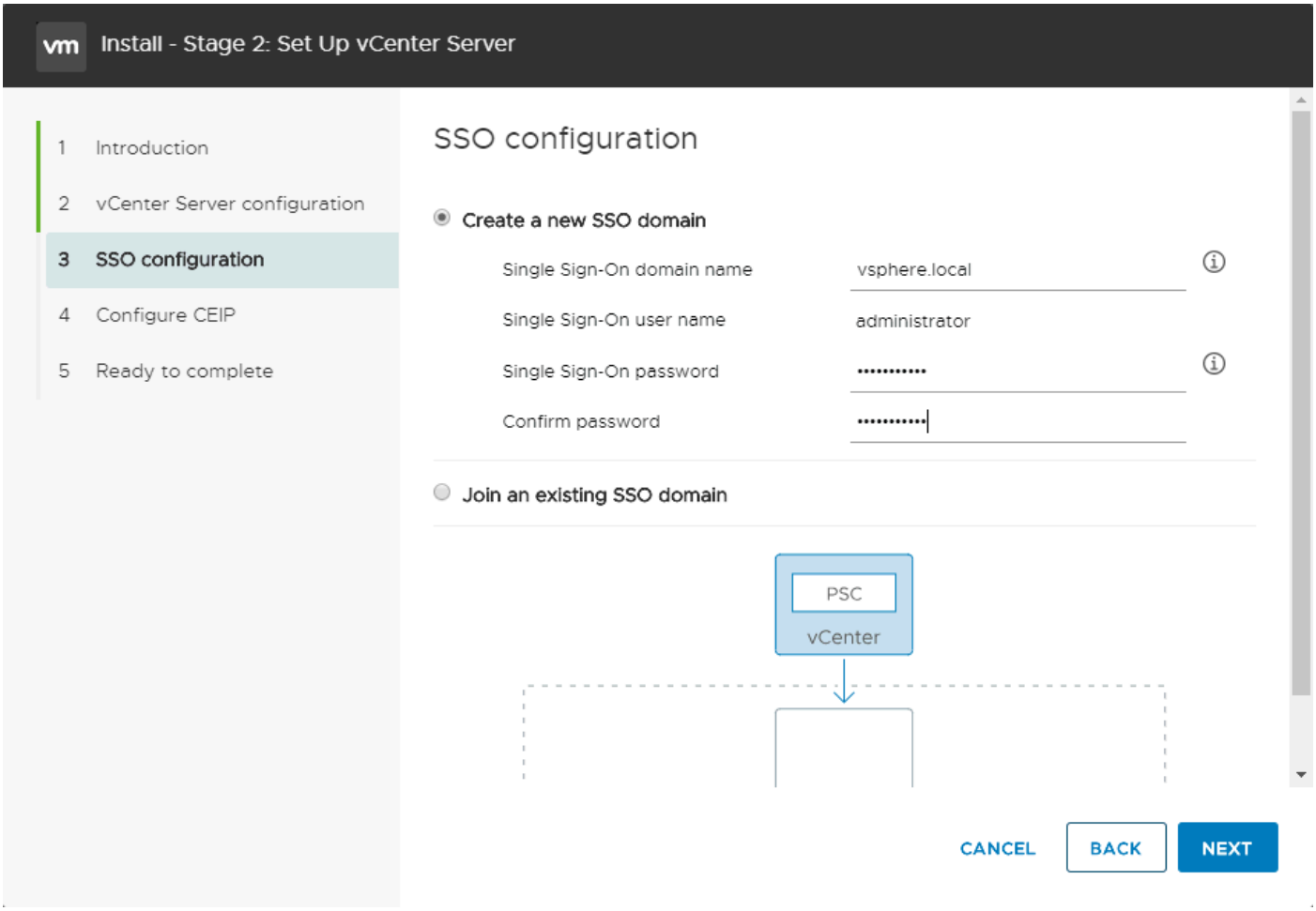

Install - Stage 2: Set Up vCenter Server
1 Introduction
2 vCenter Server configuration
3 SSO configuration
4 Configure CEIP
5 Ready to complete
SSO configuration
Create a new SSO domain
Single Sign-On domain name
vsphere.local
Single Sign-On user name
administrator
Single Sign-On password
Confirm password
Join an existing SSO domain
PSC
vCenter
CANCEL
BACK
NEXT

Figure 3.16

4. **Configure CEIP.** Leave the checkbox unselected (see *Figure 3.17*) if you don't want to participate in the VMware's Customer Experience Improvement Program (CEIP).

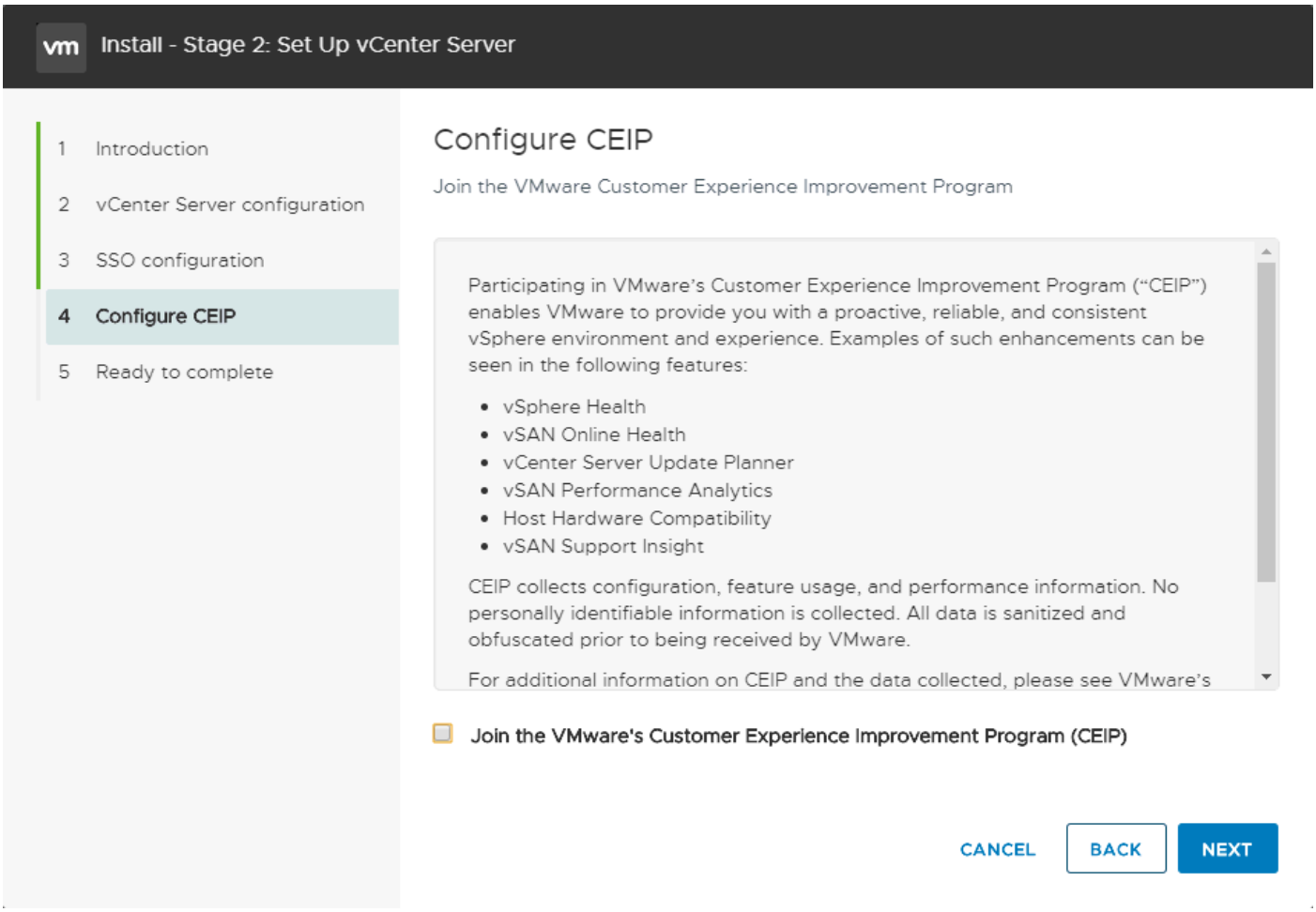


Figure 3.17

5. **Ready to complete.** Review your settings (see *Figure 3.18*) and, if everything is correct, click **Finish**.

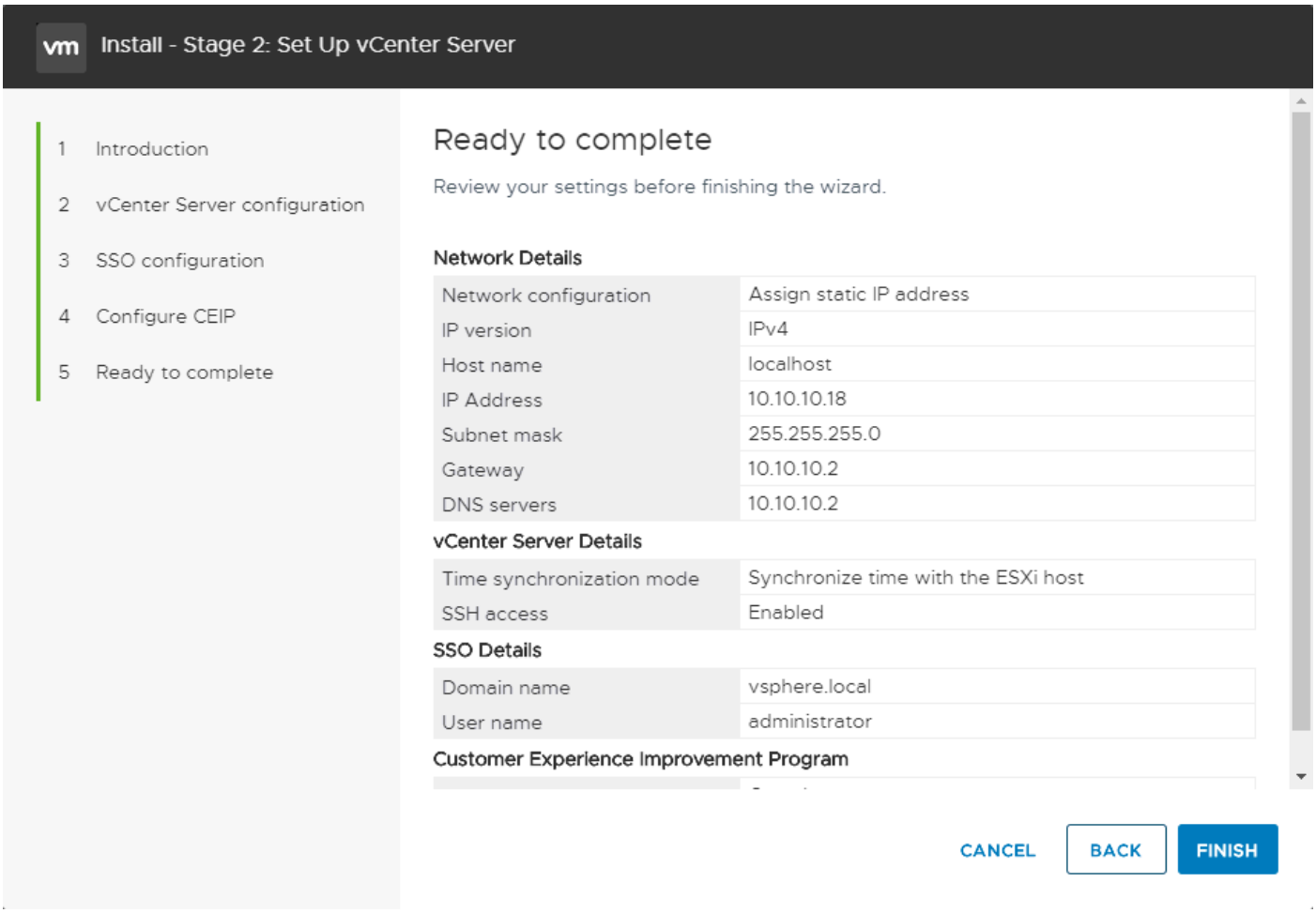


Figure 3.18

The warning message is displayed (see *Figure 3.19*). Don't interrupt the installation process. Click **OK** to continue.

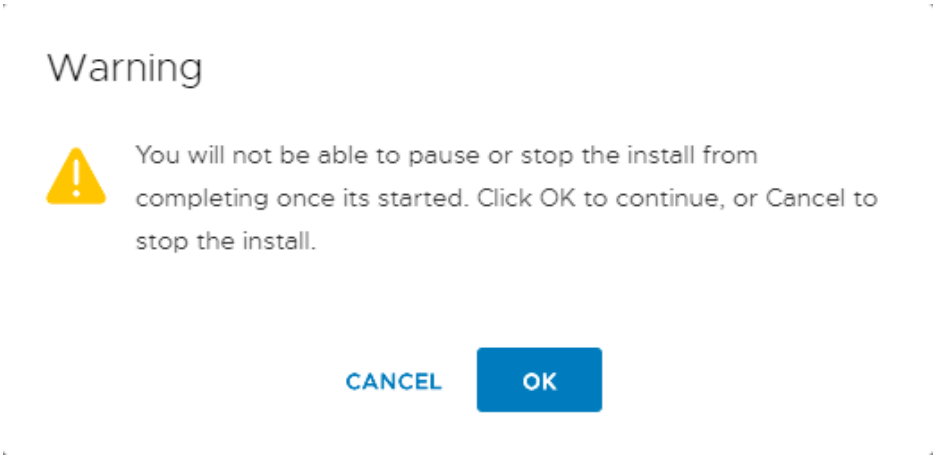


Figure 3.19

Wait until the *Stage 2* of vCenter Server setup is finished (see *Figure 3.20*). You can monitor the progress bar.

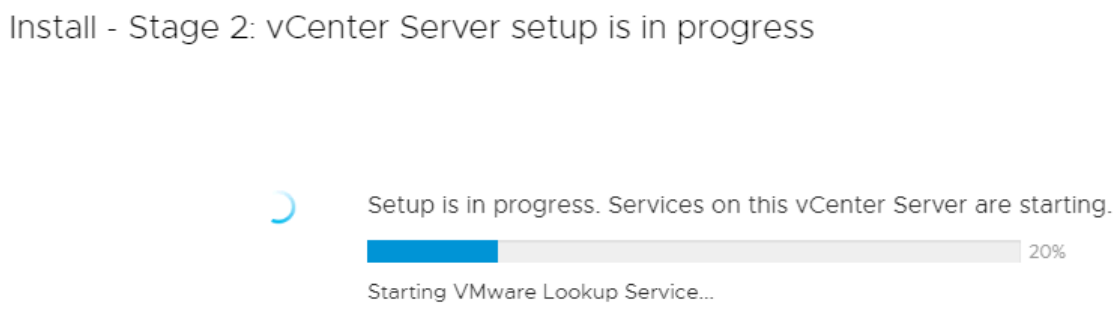


Figure 3.20

When the vCenter installation process is complete, click **Close**.

Now you can enter the IP address of your vCenter Server in a web browser, log in to vSphere Client by using vCenter credentials (*administrator@vSphere.local*) and manage your vSphere virtual environment (see *Figure 3.21*). You can enter your vCenter key and install the license. Otherwise, a full-featured 60-day trial will be used.

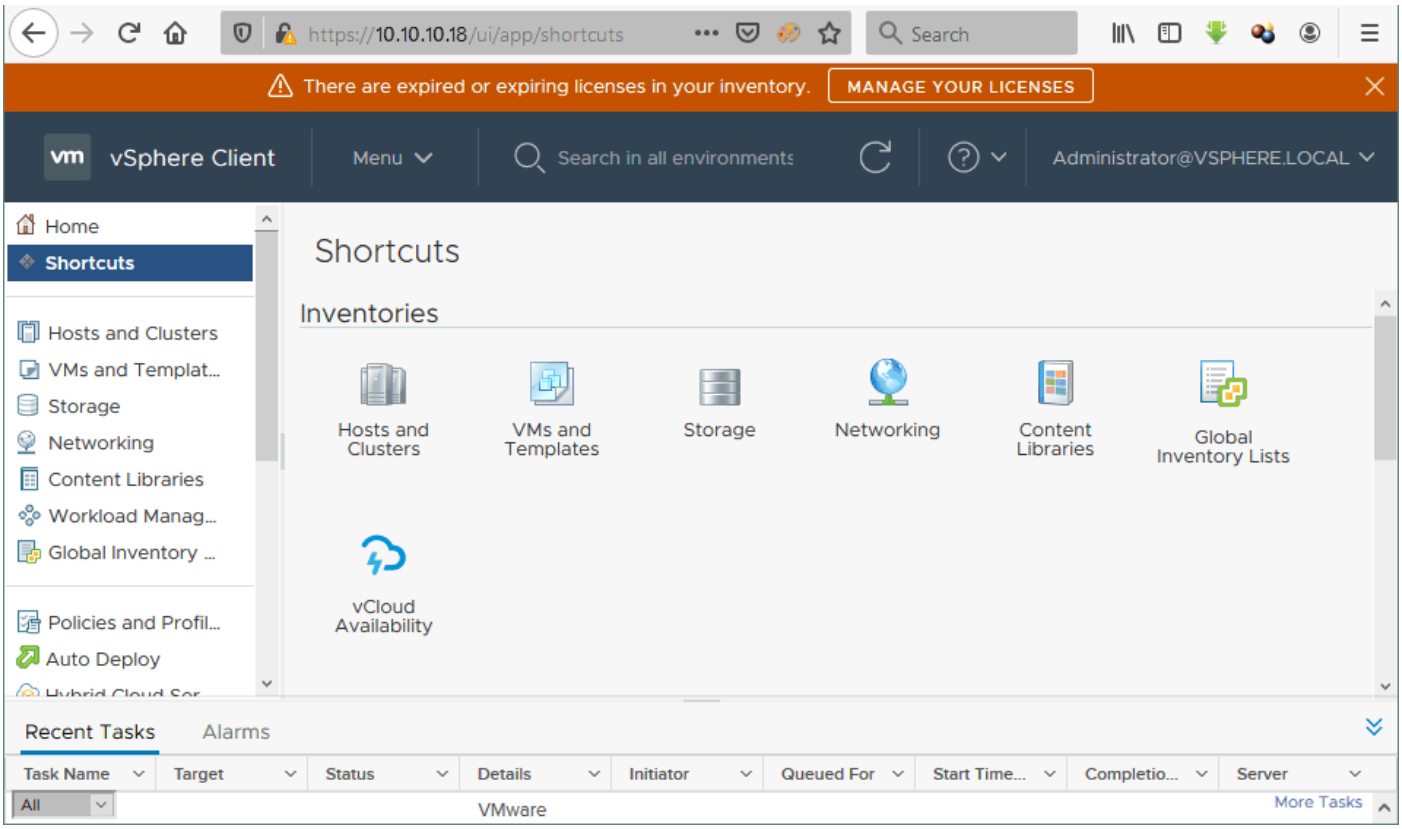


Figure 3.21

You can enter the vCenter IP address and port 5480 to access vCenter settings.

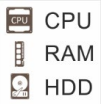
https://vcenter-ip-address:5480

It is recommended that you log in to this page and set the root password expiration date. This configuration can help you avoid situations when the root password expiration is unexpected.

ESXi hosts and vCenter connection scheme

There are several ways to connect vCenter and ESXi hosts in vSphere clusters:

Note
Icons in the diagrams below mean the following:



1. vCenter Server is a virtual machine that is installed on an ESXi host and uses CPU, RAM, and the storage of the ESXi host (see *Figure 3.22*).

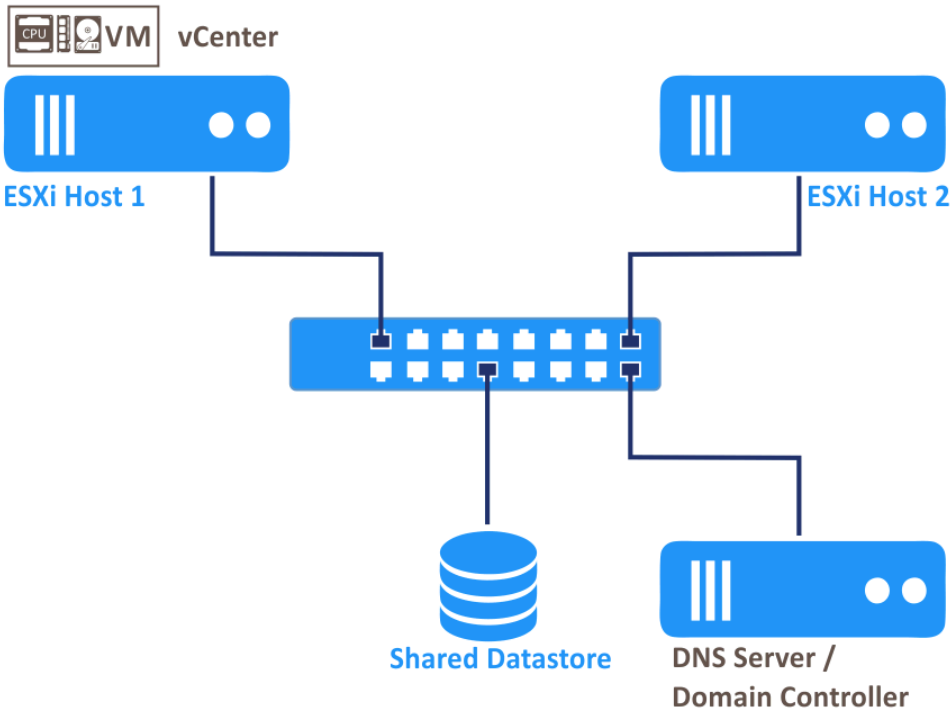


Figure 3.22

2. vCenter Server is a virtual machine running on the ESXi Server that uses CPU and RAM of ESXi server, but the virtual disk is stored on a shared datastore (see *Figure 3.23*). This method of connecting hosts in a cluster allows you to use cluster features, such as High Availability, Distributed Resource Scheduler, and Fault Tolerance.

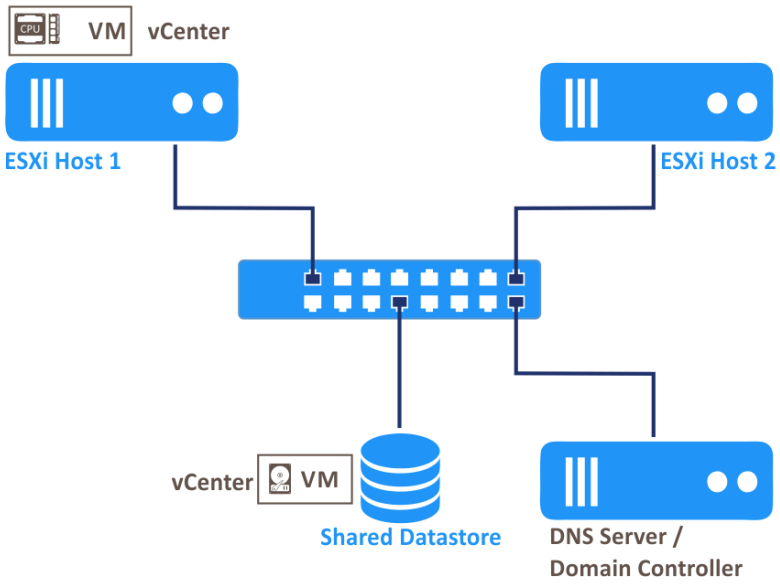


Figure 3.23

3. A domain controller and vCenter Server are both installed and running on an ESXi server using CPU, RAM, the local storage of the ESXi server (see *Figure 3.24*).

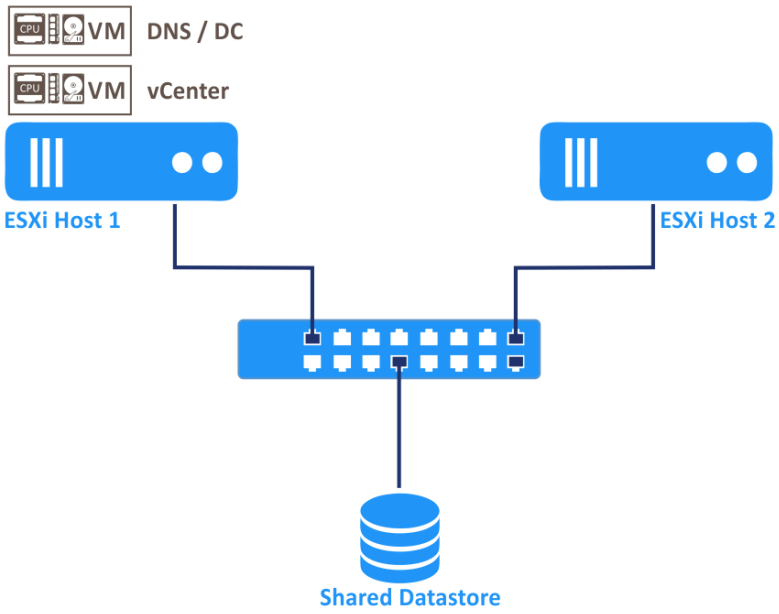


Figure 3.24

4. A domain controller and vCenter Server are running on an ESXi server using CPU, RAM, and the storage of the ESXi server, but virtual disks of these VMs are stored on the shared datastore (see *Figure 3.25*). The advantages of this connection method are similar to those of method No. 2.

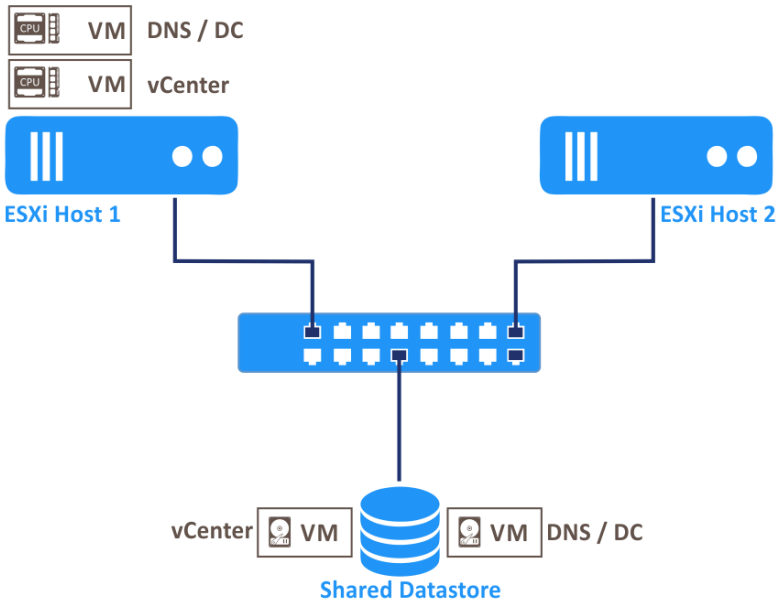


Figure 3.25

Adding Items to the vCenter Inventory

Once we have deployed vCenter, we need to add ESXi hosts and other items to the vCenter inventory.

Adding a New Datacenter

A datacenter is the highest logical unit of separating resources in vCenter. A datacenter is a container for all the inventory objects required to complete a fully functional environment for operating virtual machines.

You can create multiple datacenters for each department in your enterprise or for different purposes such as low and high-performance tasks. Your virtual machines can migrate (including hot migration) from one ESXi host to another in the same datacenter. However, they cannot migrate from a host in one datacenter to a host in a different datacenter.

1. Right-click the **vCenter** item in the navigation pane of VMware vSphere Client (the **Hosts and clusters** view is selected).
2. In the menu that opens, click **New Datacenter** (see *Figure 3.26*).

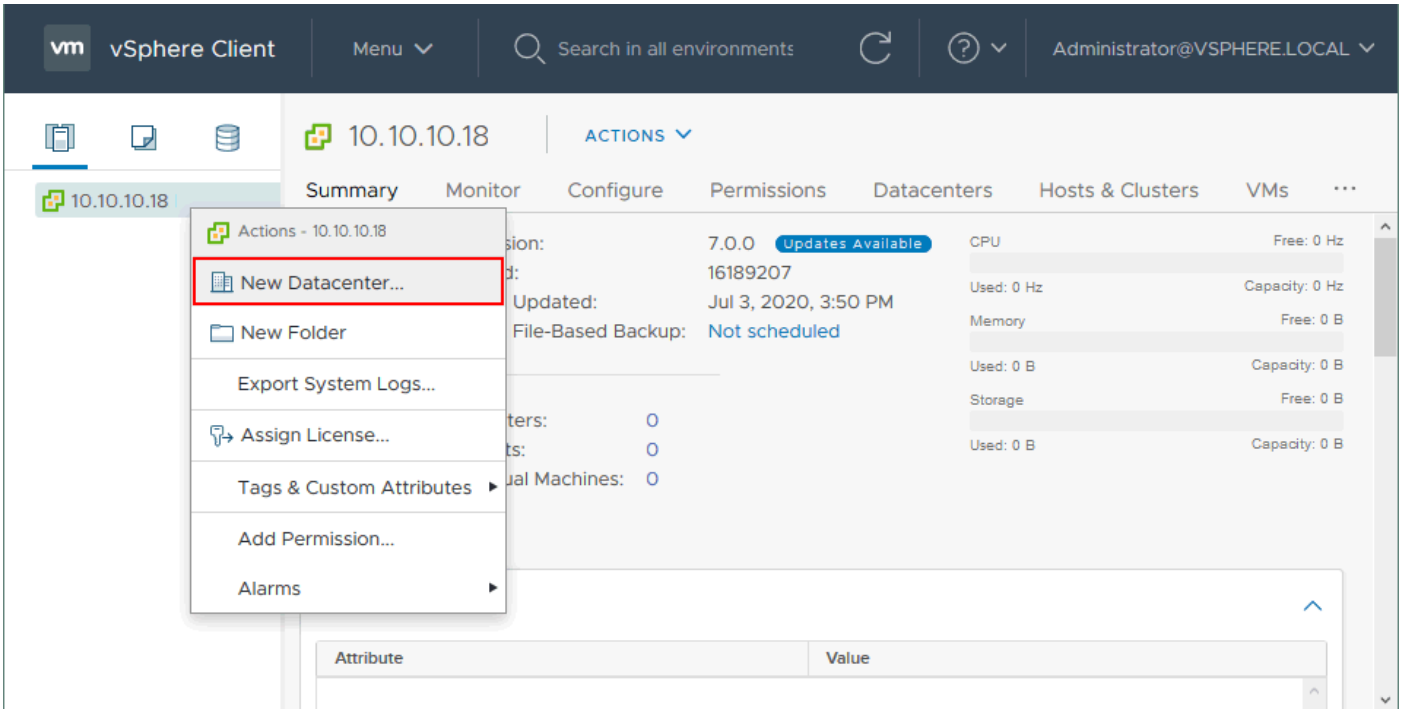


Figure 3.26

3. Type the name of your new datacenter, for example, *Datacenter1* (see *Figure 3.27*) and click **OK**.

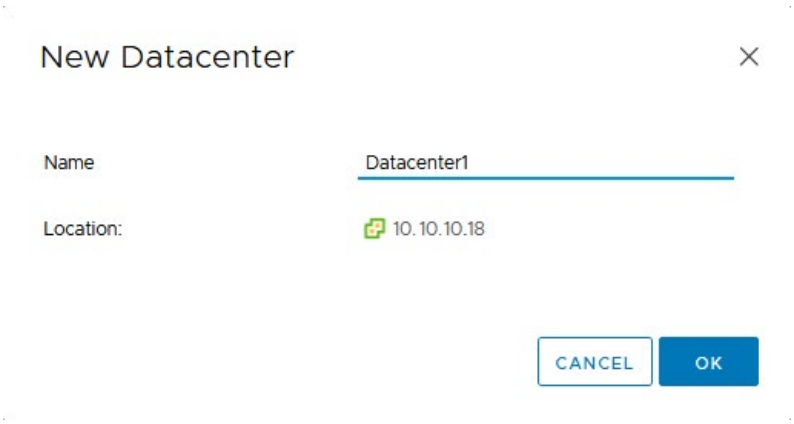


Figure 3.27

Adding ESXi hosts to the datacenter

Now you need to add ESXi hosts to the Datacenter.

Right-click on the newly-created datacenter (*Datacenter1* in our case) and select "**Add Host**" (see *Figure 3.28*).

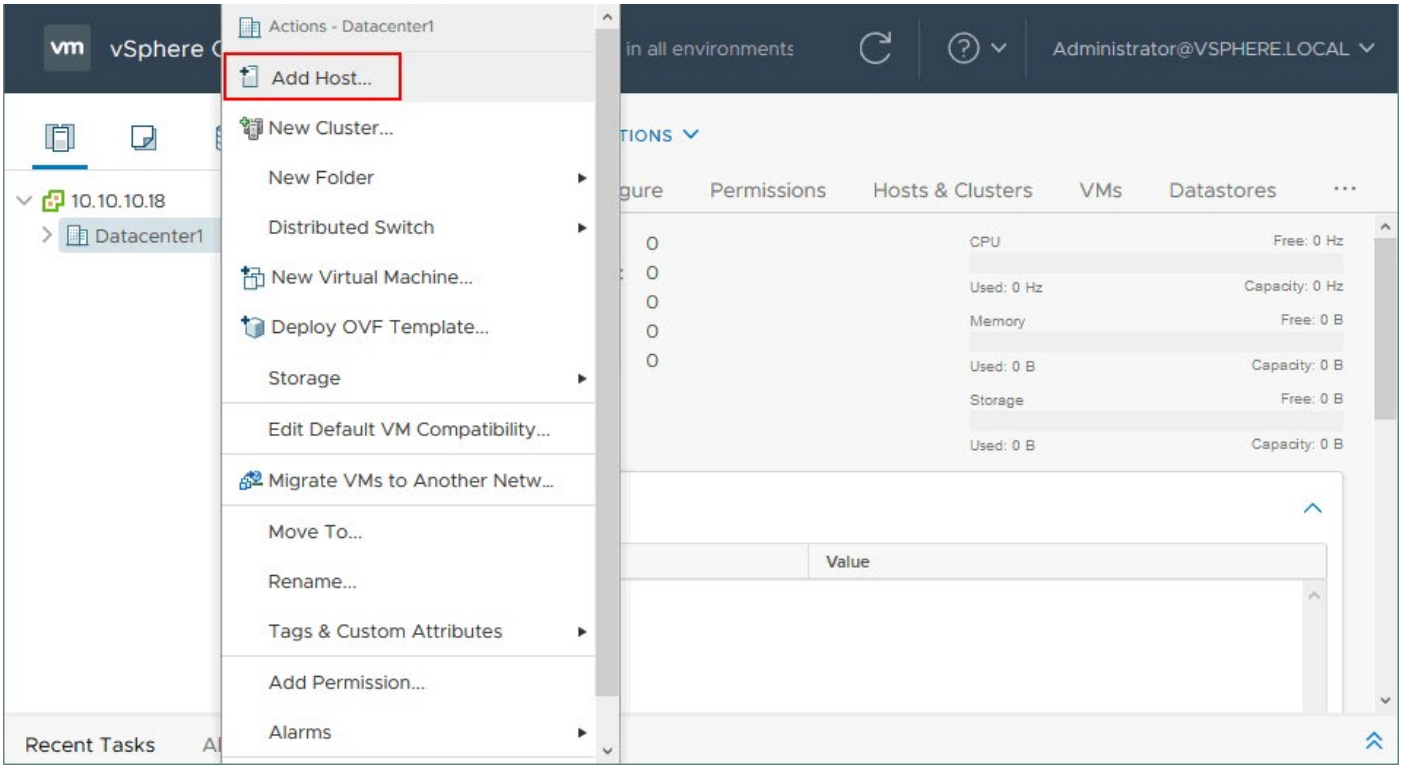


Figure 3.28

The *Add Host* wizard opens.

1. **Name and location.** Enter a hostname and IP address (see *Figure 3.29*). If you have multiple datacenters, you can select a datacenter to which you will add the host. Hit **Next** at each step to continue.

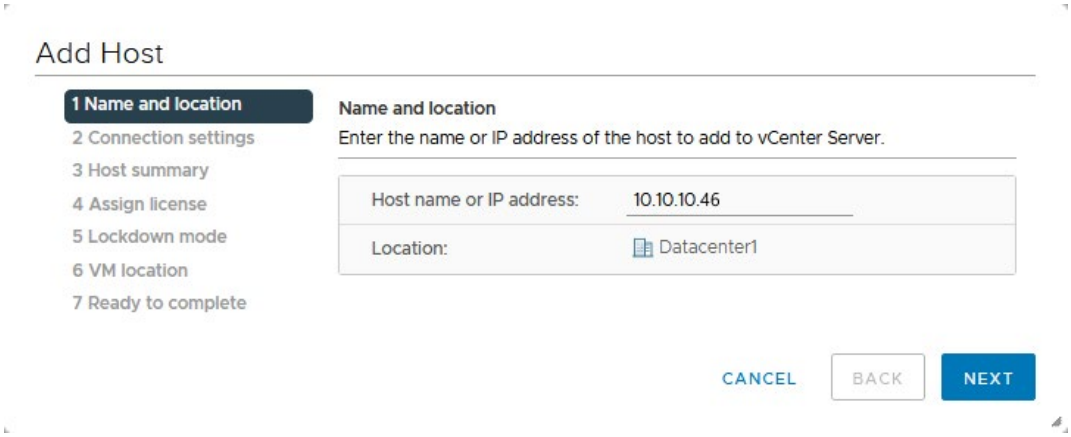


Figure 3.29

2. Connection settings. Enter a user name and password of the ESXi server (see *Figure 3.30*).

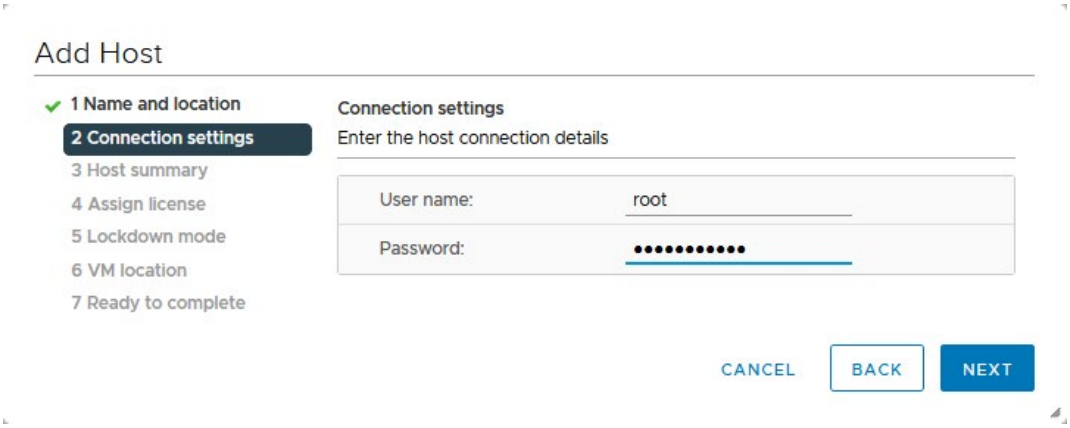


Figure 3.30

The security alert is displayed. Click **Yes** to continue (see *Figure 3.31*).

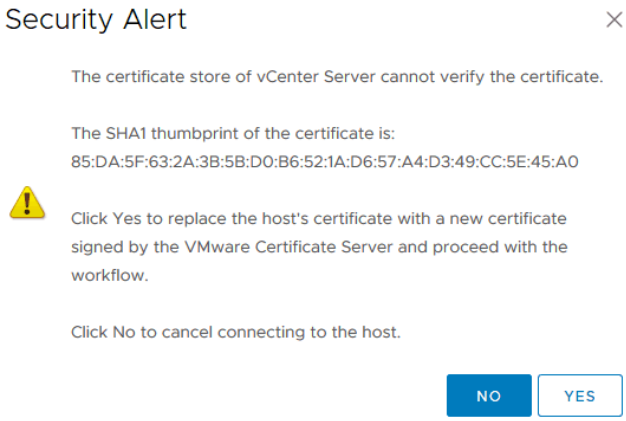


Figure 3.31

3. Host summary. Check your host summary and continue (see *Figure 3.32*).

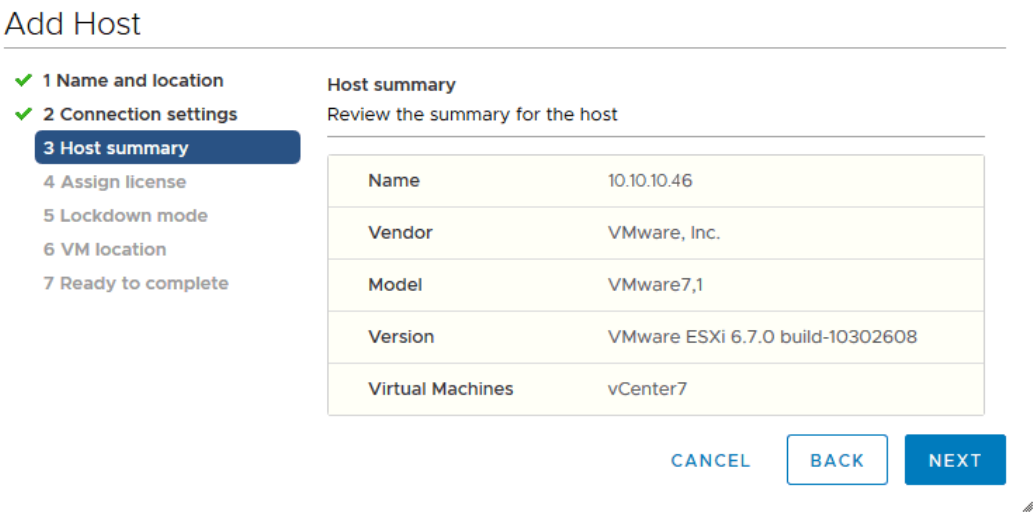


Figure 3.32

4. **Assign license.** Assign an existing or a new license to this host (see *Figure 3.33*). You can temporarily use an evaluation license for test purposes. If you have not assigned a license in VMware Host Client, you can use a license manager in vCenter and assign licenses to vCenter and each ESXi host.

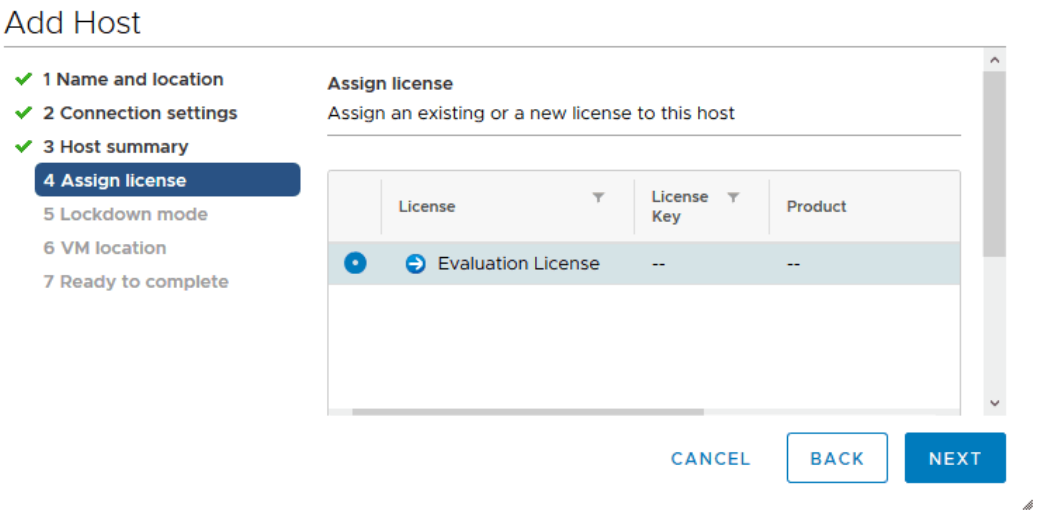


Figure 3.33

5. **Lockdown mode.** Select **Disabled** to preserve the ability to log in to the ESXi hosts directly and use VMware Host Client to manage this host (see *Figure 3.34*).

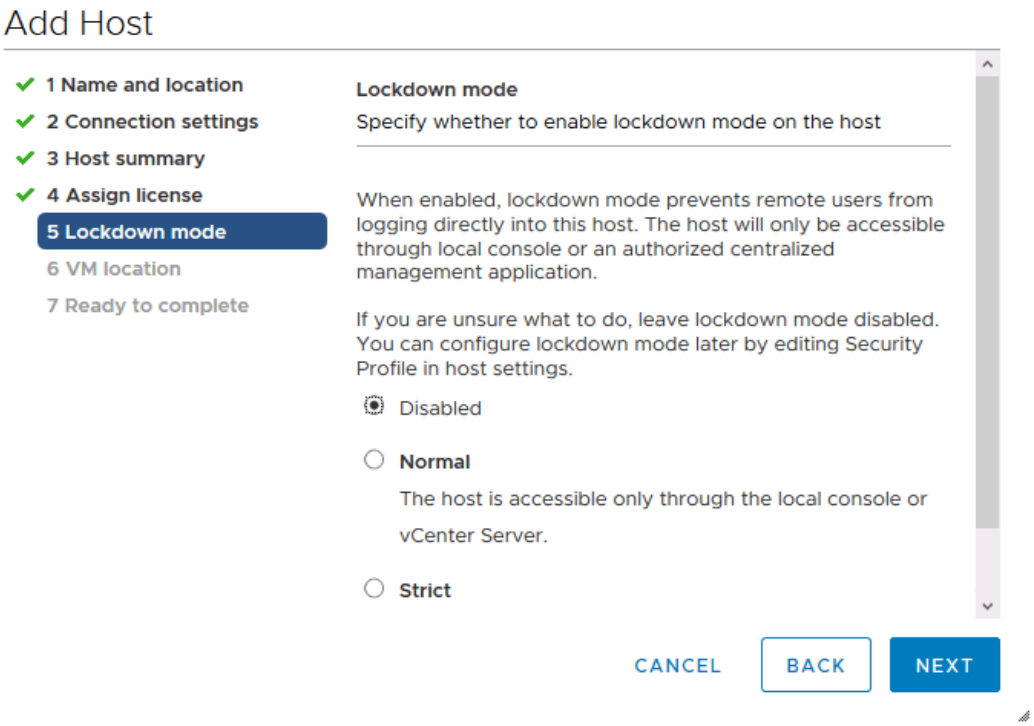


Figure 3.34

6. **VM location.** Select a datacenter (of vCenter) in which to store VMs residing on this ESXi host (see *Figure 3.35*).

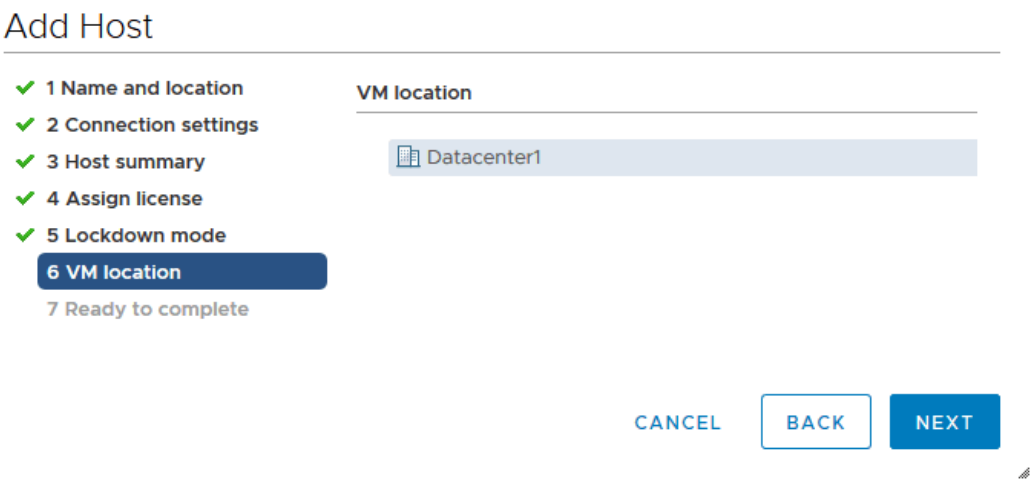
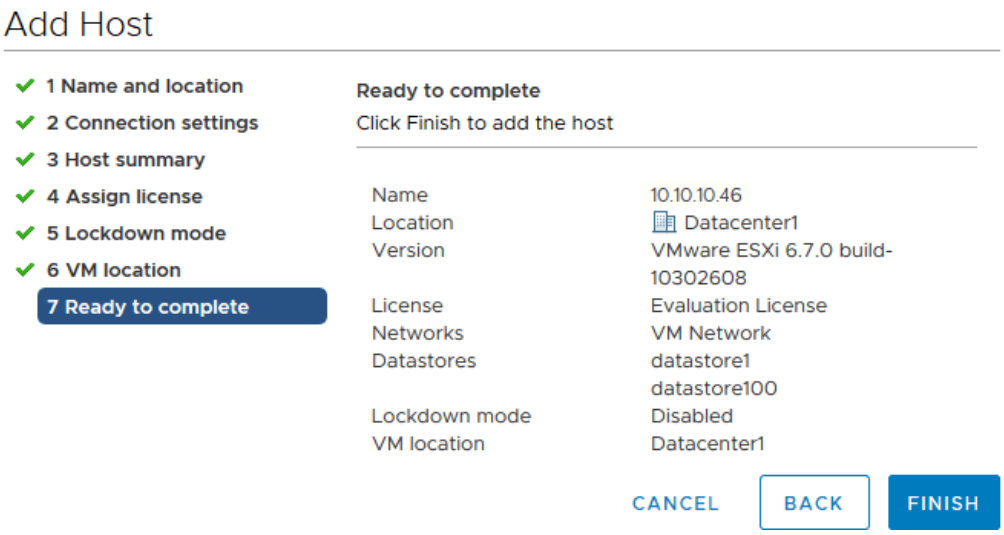


Figure 3.35

7. **Ready to complete.** Review selections and click **Finish** (see *Figure 3.36*).



Now the ESXi host is controlled by vCenter. You can see the added host in the vCenter inventory by using VMware vSphere Client (see *Figure 3.37*).

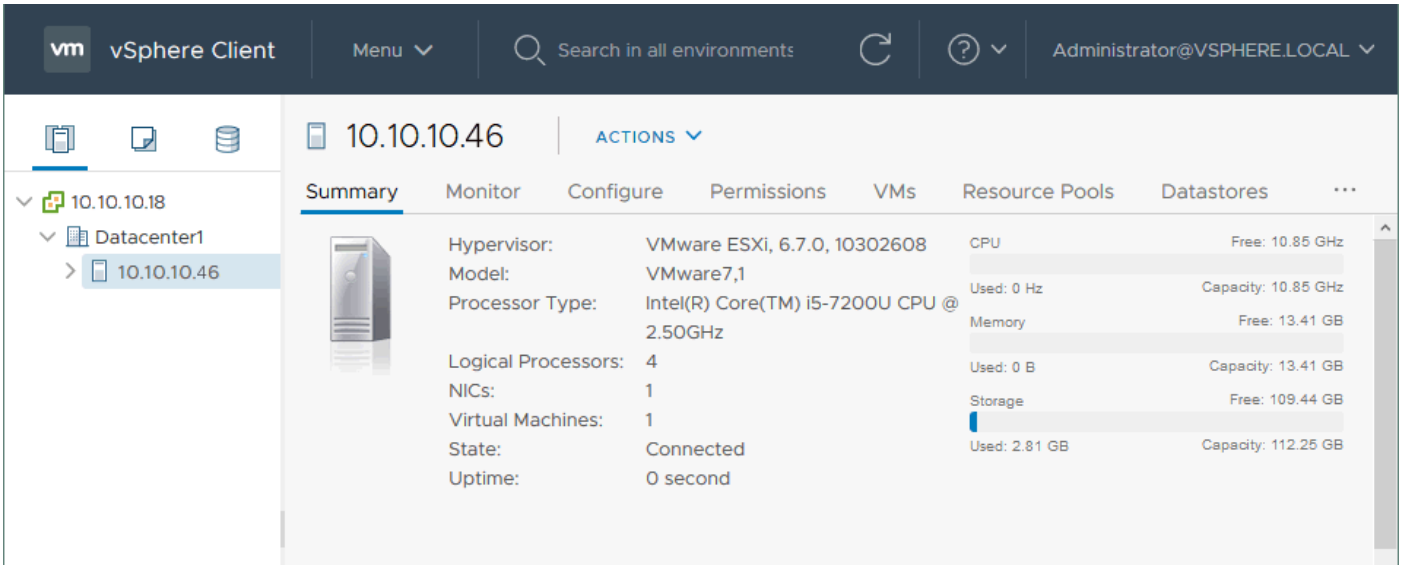


Figure 3.37

Do the same for each ESXi host that you want to add to the datacenter.

When all needed ESXi hosts have been added to vCenter (see *Figure 3.38*), you can go to the next stage of preparing vSphere configuration for a cluster. You need to configure a storage network and connect shared storage before creating a new cluster.

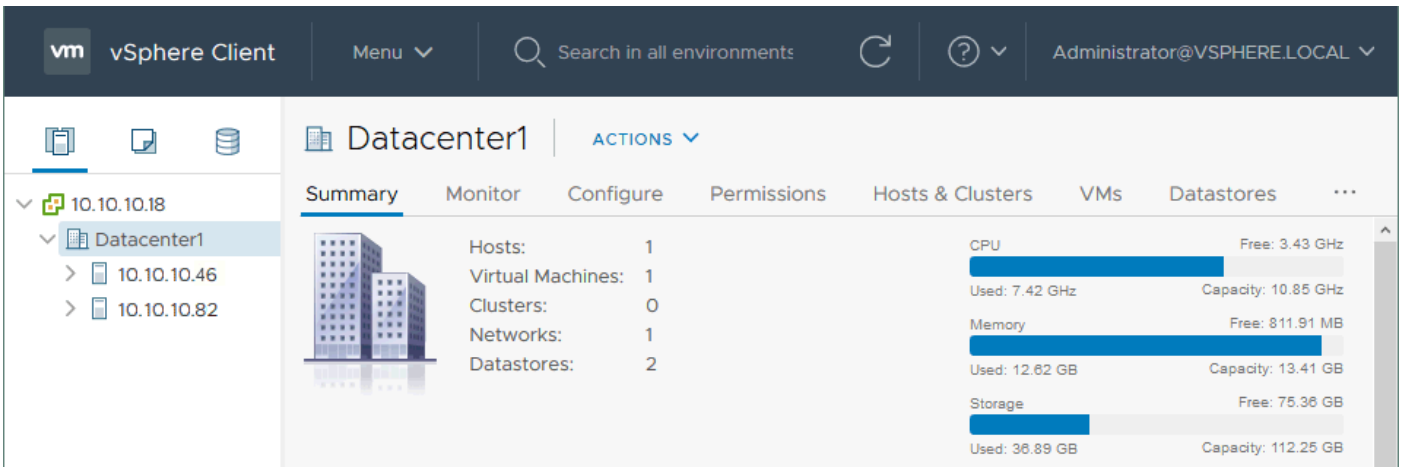


Figure 3.38

ESXi Networking and Storage Configuration

Networking is an important point in VMware vSphere environments and configuring networks is crucial for building a VMware cluster. You can use multiple networks and use each network for a different purpose.

Network types:

- Management network
- Storage network
- VM migration network (vMotion)
- Fault tolerance logging

If more than two Network Interface Controllers (NICs) are available in the ESXi server, create two virtual switches. One of them should host the Service Console and VMkernel. The other one should be dedicated to virtual machine traffic. The recommended practice is to also use separate networks for shared storage and vMotion traffic. The NICs carrying the iSCSI traffic should be connected to redundant Ethernet switches (see *Figure 4.1*).

There are two types of virtual switches in vSphere: standard and distributed. The standard switch is configured manually on each host and is used for small environments. The distributed switch allows managing networks for multiple hosts from a single vCenter interface and is useful when you have a high number of ESXi hosts. If you manage two or three ESXi hosts in your vCenter, you can use the standard switch.

Note
The recommended redundant storage network scheme is as follows: two ESXi hosts connected via the redundant network to a SAN with two storage processors (see *Figure 4.1*). However, you can use a NAS for this purpose.

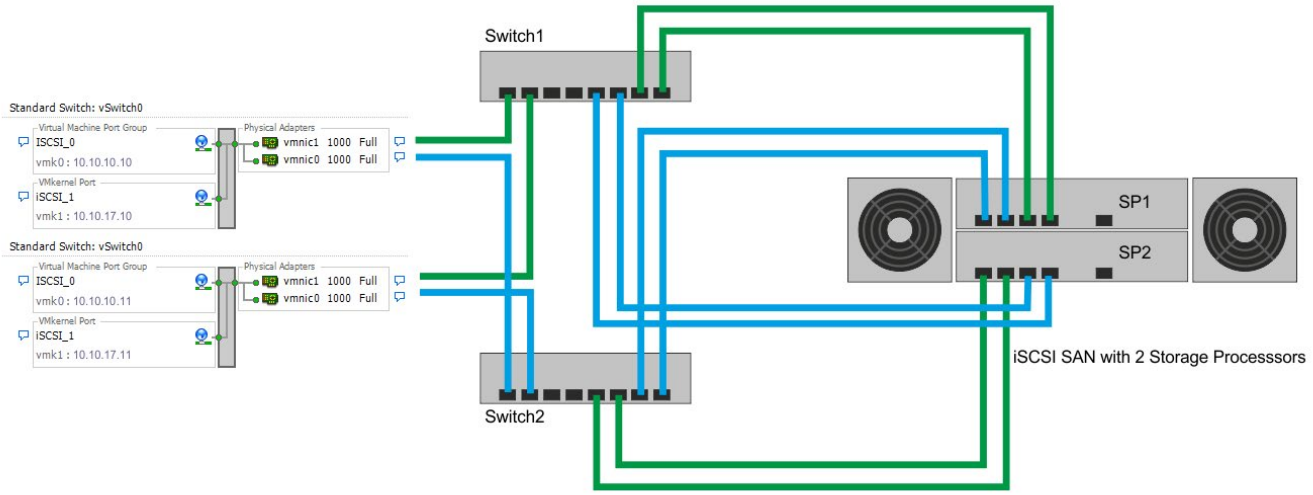


Figure 4.1

Below you can see the example of the vMotion virtual network scheme on an ESXi host (see *Figure 4.2*). In this example, adapters are mapped by using a single virtual switch. This connection scheme is applicable for a vMotion network, storage, Fault Tolerance logging, and other networks in a cluster.

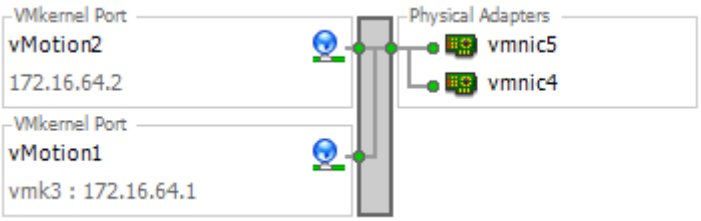


Figure 4.2

As an alternative, you can use adapter mapping on separate virtual standard switches (see *Figure 4.3*).

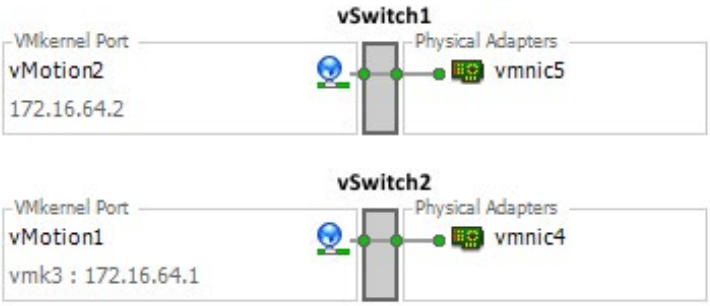


Figure 4.3

vMotion is a feature required for hot migrating powered-on virtual machines from one ESXi host to another. Enable vMotion if you want to create a DRS or HA cluster. Use separate networks for vMotion, Storage and Production. This approach can help you prevent overloading and reduce network bandwidth (see *Figure 4.4*). The production network is a network to which your physical computers (workstations), routers, printers, etc., are connected.

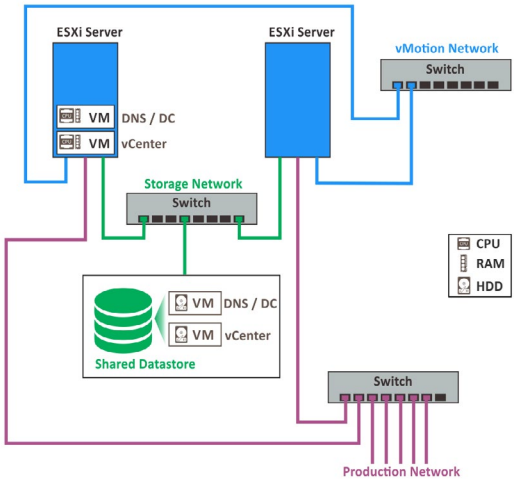


Figure 4.4

Using Shared Storage in vSphere

Shared storage is a requirement for deploying a vSphere cluster. You can use iSCSI or NFS shared storage to store VM files in the cluster. iSCSI storage sharing protocol provides block-level storage, and NFS is a file-level network sharing protocol.

In simple words, a VM is a set of files located on a datastore and a process (processes) running on the ESXi host (consuming CPU and memory resources of the host). When we run a VM in a cluster, VM files are stored on the same datastore, which is shared between all ESXi hosts of the cluster. VM processes can run on different ESXi hosts, but VMs use the files located on the same storage at the same time. The datastore is accessible for all cluster nodes simultaneously.

Shared Storage Configuration

Prepare two physical network adapters on each ESXi host for connecting to shared storage via the network. Two network adapters are used to provide network redundancy. These two network adapters must not be used for other networks.

Let's configure the first ESXi host, that is, the first cluster node.

Adding a virtual switch

First, you need to add a virtual switch. We carry out all operations on the first ESXi host to configure shared storage in VMware Host Client.

1. Open the web interface of VMware Host Client (10.10.10.46 in our case).
2. Click **Networking** in the **Navigator** pane and select the **Virtual switches** tab. You should see *vSwitch0*, which was created by default when we installed ESXi.
3. Click **Add standard virtual switch** (see *Figure 4.5*).

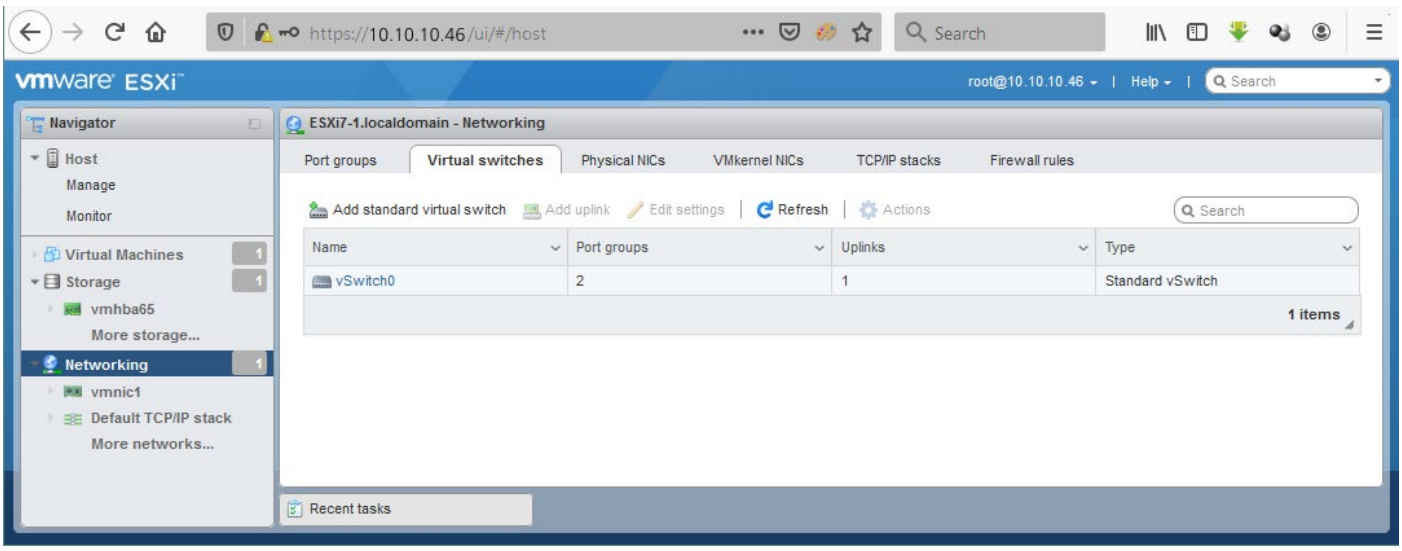


Figure 4.5

4. A new vSwitch configuration window opens (see *Figure 4.6*). Edit the configuration as follows.

   vSwitch name: *vSwitch1-storage*

   MTU: 1500

   Enable Jumbo frames - change from 1500 to 9000 bytes for better performance.

> **Note**
> Your network equipment must support Jumbo frames. Jumbo frames help improve performance when transferring large amounts of data over the network by increasing the carried packet size and reduce overheads.

   Uplink: *nic1* (select a physical network adapter)

   You can leave the default values for other settings.

5. Click **Add** to save settings and add a new vSwitch.

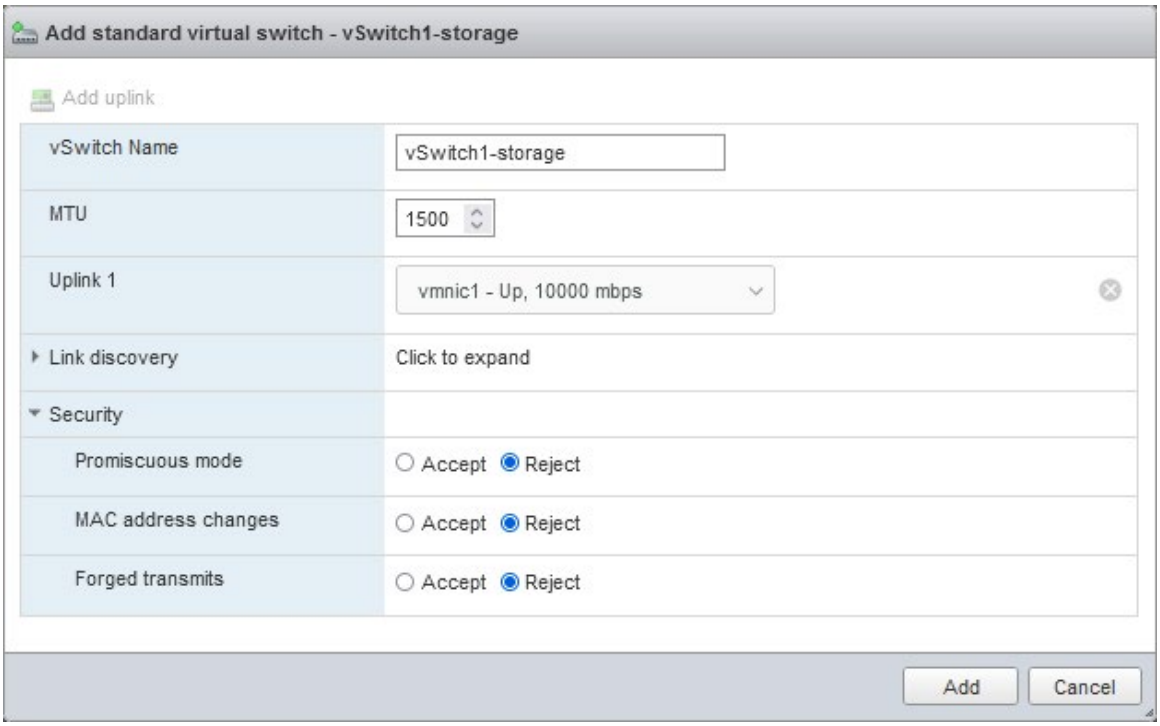


Figure 4.6

A new vSwitch (whose name is *vSwitch1-storage* in our case) has been created and is now displayed in the Virtual switches tab.

Adding an uplink

Now we need to add the second uplink by connecting the second physical network adapter to our virtual switch (vSwitch1-storage). With two uplinks, we can enable NIC teaming.

1. Select the *vSwitch1-storage* virtual switch and click **Edit settings** to edit virtual switch settings (see *Figure 4.7*).

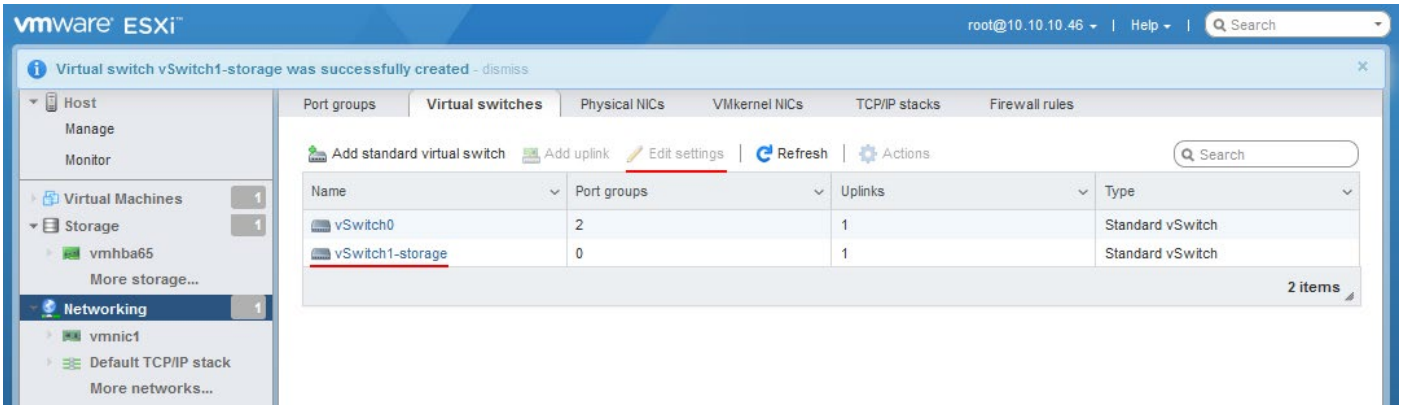


Figure 4.7

2. The *Edit standard virtual switch* window opens (see *Figure 4.8*). Click **Add uplink**.

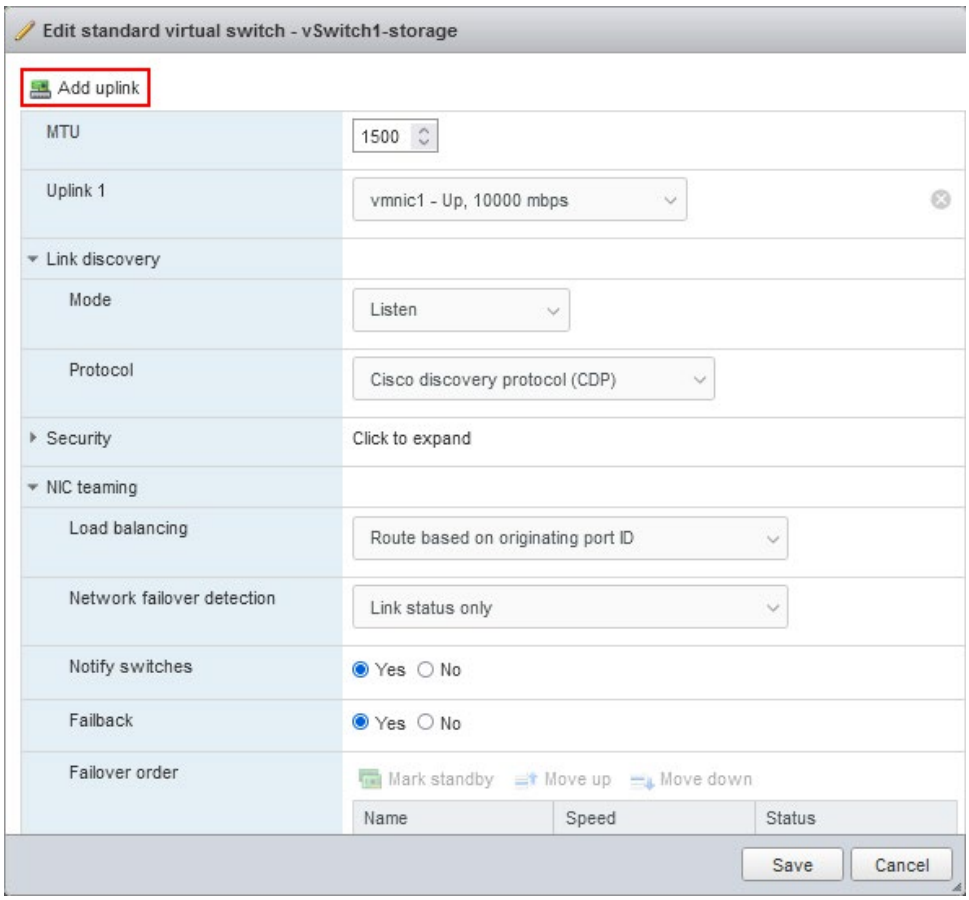


Figure 4.8

Enable NIC teaming

The *Uplink 2* line has been added. Select *nic2* in the drop-down list for this option (the next free/unused network adapter is selected automatically in this list).

Click **NIC Teaming** to expand NIC teaming settings (see *Figure 4.9*).

In the *Failover order* section, make sure that both NICs are configured as **Active** (have the Active status). To change the NIC status, select a NIC and click **Mark Active/Mark Standby**.

Click **Save** to save virtual switch settings and close the window.

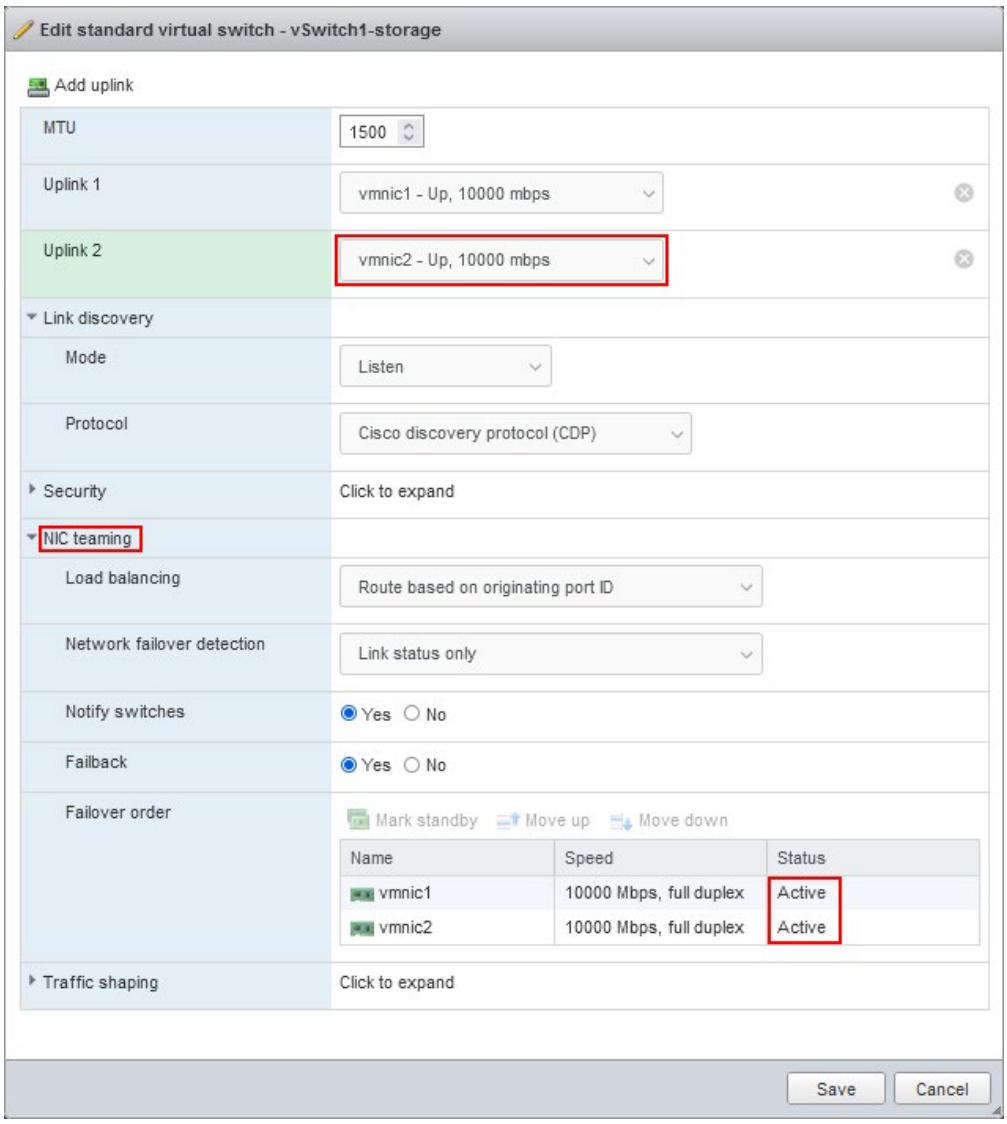


Figure 4.9

We have connected two physical network adapters to our virtual switch intended for connecting to shared storage (see *Figure 4.10*).

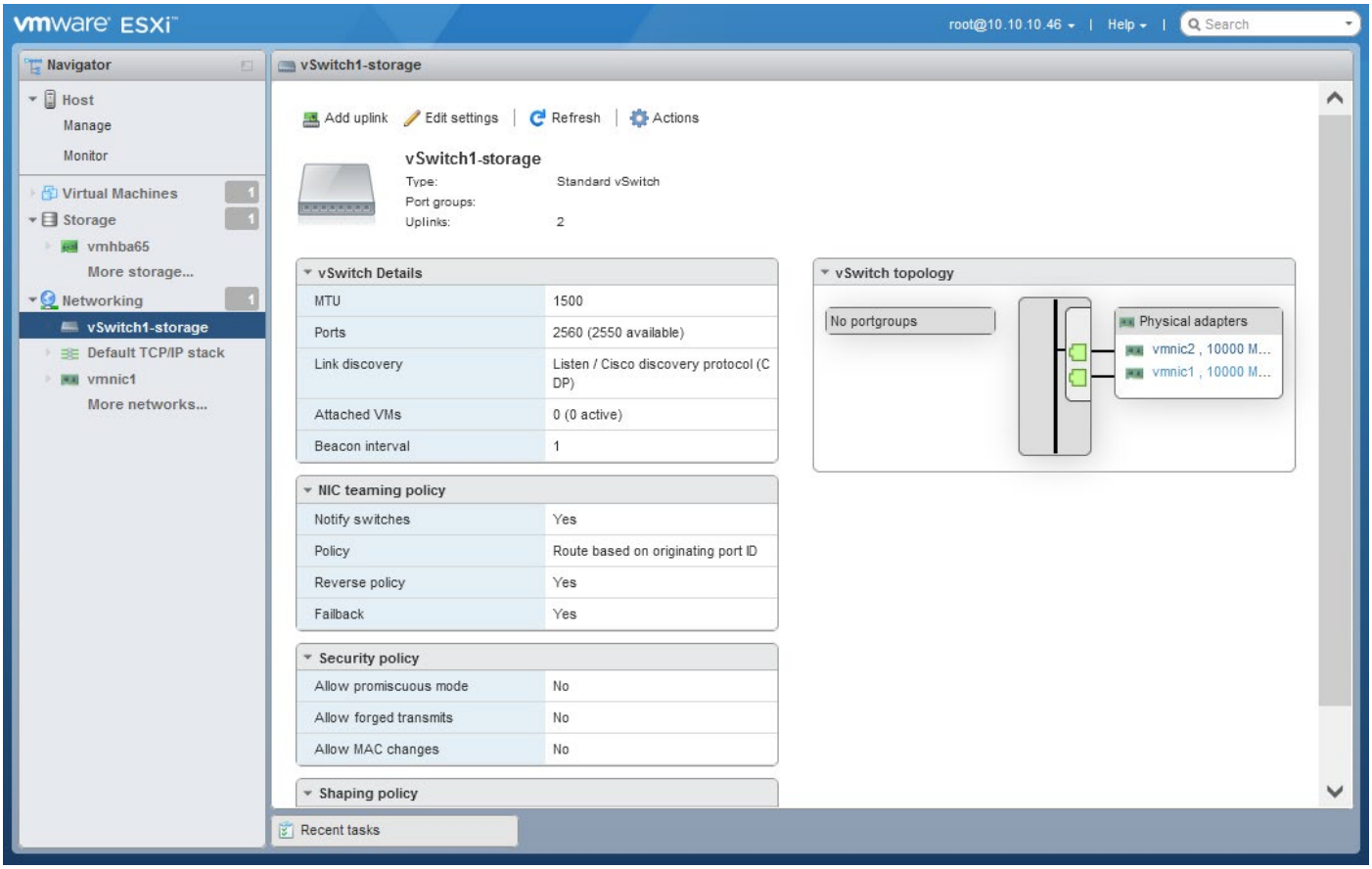


Figure 4.10

Add a storage network

VMkernel is the main element of an operating system running on an ESXi hypervisor as a type-1 hypervisor used to create, run, and manage virtual machines, as well as work with device drivers, I/O stacks, and resource scheduling. VMs use VMkernel to communicate with the physical hardware of an ESXi server. VMkernel controls services such as vMotion, shared storage, fault tolerance, and traffic management. When we use shared storage we need to connect shared storage to VMkernel by using a VMkernel network adapter.

1. Add a new VMkernel NIC to the virtual switch (vSwitch1-storage) for the storage network.

   Click **Networking** in the **Navigator** pane and select the **VMkernel NICs** tab (see *Figure 4.11*). You can see *vmk0*, which is a VMkernel network adapter created automatically when you installed ESXi.

   Click **Add VMkernel NIC**.

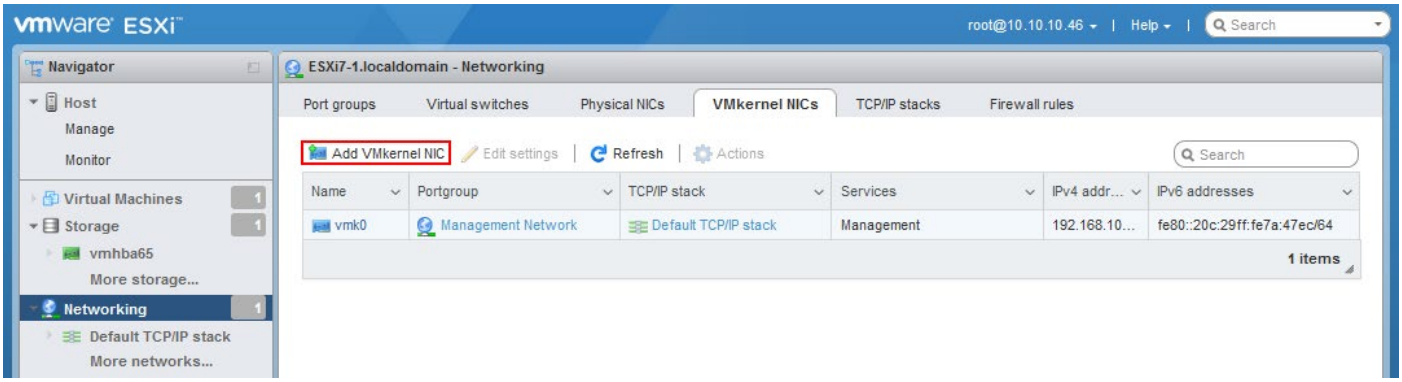


Figure 4.11

2. The *Add VMkernel NIC* window opens (see *Figure 4.12*).

   Define the following parameters.

   - Port group: *New port group*

   Set a name for your new port group.

   - New port group: *Storage-PGroup*

   Select the virtual switch you have created before for connecting to shared storage.

   - Virtual switch: vSwitch1-storage

   Enter a VLAN ID if needed. We don't use VLANs in this example.

   - VLAN ID: 0

   If you use Jumbo frames, set MTU to 9000. Otherwise, leave the default value of 1500 bytes. It is recommended that you use Jumbo frames for a storage network.

   - MTU: 1500
   - IP version: IPv4 only

   Click IPv4 settings to expand the settings.

   We need to use static IP configuration and manually set an IP address and network mask. Enter an IP address and netmask according to your storage network configuration (192.168.105.0/24 is the storage network in our example).

   - Configuration: static
   - Address: 192.168.105.46
   - Subnet mask: 255.255.255.0
   - TCP/IP stack: Default TCP/IP stack

   Services: You can select the needed VMware services for the current network and VMkernel adapter.

   Click **Create** to save settings and create a new VMkernel network adapter.

Add VMkernel NIC

Port group	New port group
New port group	Storage-PGroup
Virtual switch	vSwitch1-storage
VLAN ID	0
MTU	1500
IP version	IPv4 only
IPv4 settings	
Configuration	DHCP Static
Address	192.168.105.46
Subnet mask	255.255.255.0
TCP/IP stack	Default TCP/IP stack
Services	vMotion Provisioning Fault tolerance logging Management Replication NFC replication

Create Cancel

Figure 4.12

The new *vmk1* VMkernel network adapter has been created (see *Figure 4.13*).

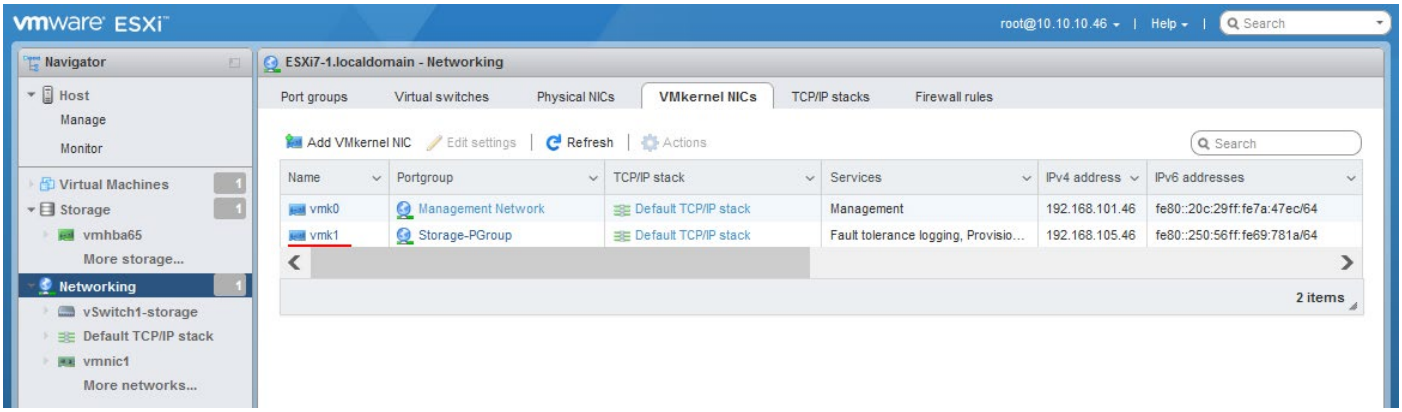


Figure 4.13

Adding a port group

Now we need to add a port group to this virtual switch (*vSwitch1-storage*).

1. Go to **Networking > Port Groups**, select the needed port group (*StoragePGroup* in our case) and click **Edit settings** to edit port group settings (see *Figure 4.14*).
2. Edit the following settings.
   - Override failover order: **Yes**

   In the failover order section, set the first network adapter as **Active** and set the second one as **Unused**.

   1 - **Active**

   2 - **Unused**

   Click **Save** to save port group settings.

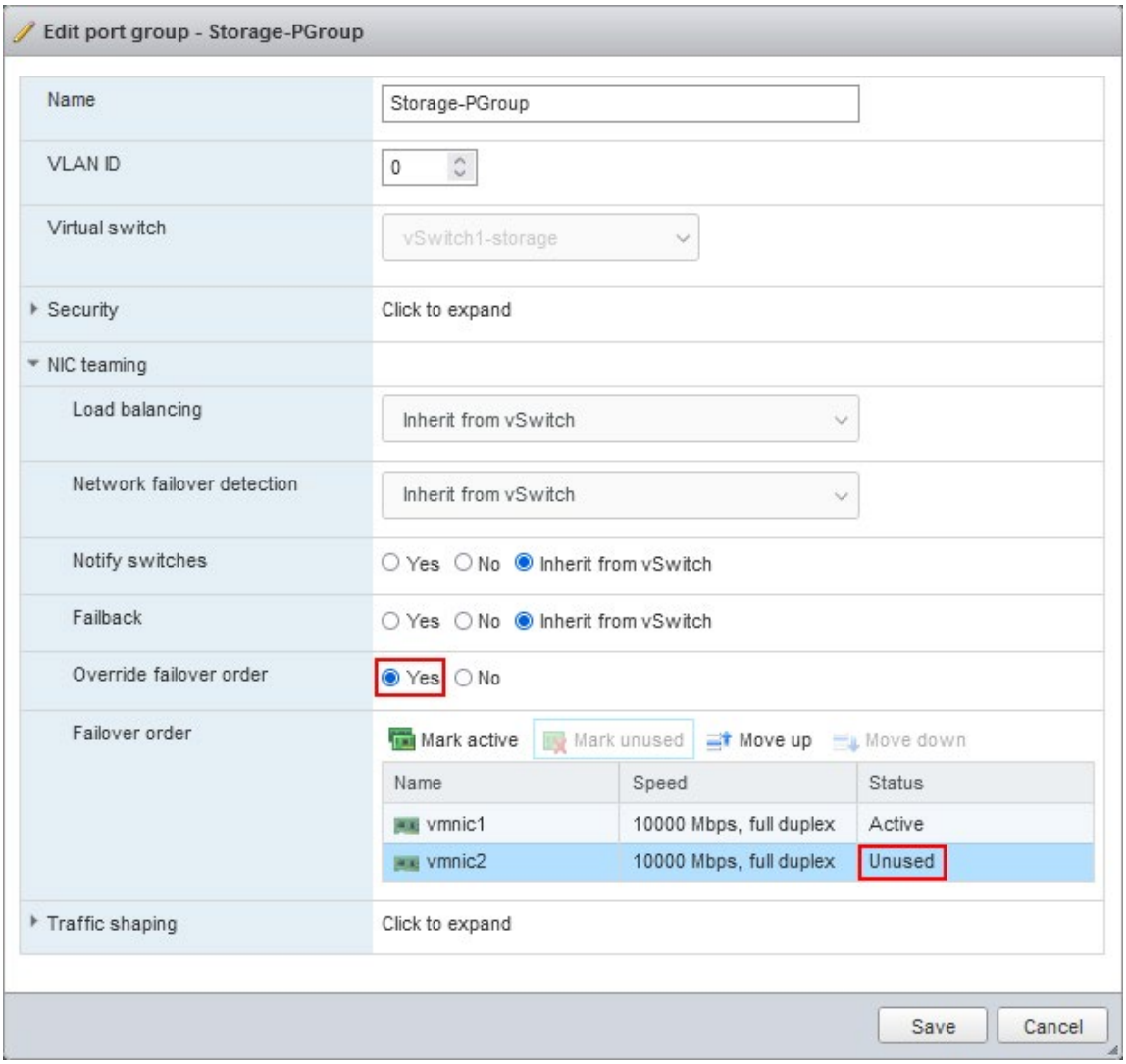


Figure 4.14

Note
If you don't change these settings, you may get an error when mapping an iSCSI target to a software iSCSI storage adapter.

Adding an iSCSI adapter

Now you need to add an iSCSI software adapter.

1. Click **Storage** in **Navigator** and select the **Adapters** tab.

   Click **Software iSCSI** to add a software iSCSI adapter or edit the settings of an existing adapter (see *Figure 4.15*).

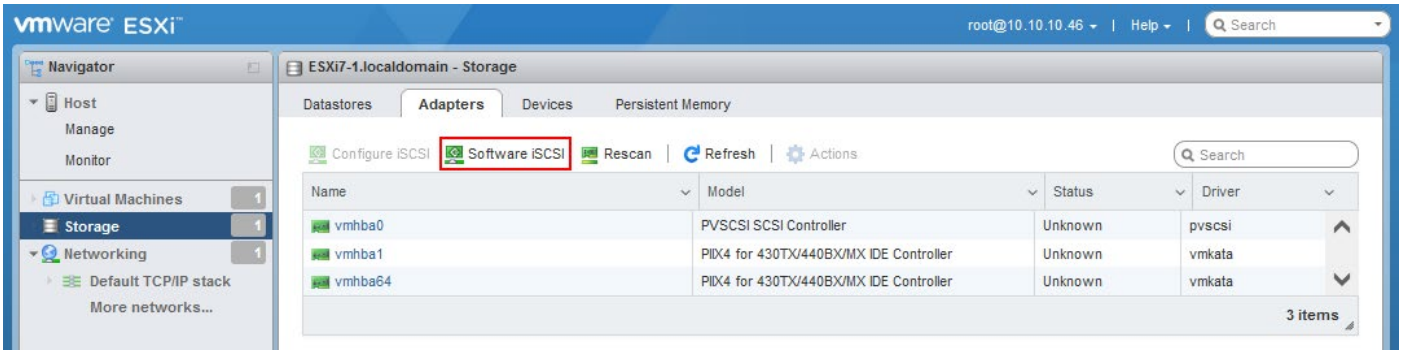


Figure 4.15

2. The *Configure iSCSI* window opens (see *Figure 4.16*).

   Configure iSCSI as follows.

   - iSCSI enabled: Enabled

   Click **Add port binding** in the *Network port bindings* section.

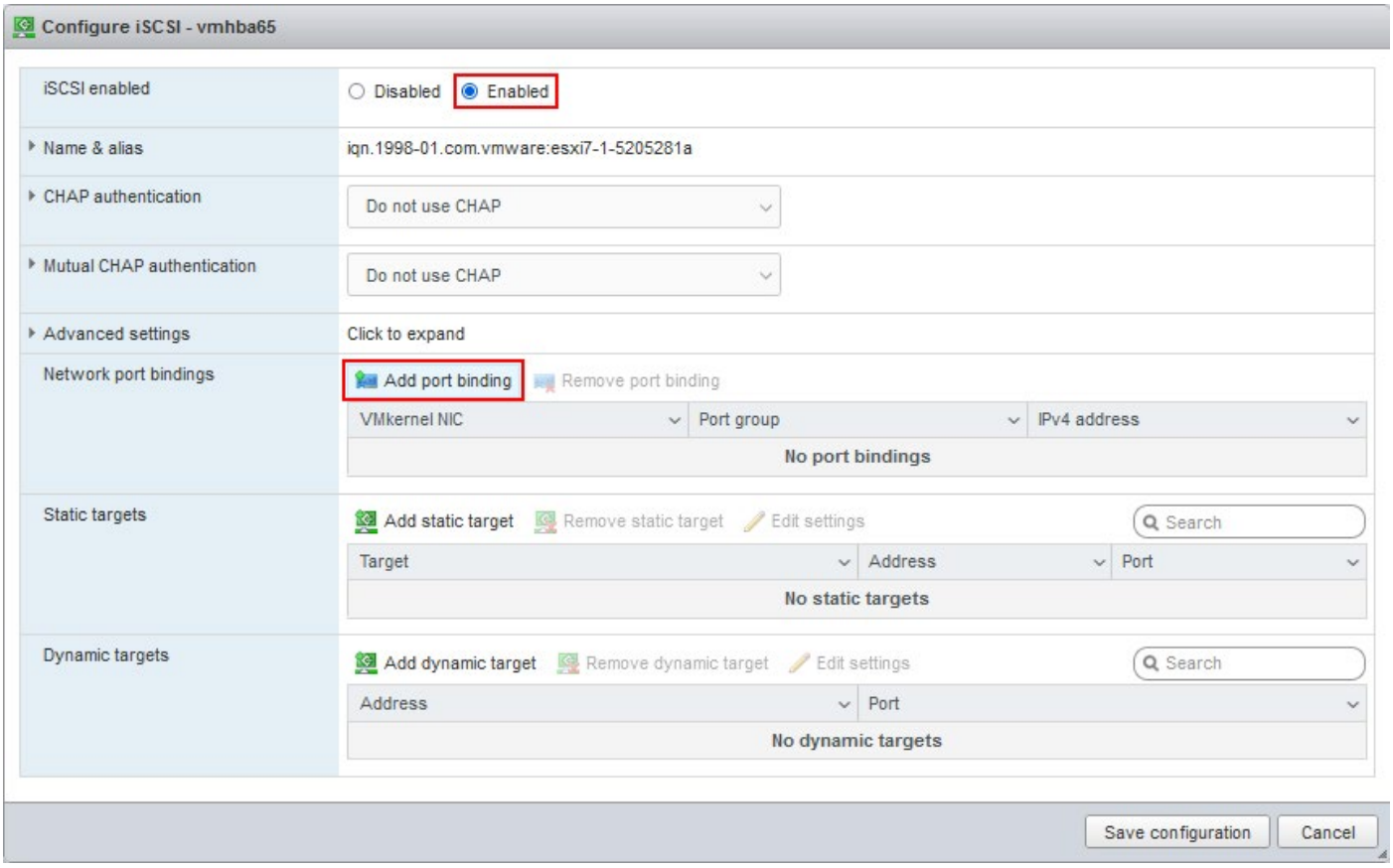


Figure 4.16

3. Select a VMkernel interface (*vmk1*) which IP address is 192.168.105.46 in our example (see *Figure 4.17*). Click **Select**.

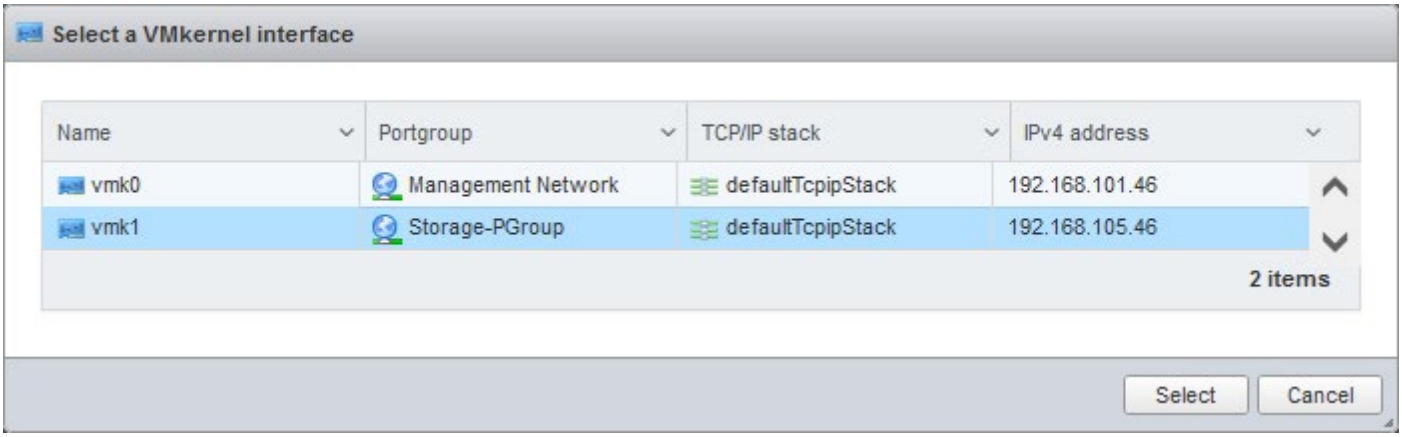


Figure 4.17

Port binding has been added (see *Figure 4.18*). You can click **Save Configuration** to check whether everything is correct and then open this configuration window once again.

Note
If two physical network adapters of your ESXi host are connected to a virtual switch used for the storage network, and both adapters are Active/Active, you can get the error:
Failed! The VMkernel virtual NIC adapter vmk1 has multiple physical uplinks.

This error occurs if you haven't set a port group for this virtual switch to use adapters as *Active/Unused* and override failover order (Yes) in the NIC teaming settings of the port group connected to a VMkernel adapter of this virtual switch.

This error should disappear if you remove one physical adapter from the NIC teaming configuration of a virtual switch. However, if you remove one physical network adapter, you will get a warning that the network redundancy is lost. Note that a configuration without network redundancy is not recommended for use in a production cluster.

After successful configuration of network port bindings, you can add iSCSI targets.

You can use dynamic targets and define an IP address for your NAS where an iSCSI share (iSCSI target) is configured.

Note
Your ESXi hosts must be allowed in iSCSI and firewall configuration of SAN/NAS for connecting to the iSCSI target you will use as storage for a vSphere cluster.

Click Add dynamic target in the *Dynamic targets* section of the *Configure iSCSI* window (see *Figure 4.18*).

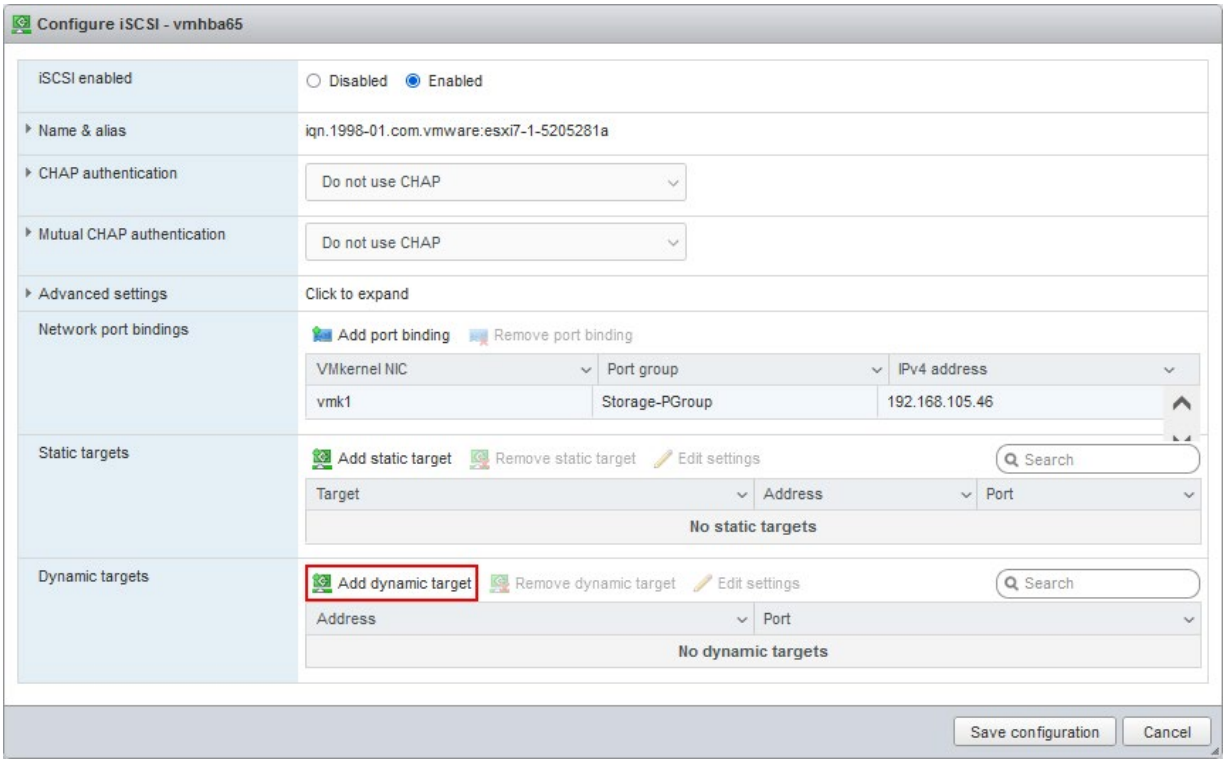


Figure 4.18

Enter an IP address of your NAS or SAN where you have configured an iSCSI target (see *Figure 4.19*). The default port for connecting to an iSCSI target is 3260. Adjust the port number if you use a custom port number on your SAN or NAS. In our example, we add *192.168.105.228* port *3260*.

Click **Save Configuration**.

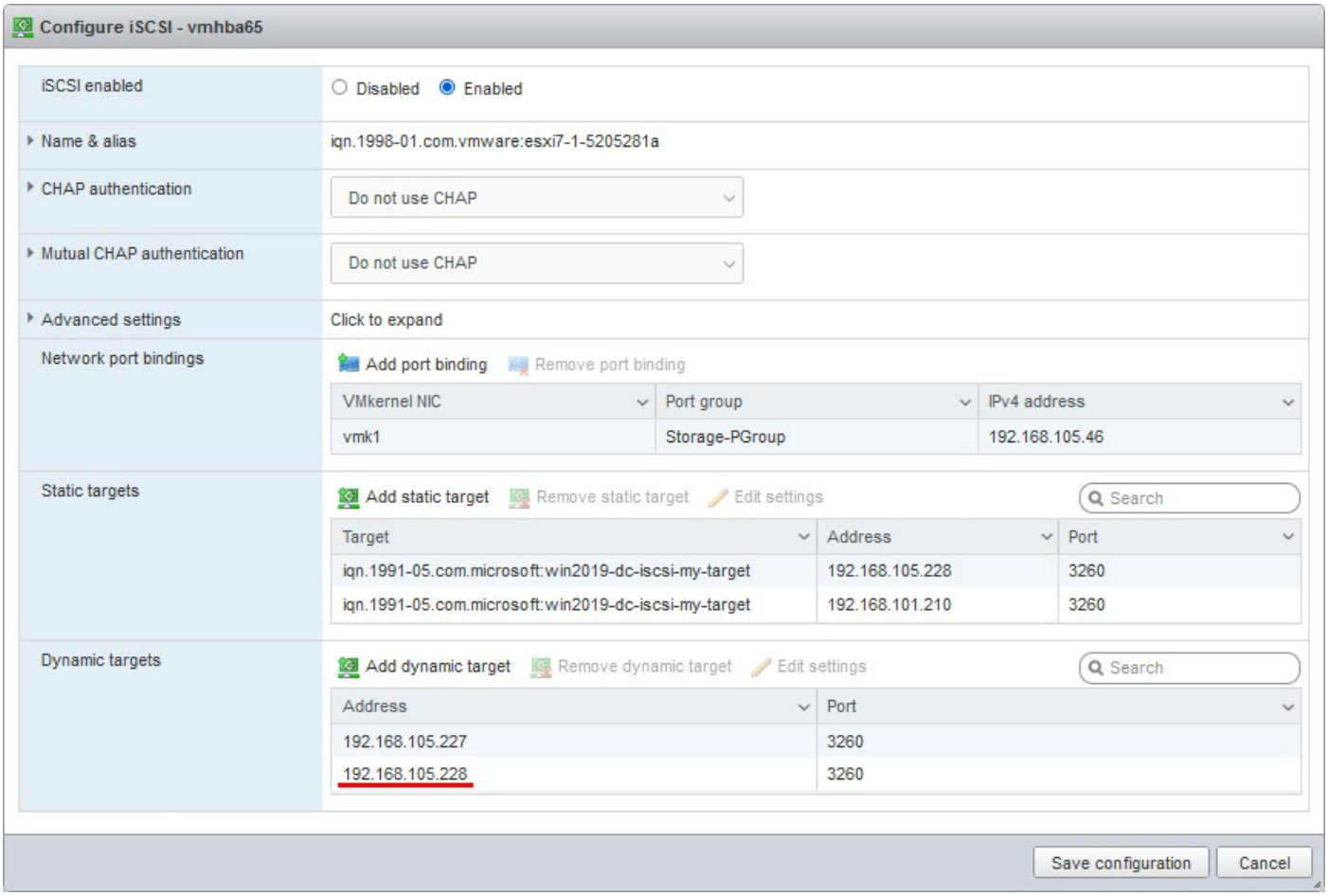


Figure 4.19

Your iSCSI adapter is now displayed in the adapters list (see *Figure 4.20*).

Click **Rescan adapters** to make sure your new configuration is applied.

If you open your software iSCSI adapter settings once again, you should see the IQN of your iSCSI target in the *Static targets* section (see *Figure 4.19*).

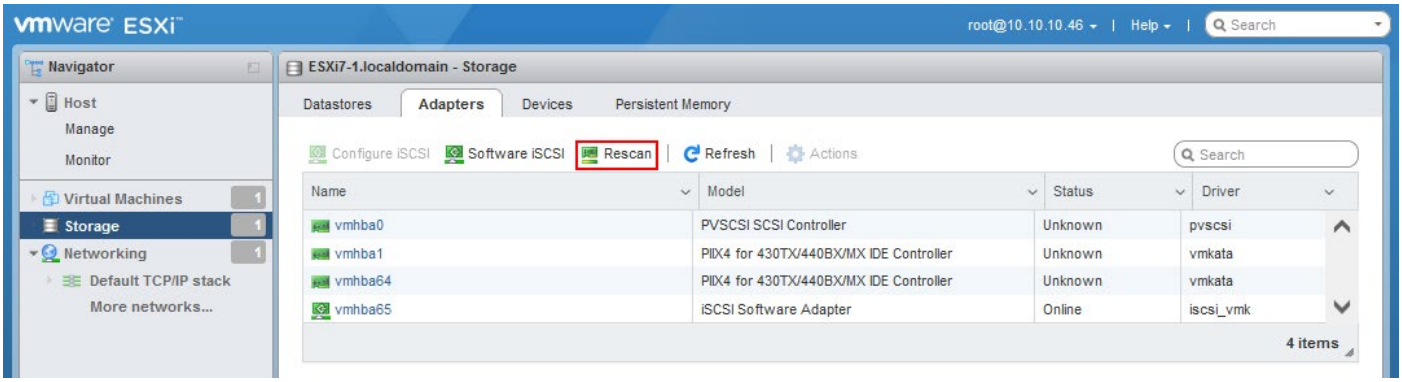


Figure 4.20

Click **Storage** in the **Navigator** pane and select the **Devices** tab to see all available storage devices on your ESXi host (see *Figure 4.21*).

You should see the iSCSI disk you have added before in this list.

It means that you can now use this iSCSI disk to create a VMFS datastore.

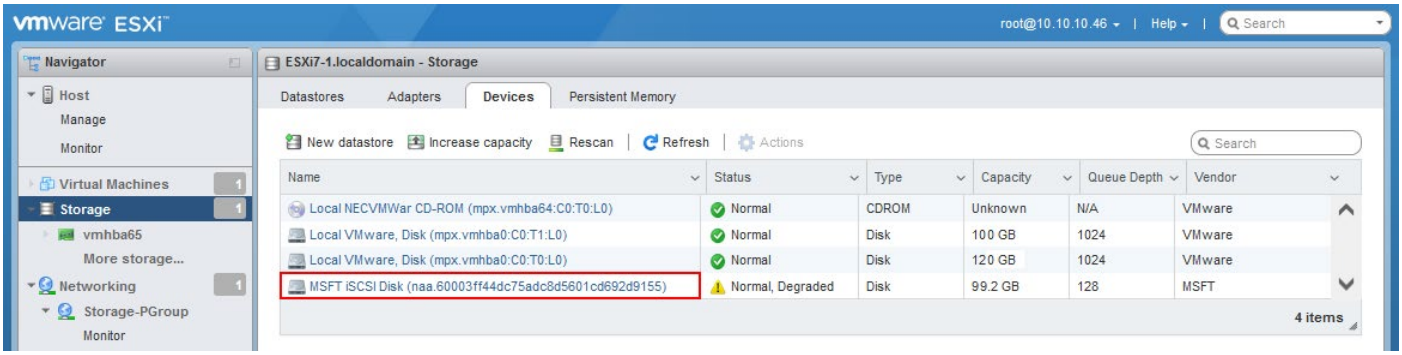


Figure 4.21

Creating a datastore

Once you have connected shared iSCSI storage to an ESXi host, you can create a VMFS datastore on this storage.

1. Go to **Storage** in **Navigator** and select the **Datastores** tab (see *Figure 4.22*).
2. Click **New datastore** to create a new datastore.

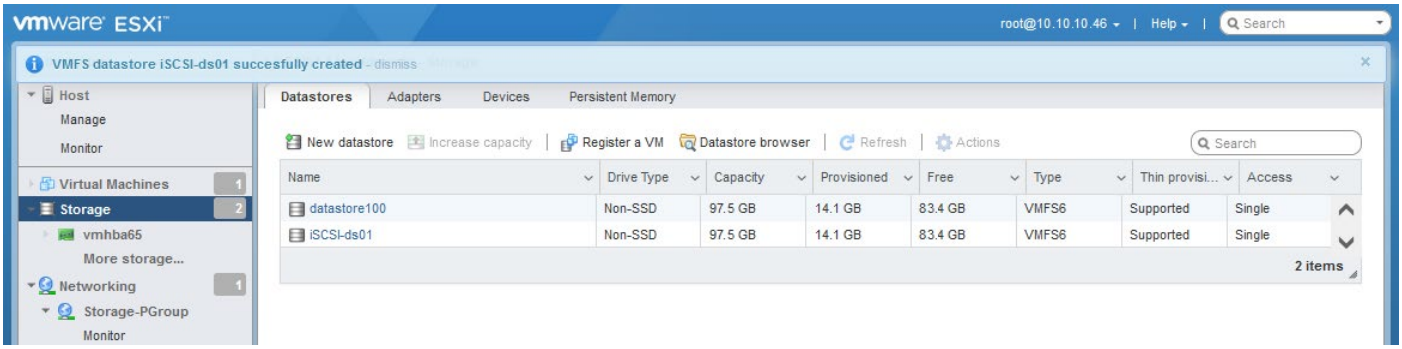


Figure 4.22

3. The *new datastore* wizard window opens (see *Figure 4.23*).

   1. Select creation type. Click **Create new VMFS datastore**. Hit **Next** at each step of the wizard to continue.

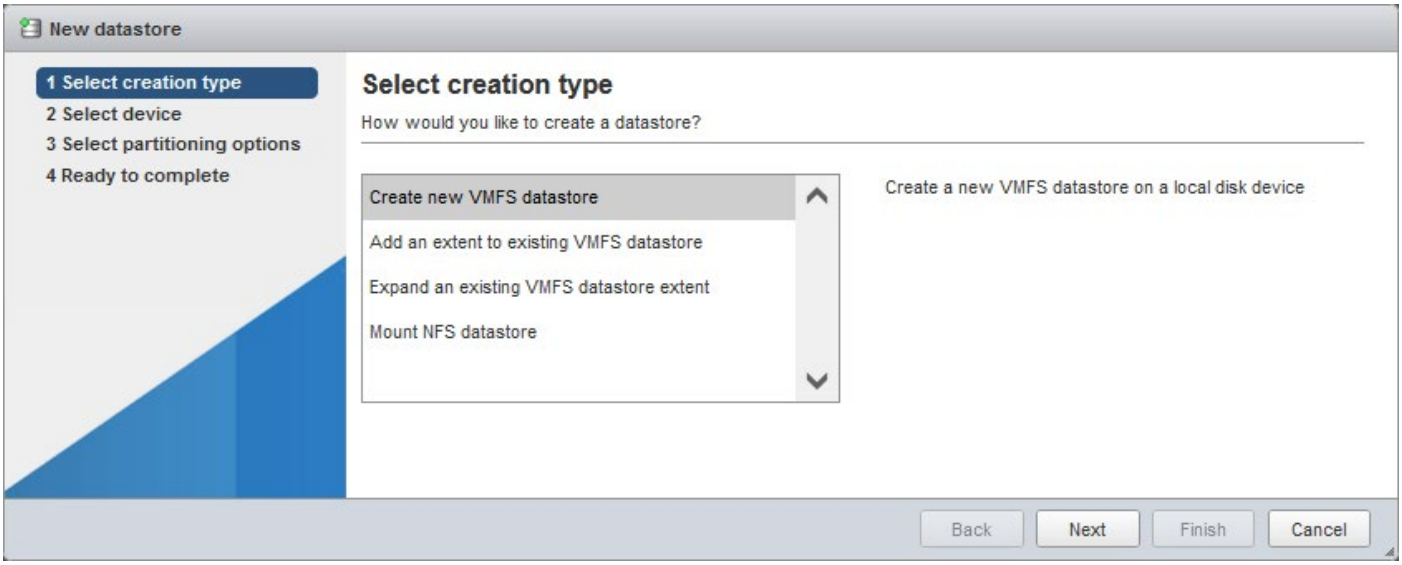


Figure 4.23

2. Select device. Enter the datastore name, for example, *iSCSI-ds01* and select the iSCSI disk that you want to use as the shared storage for a cluster (see *Figure 4.24*).

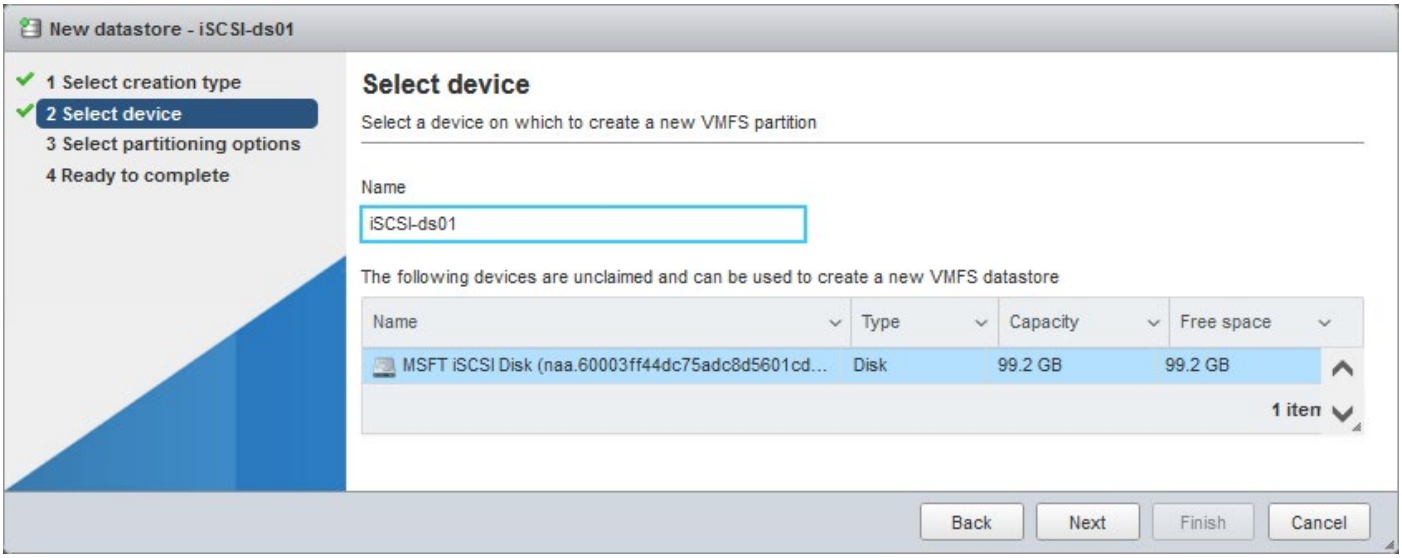


Figure 4.24

3. Select partitioning options. You can use default settings and use the full disk and the newest VMFS file system (see *Figure 4.25*).

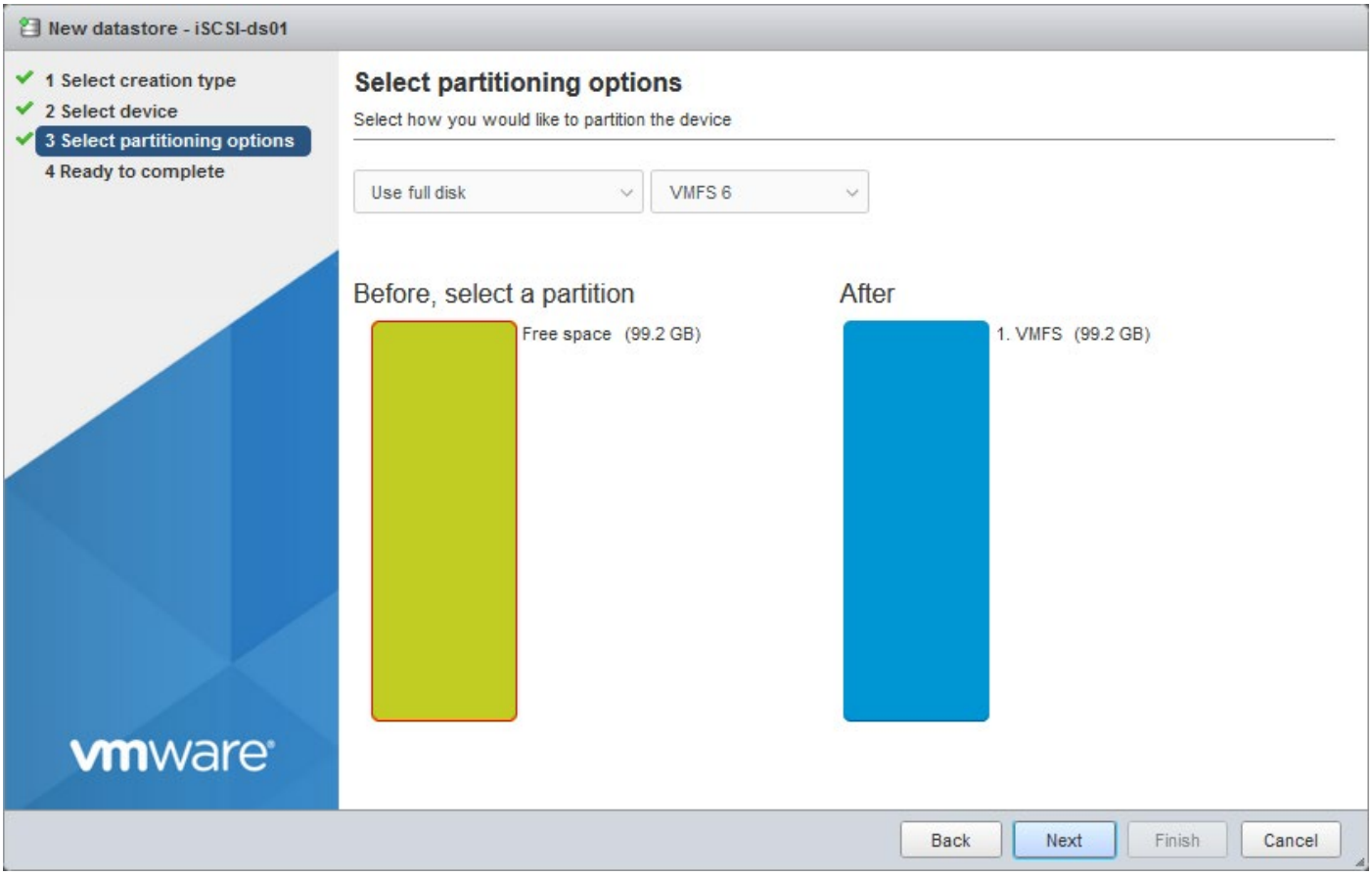


Figure 4.25

4 Ready to complete. Review your new datastore configuration (see *Figure 4.26*). If everything is correct, hit **Finish**.

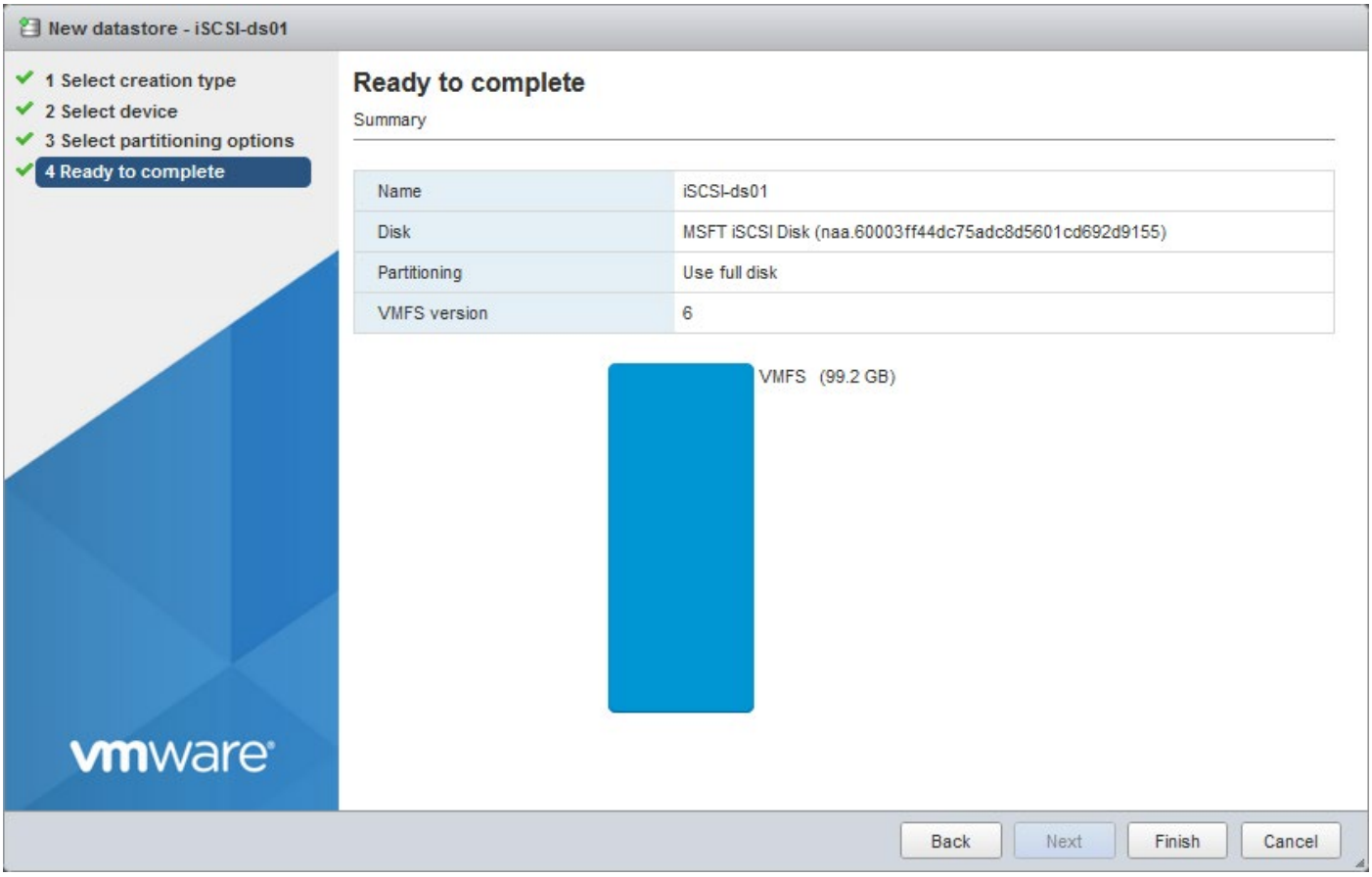


Figure 4.26

The warning message notifies you that all data on a disk will be erased (see *Figure 4.27*). If you agree, hit **Yes** to proceed.

Figure 4.27

A new VMFS datastore on iSCSI storage has been created. You can see the iSCSI-based VMFS datastore in **Storage > Datastores** in the web interface of VMware Host Client (see *Figure 4.28*).

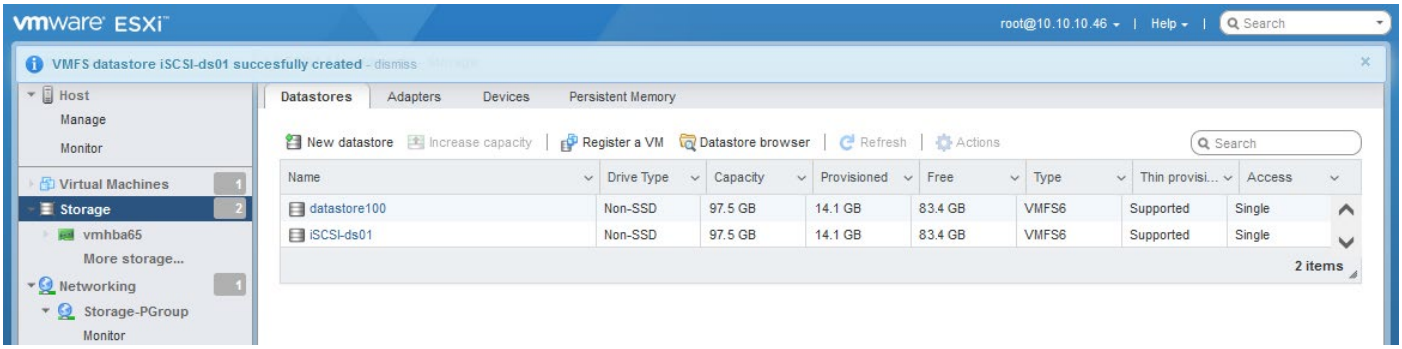


Figure 4.28

We have configured shared storage for the first ESXi host to be used in a vSphere cluster. Configure the second host (and other hosts) in the same way but don't create a new VMFS datastore because we have already created this file system. You can use VMware Host Client or VMware vSphere Client for this purpose.

Rescan storage adapters after adding the iSCSI target to the second ESXi host.

Adding a shared datastore on other ESXi hosts

The elegant solution is mounting a datastore to additional hosts by using vCenter.

1. Open VMware vSphere Client and go to **Datastores** (see *Figure 4.29*).
2. Select the needed datastore in the *Navigation* pane.
3. Right-click a shared datastore and hit **Mount Datastore to Additional Hosts** in the context menu.

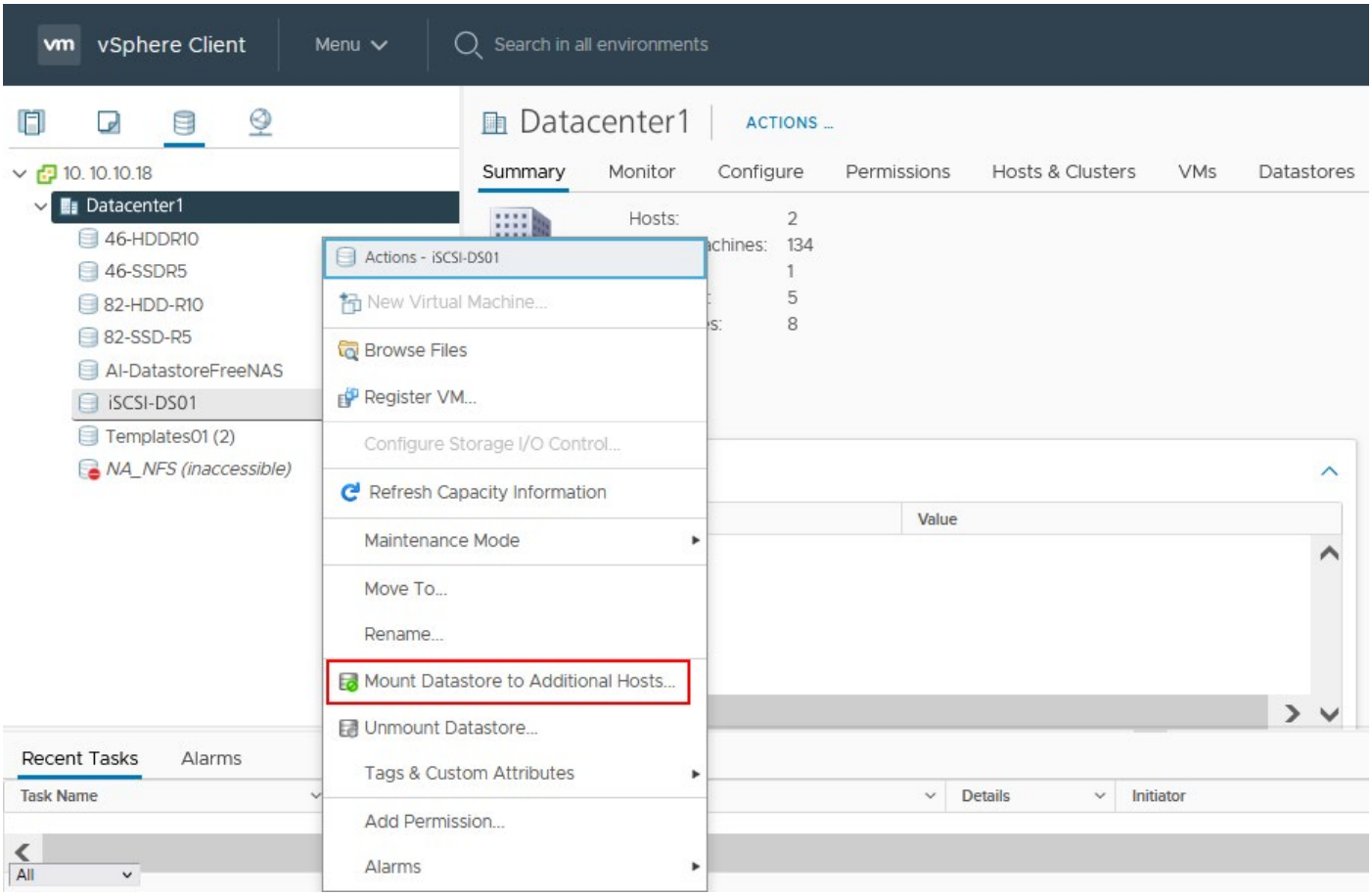


Figure 4.29

4. Select an ESXi host which you want to use this shared datastore from the list in the window that opens (see *Figure 4.30*). Then click **OK** to save settings.

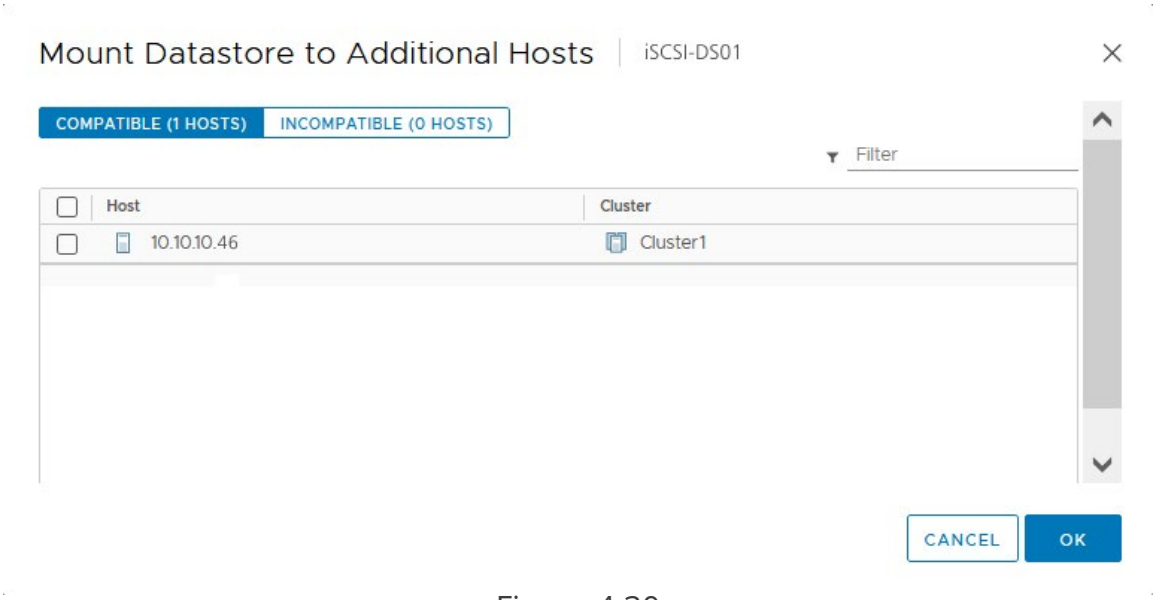


Figure 4.30

Configuration in vCenter

Use the same logic to configure an ESXi host to use shared iSCSI storage in VMware vSphere Client.

1. Select the needed ESXi hosts in the left (navigation) pane after selecting **Hosts and Clusters** (the first icon in the navigation pane).
2. To edit the configuration of virtual switches, go to the **Configure** tab and click **Virtual switches** in the **Networking** category (see *Figure 4.31*).

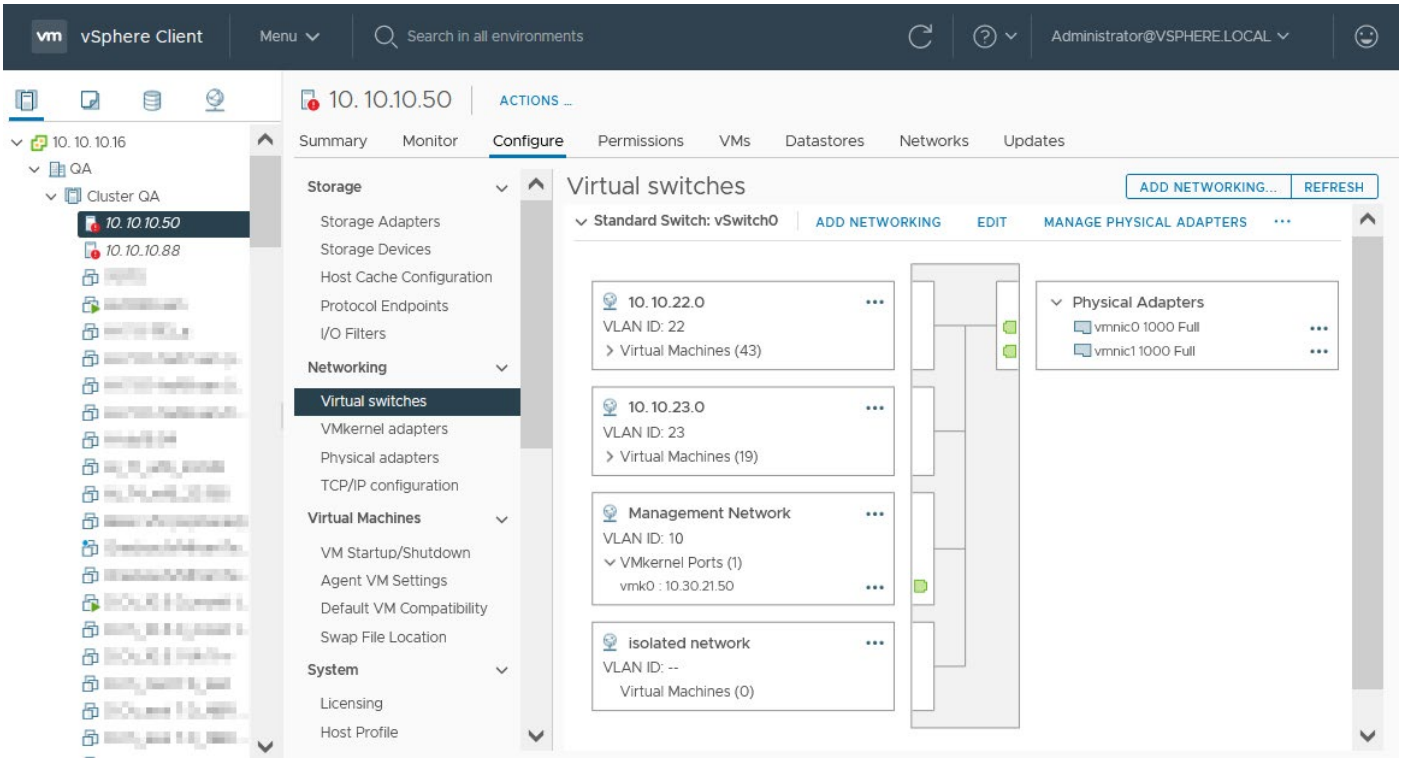


Figure 4.31

3. Go to **Configure > VMkernel adapters** to configure VMkernel adapters (see *Figure 4.32*).

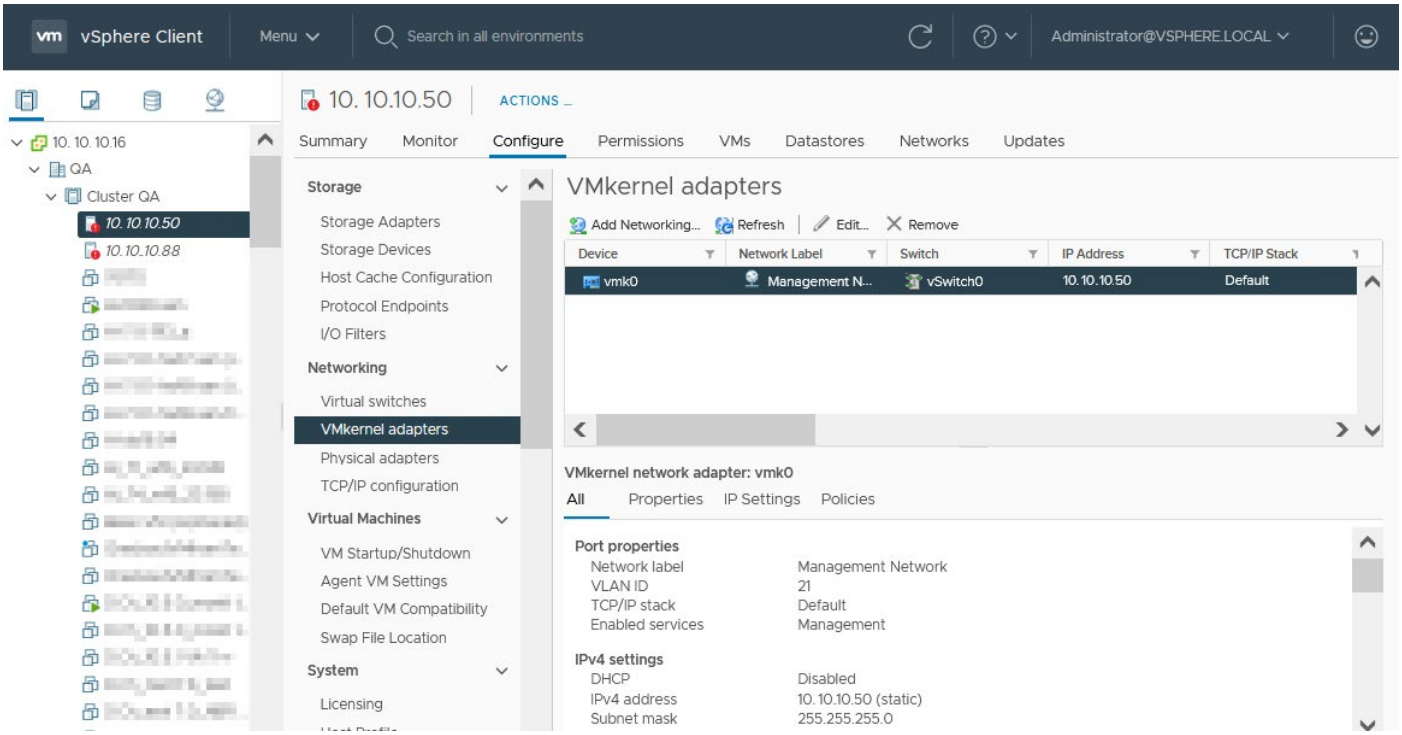


Figure 4.32

Storage adapter settings are located in **Configure > Storage** adapters (see *Figure 4.33*).

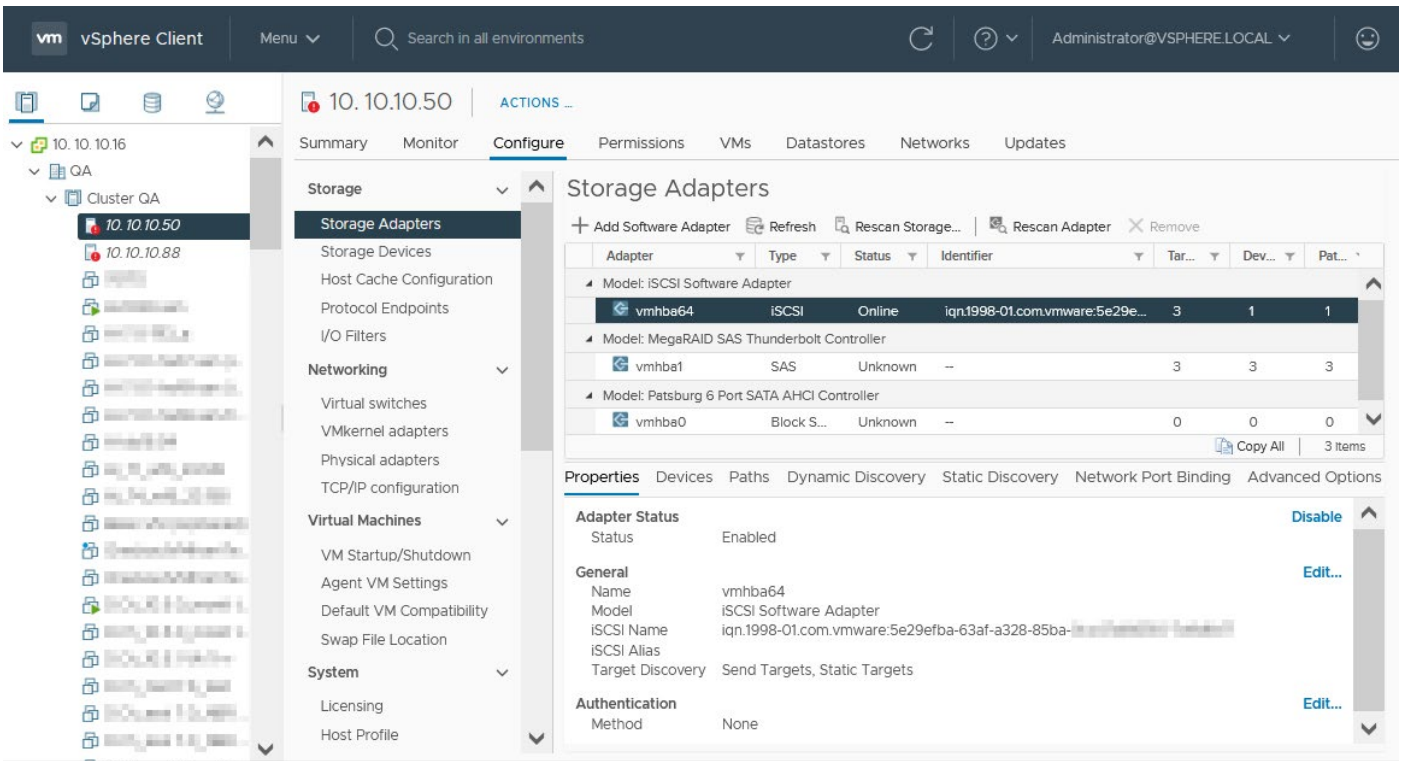


Figure 4.33

4. To view, add or edit storage devices configuration, go to the **Configure** tab and click **Storage devices** in the **Storage** section (see *Figure 4.34*).

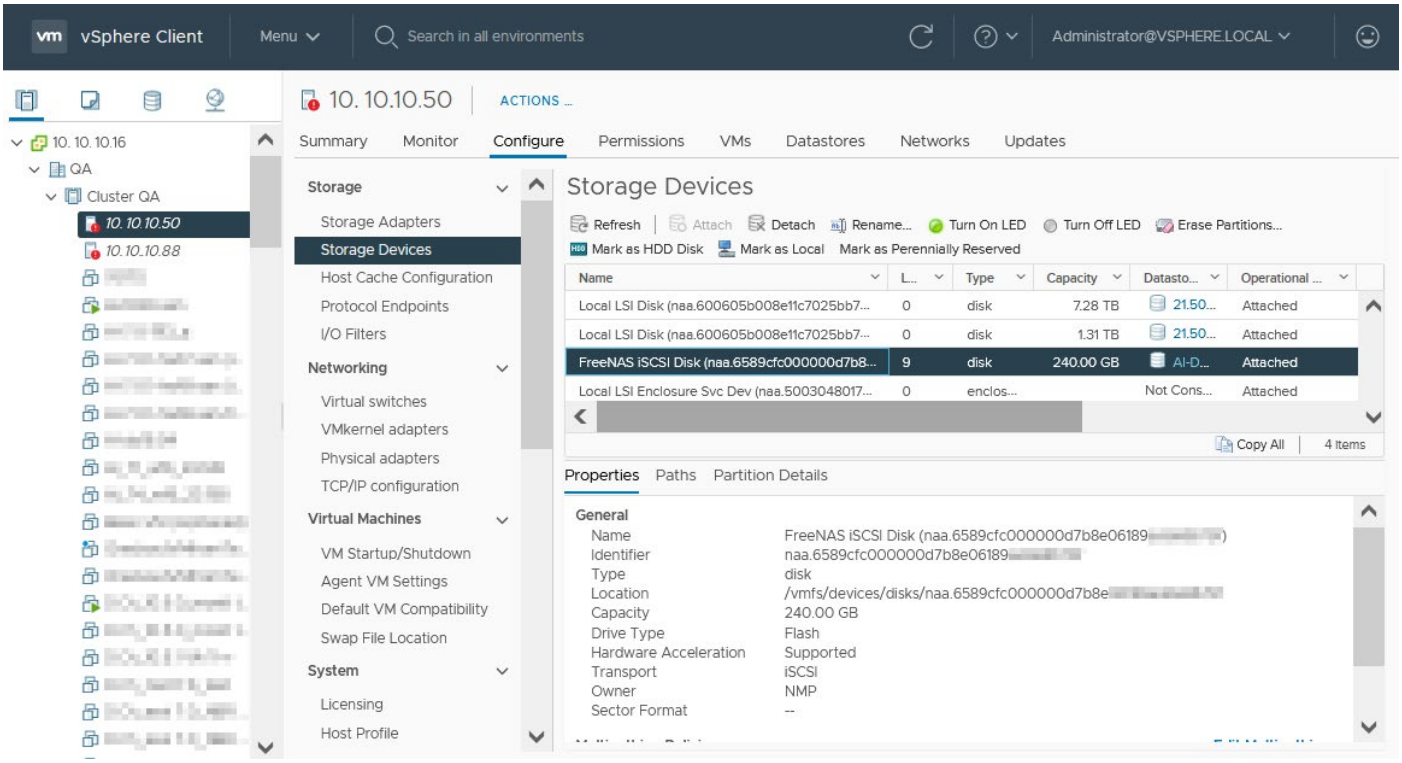


Figure 4.34

We use (configured) standard virtual switches on our ESXi hosts. You can configure a Distributed Virtual Switch in vCenter if you have a high number of ESXi hosts in the cluster for more convenience and save time.

Troubleshooting

Enable SSH connections on ESXi and use console commands to check the connectivity (availability) of the network storage from the selected VMkernel port:

- **vmkping -I vmk1 192.168.105.228**

Check whether the destination TCP port 3260 is available on the destination iSCSI network storage by using netcat:

- **nc -s 192.168.105.46 -z 192.168.105.228 3260**

Use this command to make sure that the software iSCSI is enabled on the ESXi host:

- **esxcli iscsi software get**

```
[root@ESXi7-1:~] vmkping -I vmk1 192.168.105.228
PING 192.168.105.228 (192.168.105.228): 56 data bytes
64 bytes from 192.168.105.228: icmp_seq=0 ttl=128 time=0.594 ms
64 bytes from 192.168.105.228: icmp_seq=1 ttl=128 time=3.292 ms

--- 192.168.105.228 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 0.594/1.943/3.292 ms

[root@ESXi7-1:~] nc -s 192.168.105.46 -z 192.168.105.228 3260
Connection to 192.168.105.228 3260 port [tcp/*] succeeded!
[root@ESXi7-1:~] esxcli iscsi software get
true
[root@ESXi7-1:~]
```

Figure 4.35

Heartbeat network

By default, a management network works as a heartbeat network. You can configure a dedicated heartbeat network for better reliability of the cluster.

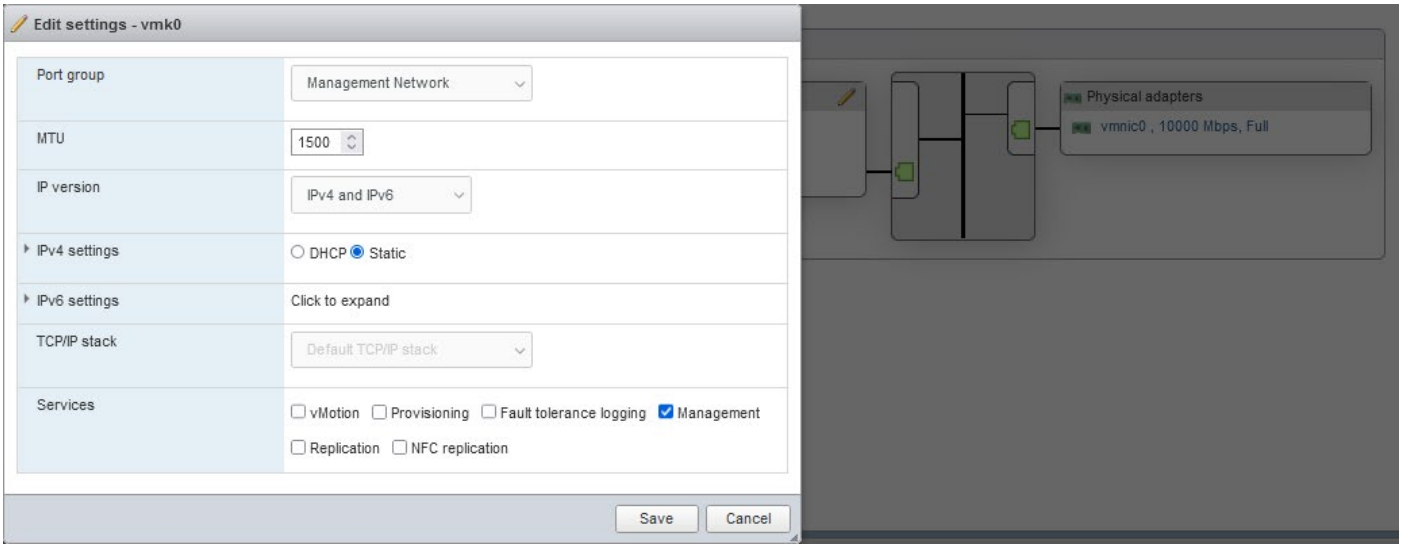


Figure 4.36

If you want to do this, create a new network. Create a vSwitch and a VMkernel adapter connected to the vSwitch. Select the **Management** checkbox for the appropriate VMkernel adapter to use this management network as a heartbeat network (see *Figure 4.36*).

Fault Tolerance logging network

Configure a fault tolerance logging network to use VMware High Availability with Fault Tolerance. Fault Tolerance is a feature that allows you to provide VM failover without any downtime.

With Fault Tolerance, a secondary ghost VM runs on the second ESXi host and completely replicates the VM state of the primary active VM. Inputs are replicated from the primary

to the secondary VM, but outputs are available only from the primary (active) VM. When a primary VM fails, the secondary VM becomes active and outputs are allowed.

The primary and secondary VMs are synchronized by using the Record/Replay technology. The main idea behind this technology is recording computer execution on one VM and saving this data to a log file. Then you can replay the recorded information on another VM. As a result, you have a VM copy that is identical to the original VM. Note that processors with Intel VT and AMD-V virtualization features must support this functionality to use Fault Tolerance.

When you have a VM with Fault Tolerance enabled, both the primary and secondary VMs use shared storage. CPU, memory, and virtual device inputs are transferred via the network from the primary VM (record) to the secondary VM (replay). You need to use the VMware FT logging network with enough bandwidth for this purpose to ensure that all required data is transferred in time to replicate the VM state. In this case, the secondary VM is ready to take over if an ESXi host with the primary VM fails.

VMware FT logging network is used to copy all input information from the primary VM to the secondary VM within a cluster to provide Fault Tolerance. For this reason, you should configure the FT logging network and connect each cluster node to this network. Let's look at how to configure FT logging on an ESXi host in vCenter by using VMware vSphere Client.

1. Select an ESXi host in the navigation pane of VMware vSphere Client (use the *Hosts and Clusters* view).
2. Select the **Configure** tab and in the **Networking** section, click **VMkernel** adapters.
3. Click **Add Networking**.
4. The *Add Networking* wizard opens (see *Figure 4.37*).

   1. Select connection type. Select **VMkernel Network Adapter**. Click **Next** at each step of the wizard to continue.

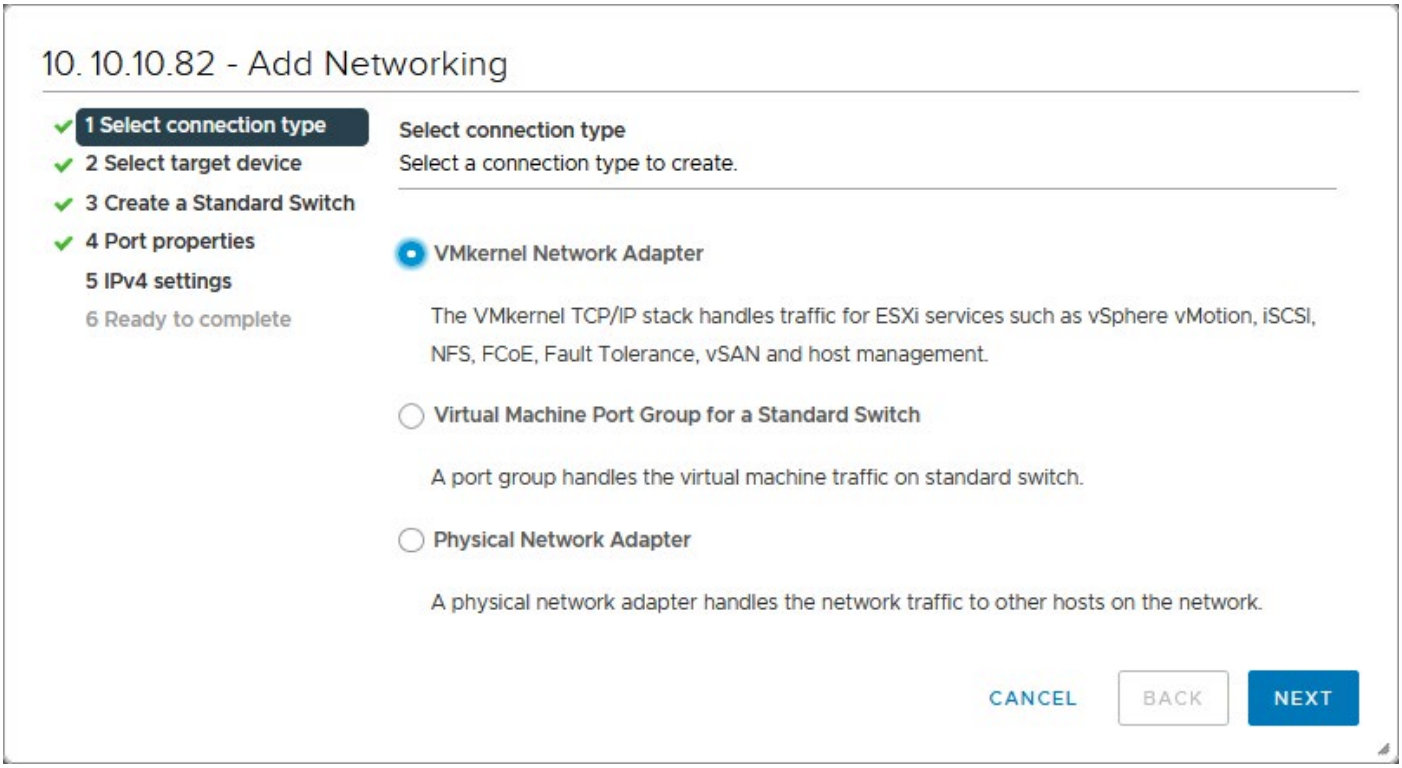


Figure 4.37

2. Select target device. You can select an existing network, select an existing standard switch or create a new standard virtual switch for connecting a VMkernel adapter to this switch (see *Figure 4.38*). In this example, we select the third option – **New standard switch**. Set the maximum transmission unit (MTU) in bytes.

If you use Jumbo frames, set 9000. Otherwise, leave the default value of 1500 bytes.

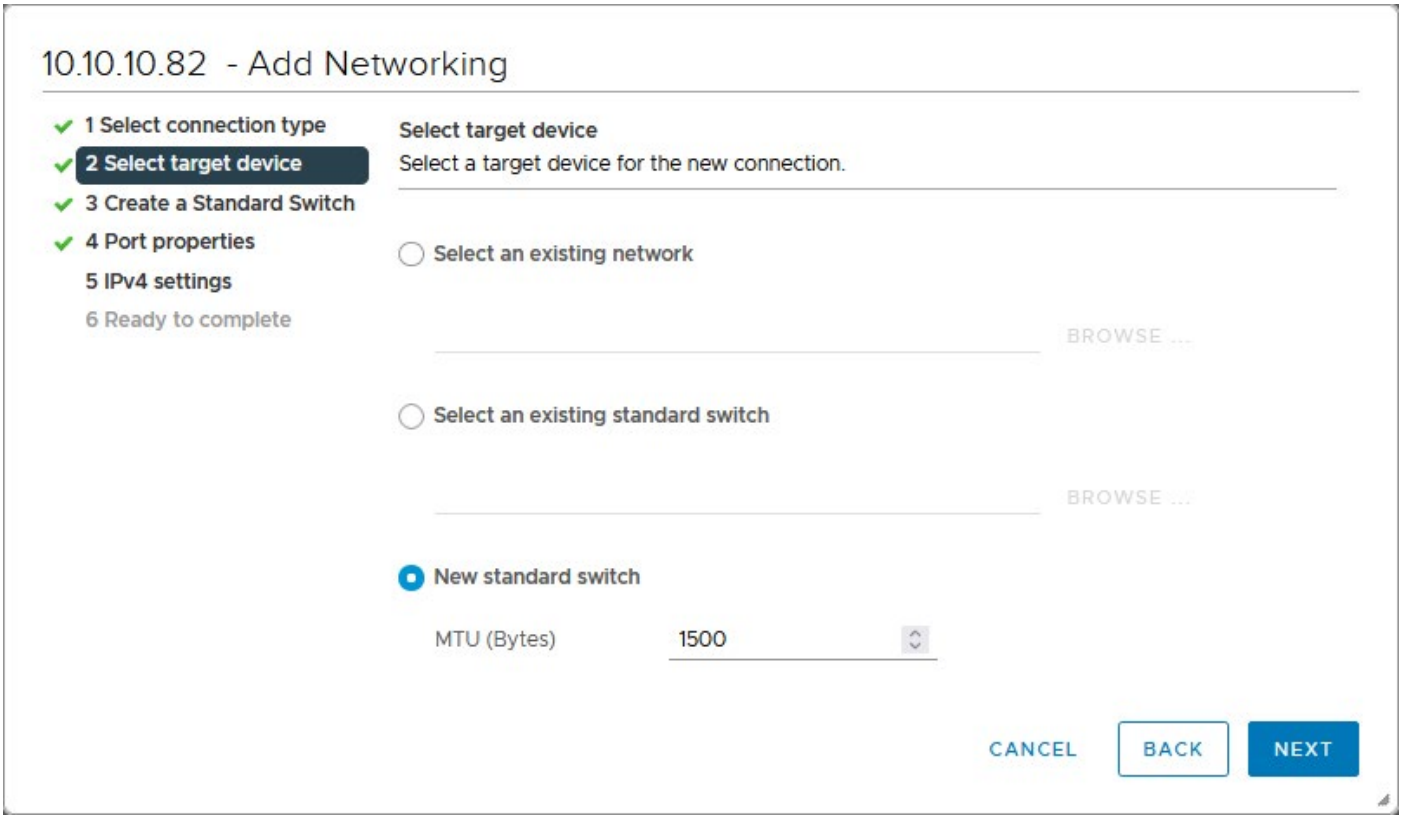


Figure 4.38

3. Create a Standard Switch. Assign physical network adapters to this virtual switch (see *Figure 4.39*).

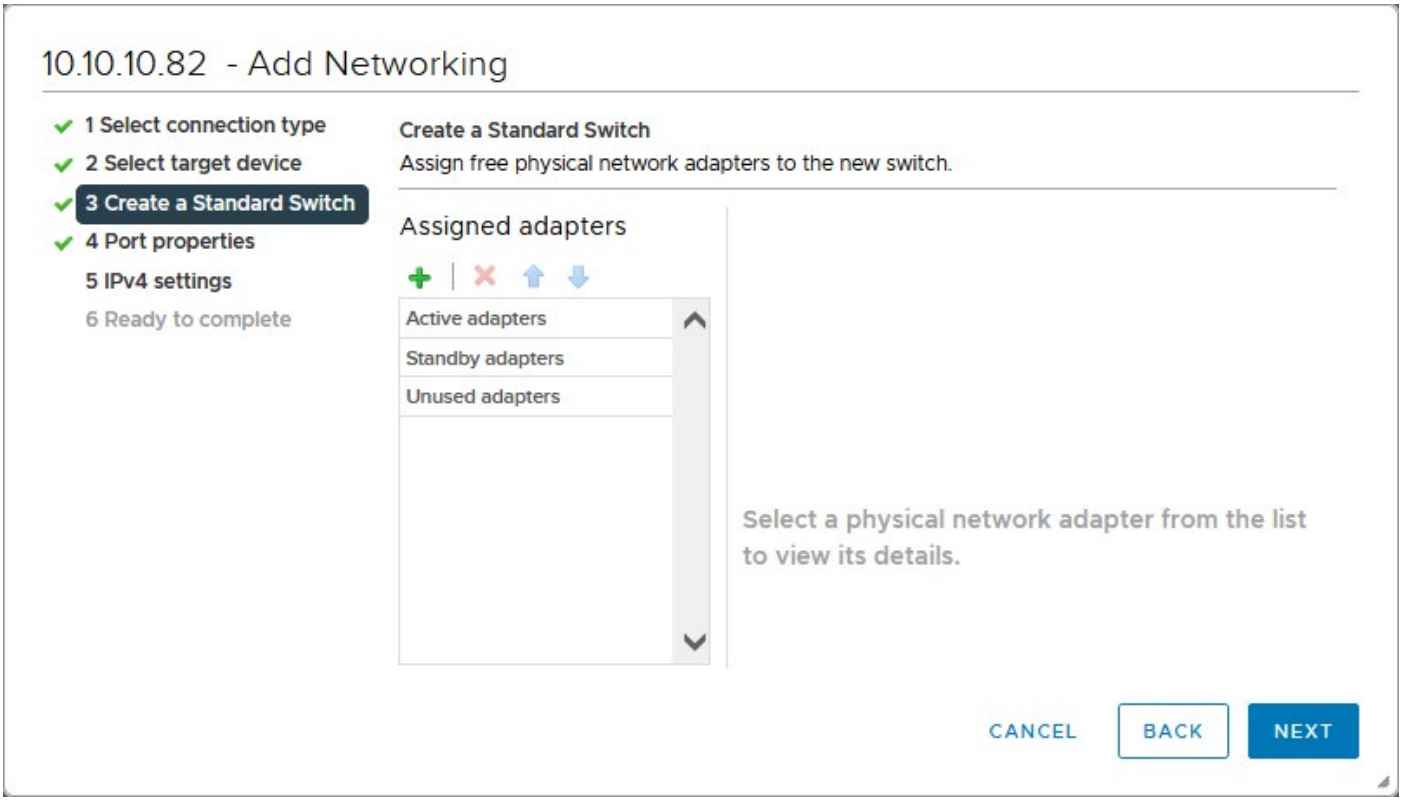


Figure 4.39

4. Port properties. Specify VMkernel port settings (see *Figure 4.40*).

Network label: *FT logging*

VLAN ID: *(optional)*

IP settings: *IPv4*

MTU: *Get MTU from switch*

TCP/IP stack: *Default*

Enabled Services: **Fault Tolerance logging** (select this checkbox)

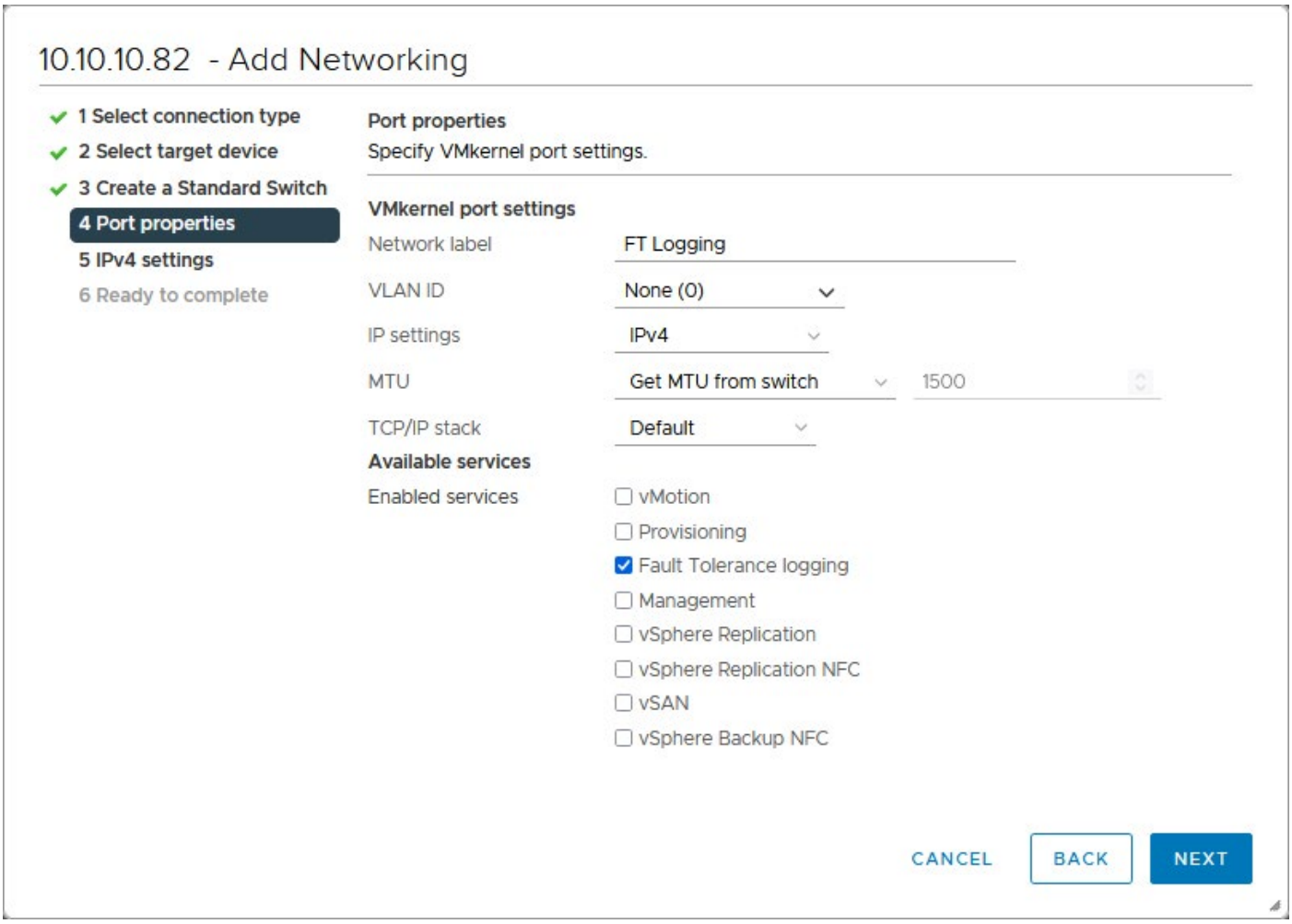


Figure 4.40

5. IPv4 settings. Specify VMkernel IPv4 settings (see *Figure 4.41*). It is recommended that you use static IPv4 settings and set the appropriate values for your FT logging network.

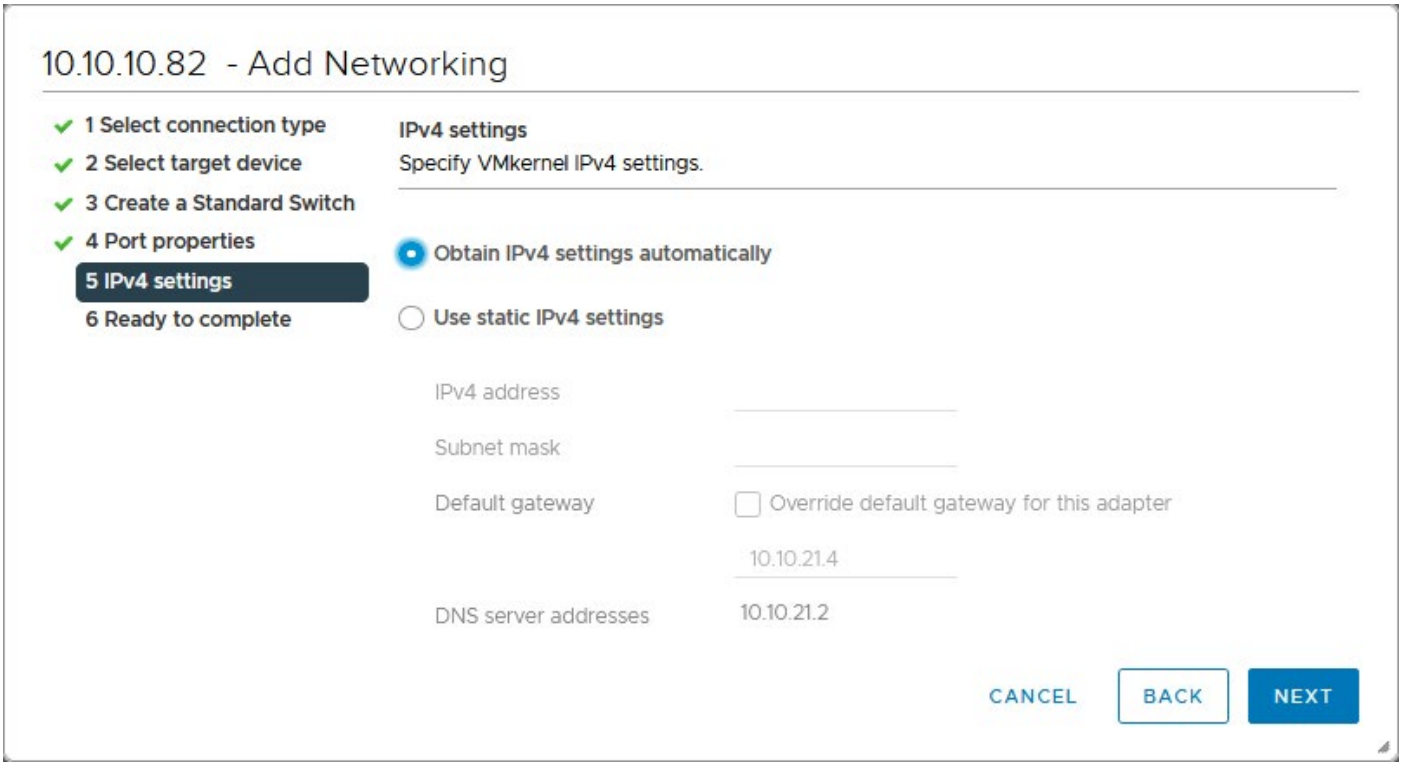


Figure 4.41

6. **Ready to complete.** Check your configuration and if everything is correct, click Finish.

How to Create and Configure a Cluster

Now, once we have configured ESXi hosts, vCenter Server, networking, and shared storage, we can create a new cluster and add ESXi hosts to the cluster. Let's look at this process step by step.

Creating a New Cluster

Do the following steps to create a new cluster in VMware vSphere:

1. Right-click the datacenter and hit **New Cluster** in the menu that opens (see *Figure 5.1*).

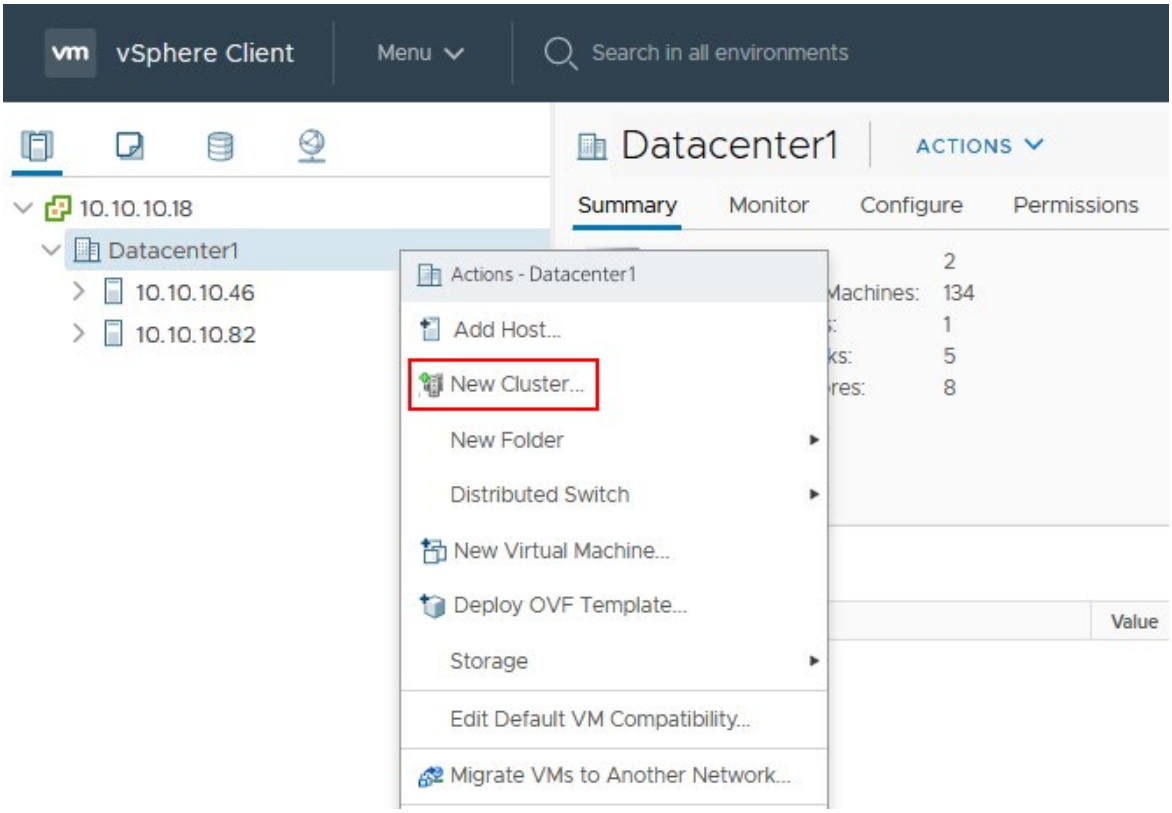


Figure 5.1

2. Set the name of your cluster, for example, *Cluster1*. Leave the sliders near the DRS and vSphere HA options unchanged (you can switch them on later) (see *Figure 5.2*). Hit **OK** to create the cluster.

Note
You can create VMware vSAN hyper-converged storage and use vSAN to store VMs for the High Availability cluster. You can use VMware vSAN, HA and DRS in the same cluster. You can use VMware vSAN datastores as traditional datastores and configure shared datastores. We don't use VMware vSAN in this tutorial.

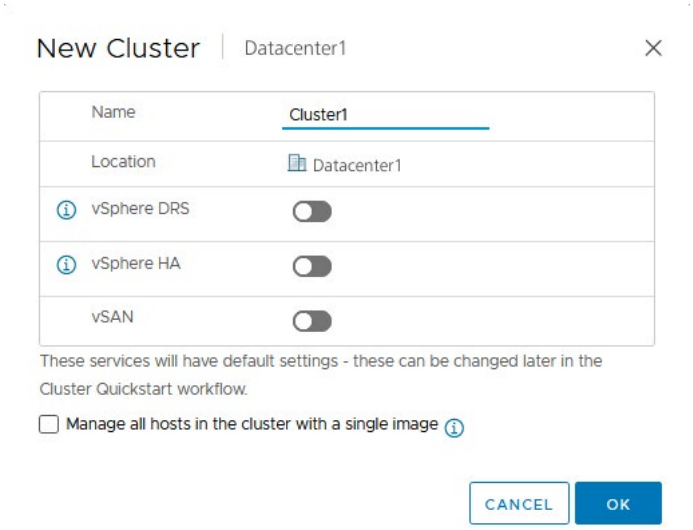


Figure 5.2

You can unify the software configuration of all ESXi hosts within a cluster by using a single image. For this purpose, you can select this checkbox:

- Manage all hosts in the cluster with a single image

Then you can select the needed ESXi version and vendor addon (optional). (see *Figure 5.3*)

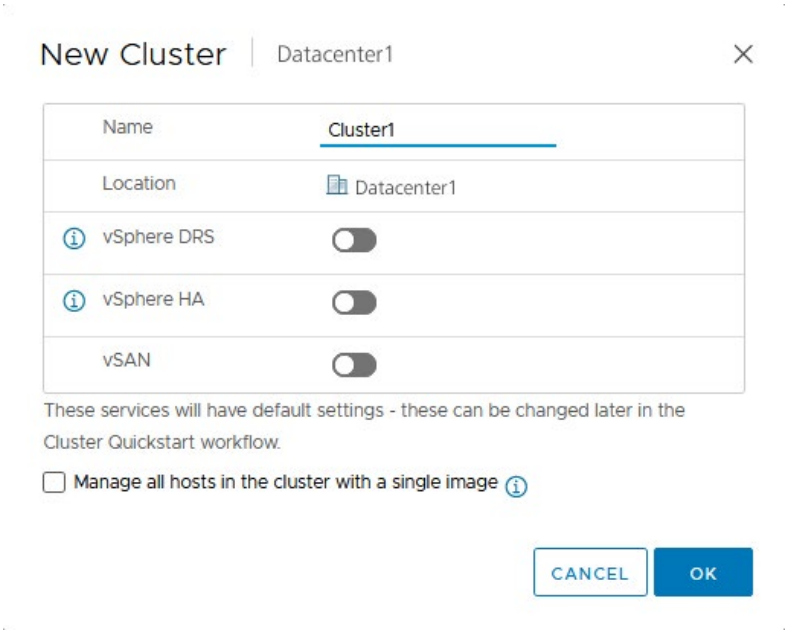


Figure 5.3

VMware Lifecycle Manager deploys the reference ESXi image to all hosts within the cluster to have exactly the same software specifications on all cluster nodes.

We don't use a single image in this tutorial.

Adding Hosts to the Cluster

1. Add ESXi hosts to the cluster. Note that your ESXi hosts must belong to the same datacenter. Right-click the cluster name and click **Add Hosts...** (see *Figure 5.4*).

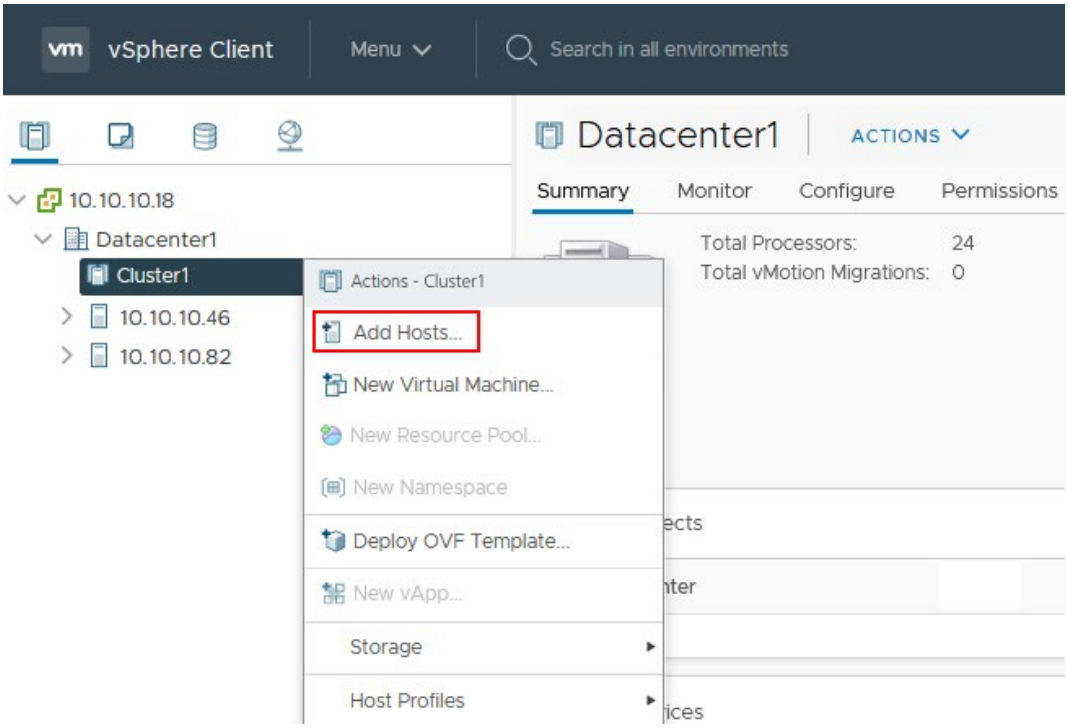


Figure 5.4

2. The *Add new and existing hosts* wizard opens (see *Figure 5.5*).

   1. Add Hosts.

   Enter the name or IP address of the ESXi host you want to add to your cluster (10.10.10.46 in our case).

   Enter a user name and password of the administrative account of the ESXi host you are adding to the cluster. The **root** user is used by default.

   Click **Add Host** to enter one more host.

   Click **Next** at each step of the wizard to continue.

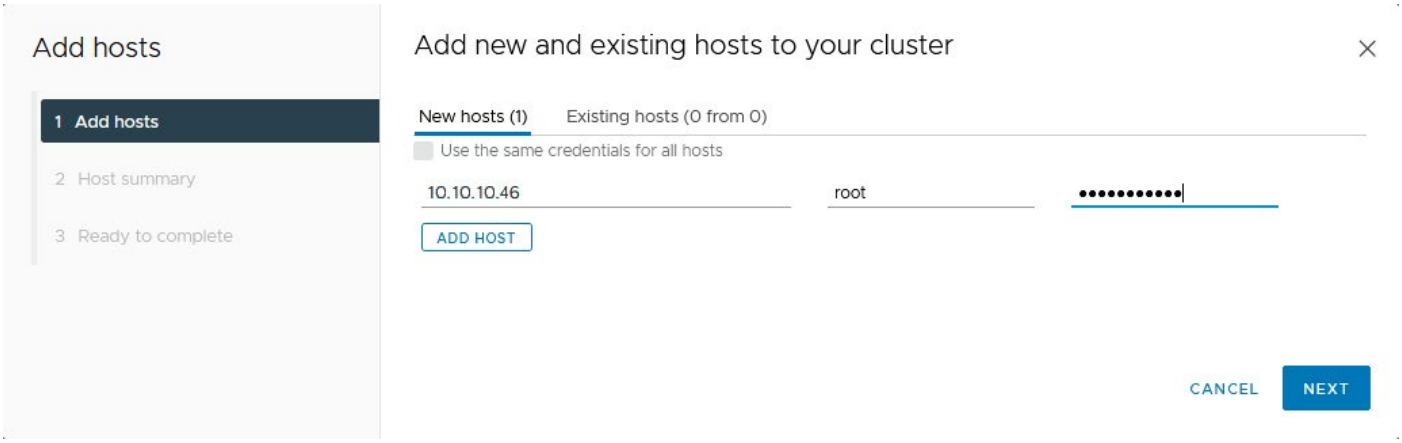


Figure 5.5

You can ignore the certificate warning and click **OK** when the security alert is displayed (see *Figure 5.6*).

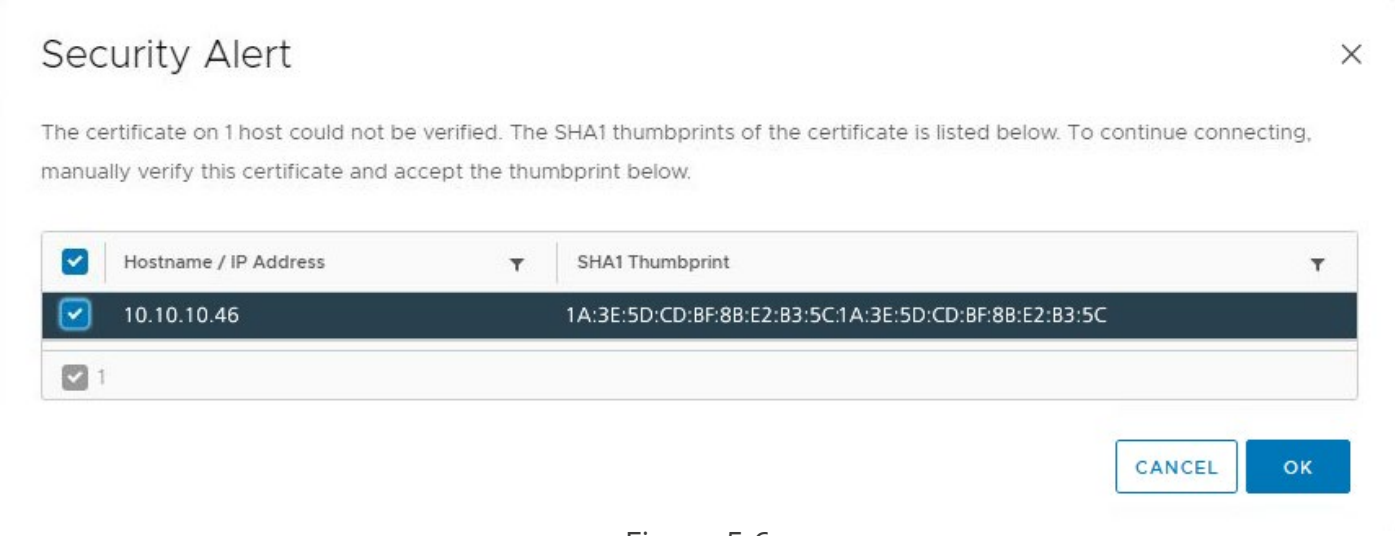


Figure 5.6

2. Host summary. Check the host summary and information about the added hosts.

3. Ready to complete. Now you are ready to finish adding hosts to your vSphere cluster (see *Figure 5.7*). Click **Finish**.

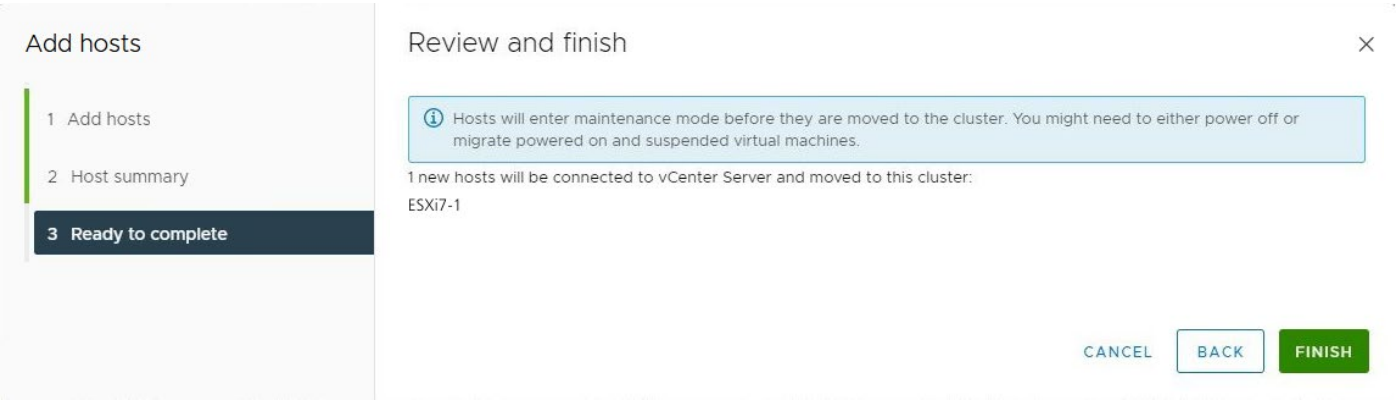


Figure 5.7

A cluster is created and ESXi hosts are added to the cluster. Now you can enable and configure High Availability and Distributed Resource Scheduler.

How to Create a DRS Cluster

In order to create a Distributed Resource Scheduler (DRS) cluster, open VMware vSphere Client and do the following actions as explained below.

1. Go to **Hosts and Clusters** and click the cluster name in the *Navigator* pane.
2. Once you have selected the cluster, click the **Configure** tab (see *Figure 6.1*). You should see **vSphere DRS** and **vSphere Availability** options in the Services section of the middle pane.
3. Select **vSphere DRS** and click **EDIT**.

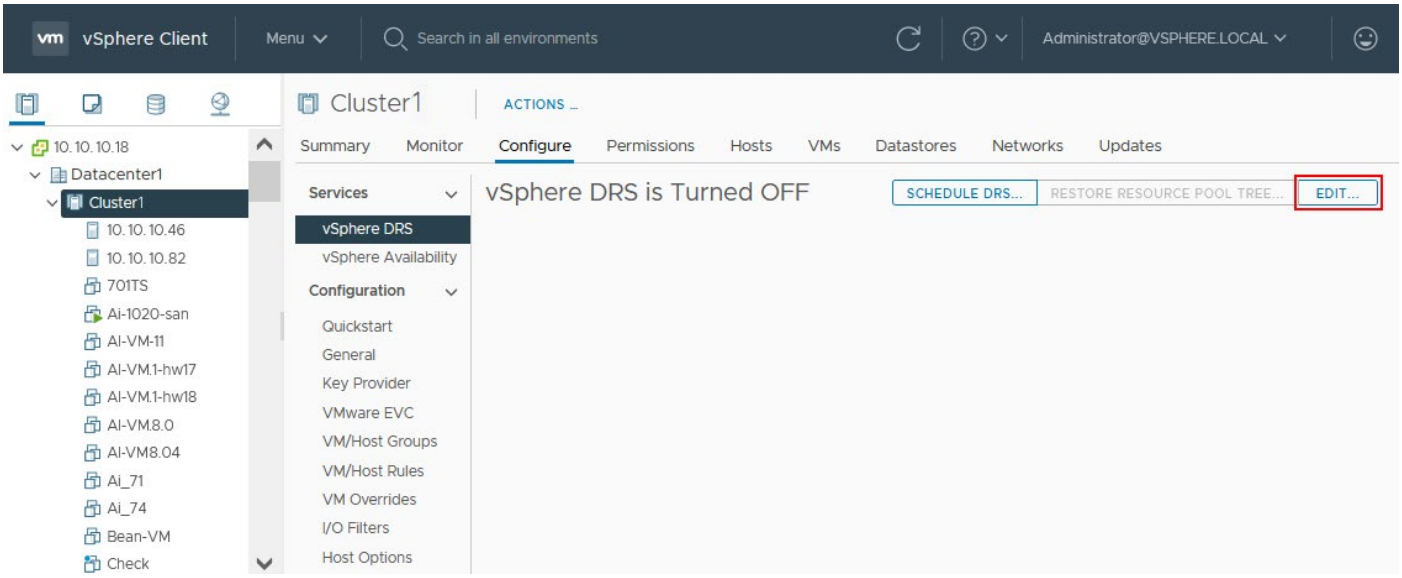


Figure 6.1

4. Click the **vSphere DRS** switcher to enable DRS in the cluster (see Figure 6.2).
5. Once you have enabled vSphere DRS, you should see four tabs:
   - Automation
   - Additional Options
   - Power Management
   - Advanced options

Each tab has a set of options you can configure. Let's explain all available options in detail. You can select and configure the options you need to use in your environment.

There are useful tips displayed in the vSphere client interface that help you understand what settings are the best to meet your needs. Click the **(i)** icon at the appropriate option to see the tip.

Automation

There are multiple settings in the automation tab:

- **Automation level** is the option that allows setting the custom automation level for a virtual machine.
  - **Fully automated.** VMware DRS applies initial VM placement recommendations and load balancing recommendations (migrating VMs to other ESXi hosts with enough resources) automatically.
  - **Partially automated.** Recommendations are applied automatically only for initial VM placement (for example, when you create a new VM).
  - **Manual.** VMware DRS provides VM initial placement recommendations and load balancing recommendations and you should apply them manually in VMware vSphere Client.
- **Migration Threshold** controls how conservative or aggressive DRS runs. You can set values from 1 to 5. Migration Threshold 1 is used for cluster Maintenance mode. Recommendations are not generated due to cluster imbalance or VM demand. Migration Threshold 2 is the most conservative and Migration Threshold 5 is the most aggressive one. The higher the Migration Threshold, the more migrations are allowed, even if this can result in only the slightest benefit to performance or cluster balance. Change the migration threshold value by moving the slider.
- **Predictive DRS** can be used in combination with vRealize Operations Manager. Predictive DRS allows a cluster to predict resource demand for VMs and migrate VMs to ESXi hosts rationally before resource consumption spikes occur. This feature helps prevent resource contention and performance degradation for VMs.
- **Virtual Machine Automation.** Swtich on the toggle to allow setting a custom DRS automation level for individual virtual machines. DRS automation settings for an individual VM override the general DRS automation settings of a cluster in this case.

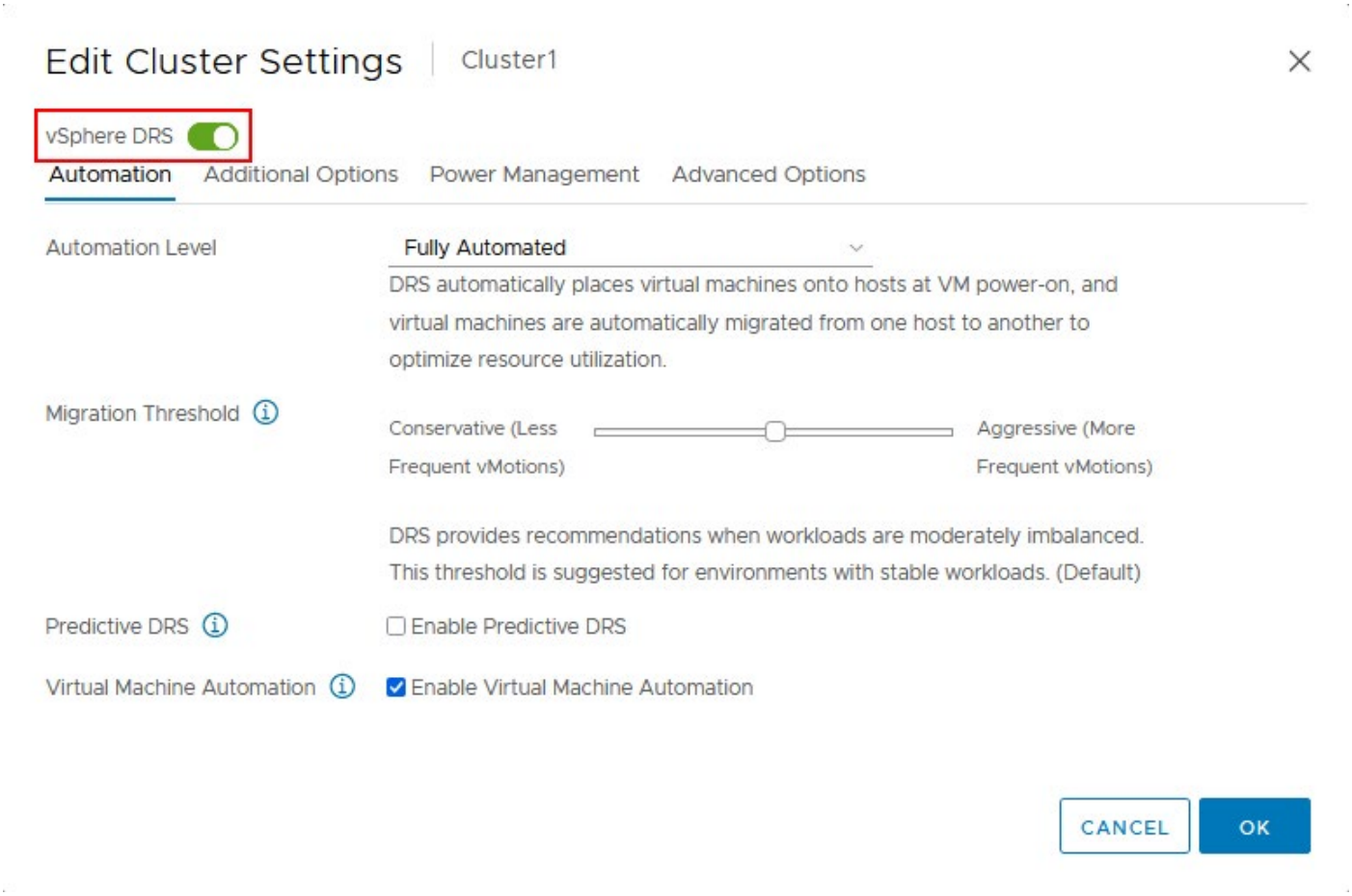


Figure 6.2

Additional Options

There are three additional options that you can configure for DRS:

- **VM Distribution** to distribute virtual machines across ESXi hosts in the cluster for better availability.
- **CPU Over-Commitment** limits CPU over-commitment for all hosts in the cluster. The format is vCPU:pCPU and is set as a custom value X:1.
- **Scalable Shares** optimizes resource scheduling for VMs when using resource pools with different priorities (High, Normal, Low). VM workloads for the higher priority CPU shares can now receive a higher entitlement. It is recommended that you enable this option.

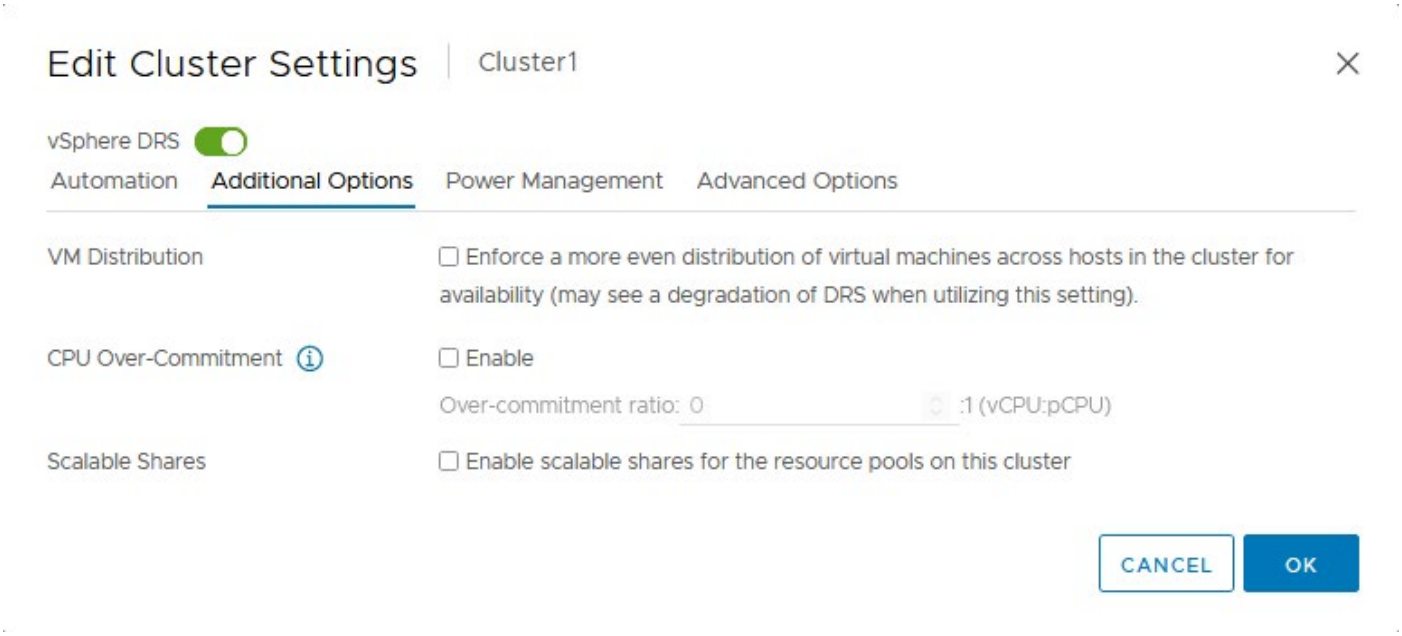


Figure 6.3

Power Management

The power management tab contains options for power saving (see Figure 6.4). Enable Distributed Power Management (DPM) only if you want to enable energy-saving, migrate VMs to ESXi hosts with enough resources and power off unused ESXi hosts. Otherwise, DPM must be disabled. If DPM determines that an ESXi host within a cluster is not needed, this host is shut down.

- **Automation Level**

  There are three options:

  - Off – the DPM feature is disabled.
  - Manual – DPM provides VM migration recommendations and ESXi host shut down recommendations, but you need to apply or reject them manually.
  - Automatic – if all VM migrations can be run, then the recommended ESXi hosts are powered off automatically by DPM (the VM migration and hosts shut down recommendations are applied automatically).

- **DPM Threshold.** There are five priorities for DPM recommendations that are based on the amount of over-utilization and under-utilization detected in the DRS cluster. Recommendations are changed from 1, which is mandatory, to 5, which provides slight improvement. Use the slider to configure the DPM threshold.

  The conservative side is used to generate only priority-1 recommendations, providing a significant power-saving effect and less frequent VM migrations with vMotion. The aggressive mode (on the right side) initiates VM migrations to power off an ESXi host or host even if a small benefit can be achieved.

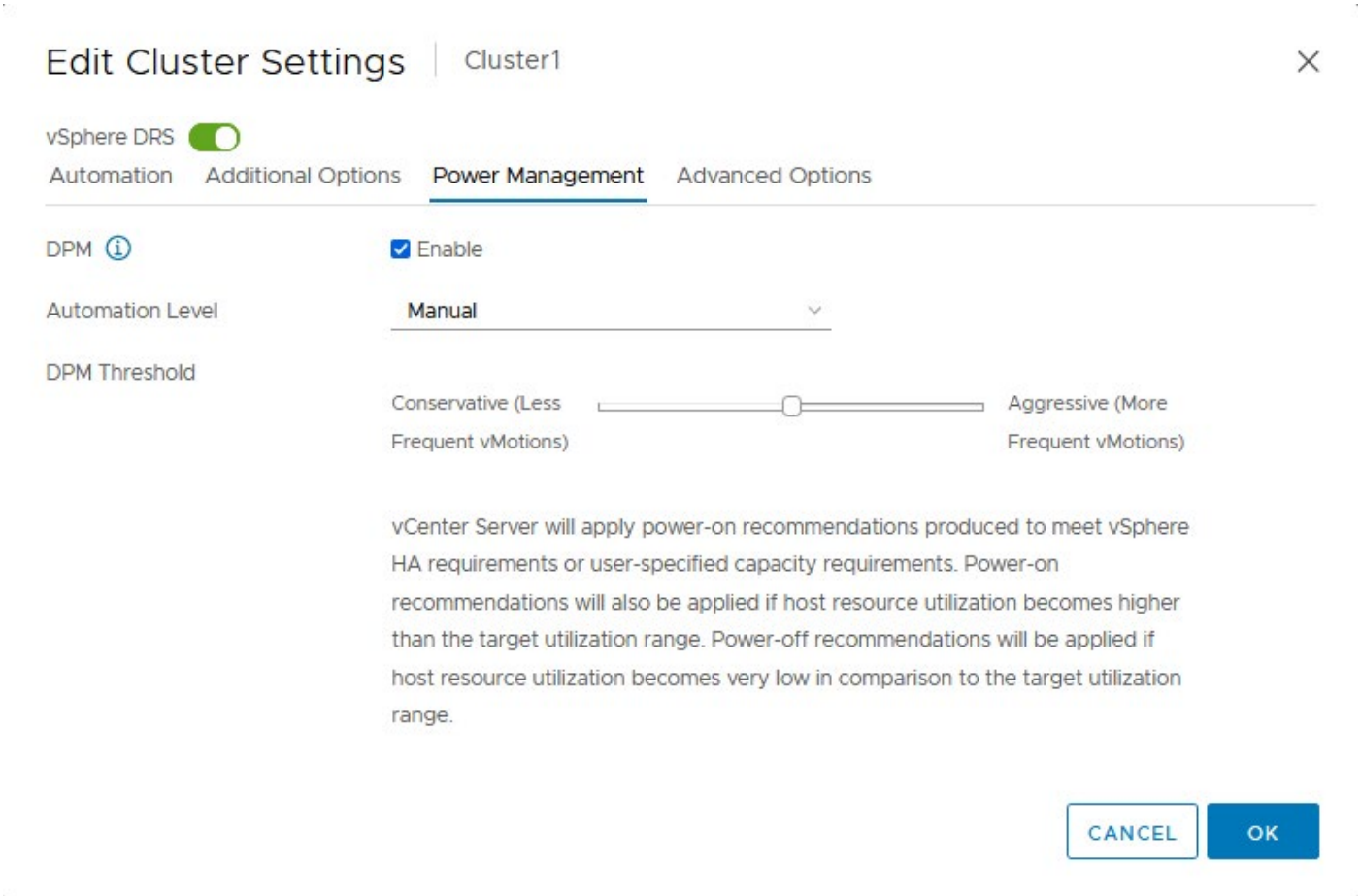


Figure 6.4

VMware Distributed Power Management (DPM) supports three power management protocols to bring a host out of standby mode:

- Intelligent Platform Management Interface (IPMI)
- Hewlett Packard Enterprise Integrated Lights-Out (iLO)
- Wake-on-LAN (WOL)

Each of these protocols requires separate hardware support and configuration. If a host does not support any of these protocols, this host cannot be put into standby mode by DPM. If a host supports multiple protocols, they are used in the following order: IPMI, iLO, WOL.

Advanced Options

Advanced options for DRS are used to fine-tune the cluster. You can also configure some options that are not available in the GUI by using advanced options (see *Figure 6.5*).

Examples of DRS parameters you can configure in advanced options:

- UseDownTime (1 to 0)
- MinImbalance (50 to 0)
- MaxMovesPerHost (Adaptive / 0)

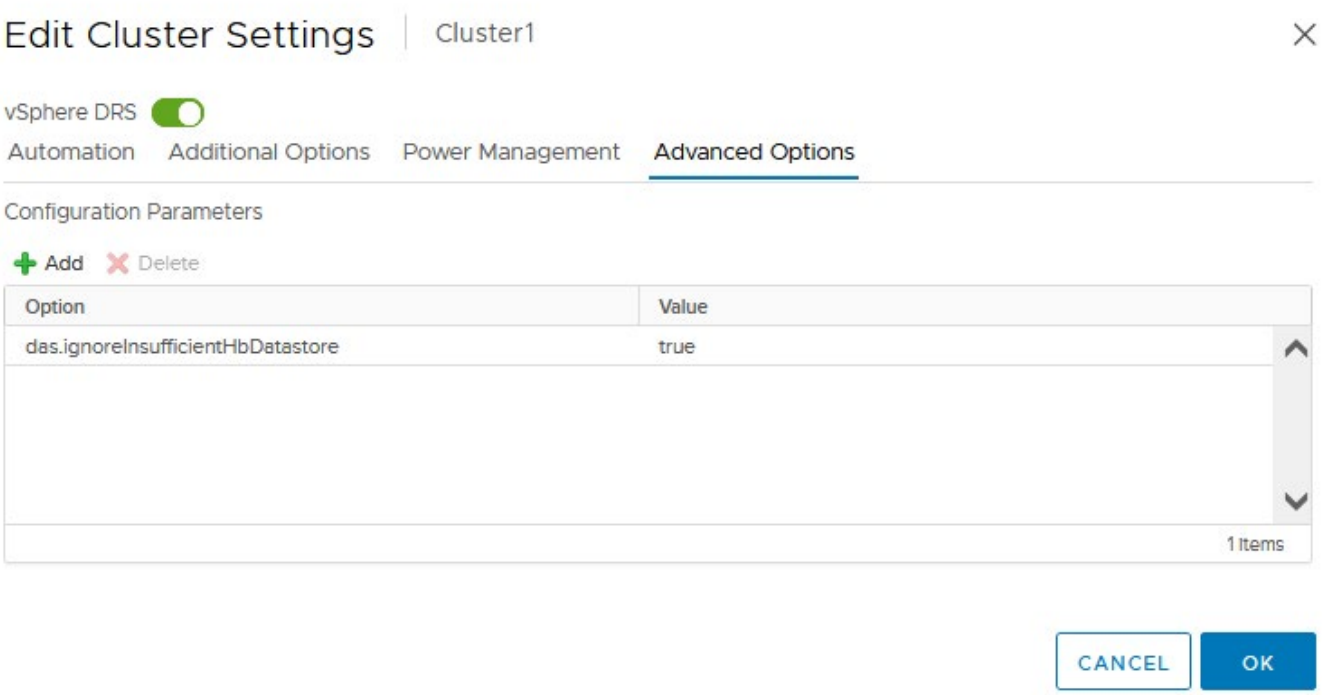


Figure 6.5

We have explained DRS options in all four tabs.

Click **OK** to save DRS settings.

Check the cluster status and DRS score by selecting a cluster in the navigation pane of VMware vSphere Client (use the Hosts and Clusters view). In our example, the cluster DRS score is 100 because many computing resources are free, and a small number of light VMs are running (see *Figure 6.6*).

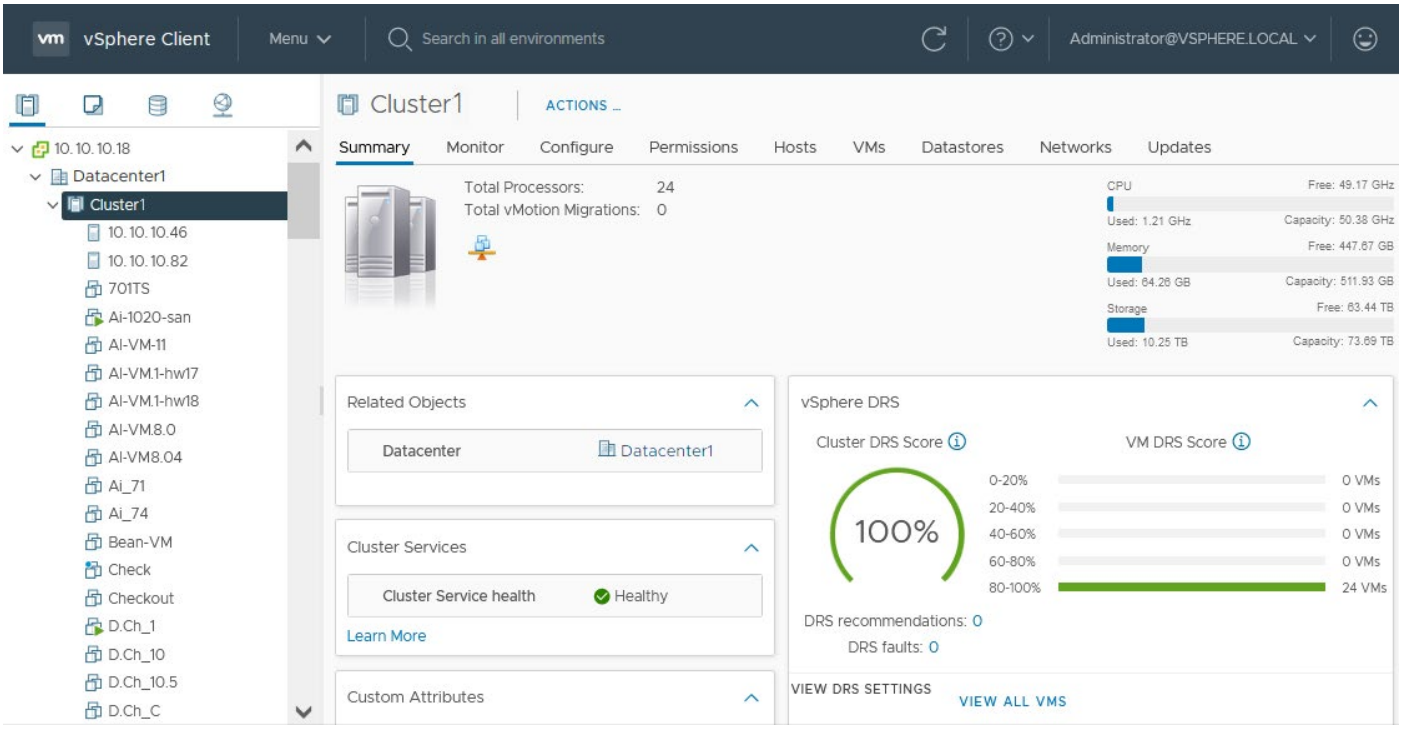


Figure 6.6

Affinity Rules

You can add Affinity Rules if necessary.

Affinity Rules allow you to control the placement of virtual machines that interact intensely with each other. For example, when you run Database Server, Web Server, and Application Server on different virtual machines, place them on one ESXi host. Residing on the same ESXi host reduces network load and can increase performance in this case.

Affinity rules also allow you to configure placing different VMs only on different ESXi hosts. Another use case of this feature is software licensing limitations when a VM must run on one host and VM migration is forbidden by the license agreement. Let's look at how to configure DRS affinity rules.

1. Select your cluster and go to the **Configure** tab.
2. Click **VM/Host Rules** in the **Configuration** section of the middle pane (see *Figure 6.7*).
3. Click **+Add** to add a new affinity rule.

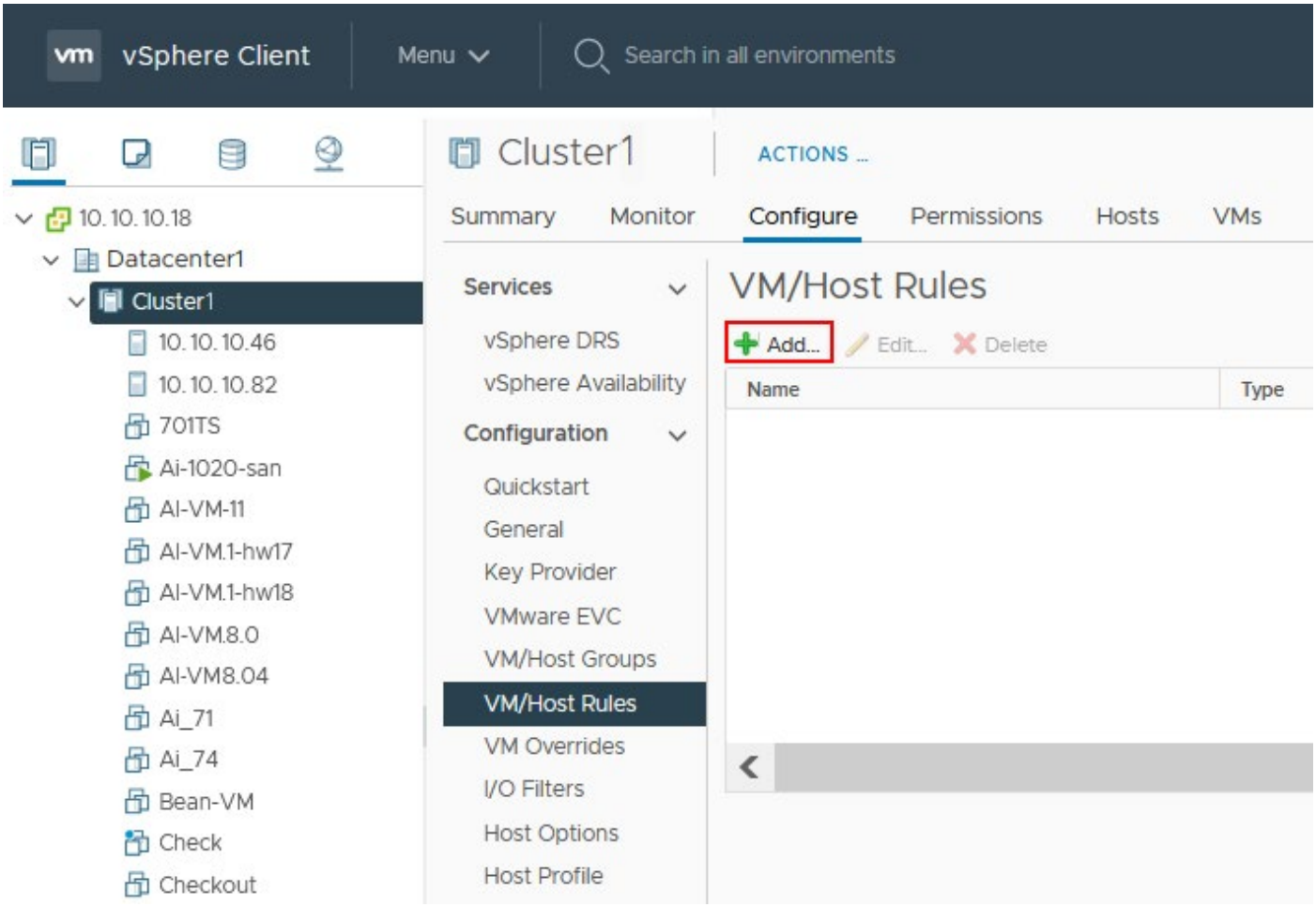


Figure 6.7

4. The VM/Host rule configuration window opens.

Enter the rule name, for example, *Affinity Rule 01*. Select the **Enable rule** checkbox to activate the rule

Available options:

- **Keep Virtual Machines Together** – allows you to keep two or more VMs on the same ESXi host
- **Separate Virtual Machines** – selected VMs must run only on different ESXi host
- **Virtual Machines to Hosts** – a specified VM or VMs must run only on the selected ESXi host or hosts
- **Virtual Machines to Virtual Machines** – is used when multiple VMs must run on the same ESXi host, for example, due to performance reasons, when one VM depends on another VM.

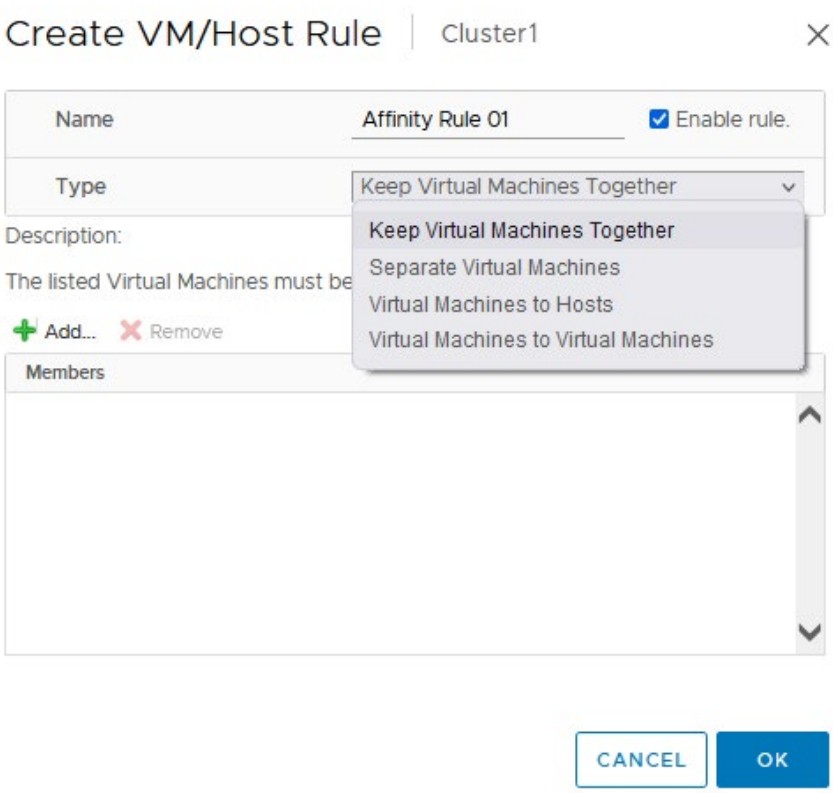


Figure 6.8

5. Once you have selected the needed affinity option (rule type), you should select the VMs that you want to store on the appropriate hosts according to the rule. We have selected to **Keep virtual machines together**.

   Click **+Add** to select VMs running within the cluster for this rule (see *Figure 6.8*).

   Once you have selected the needed VMs, click **OK** (see *Figure 6.9*).

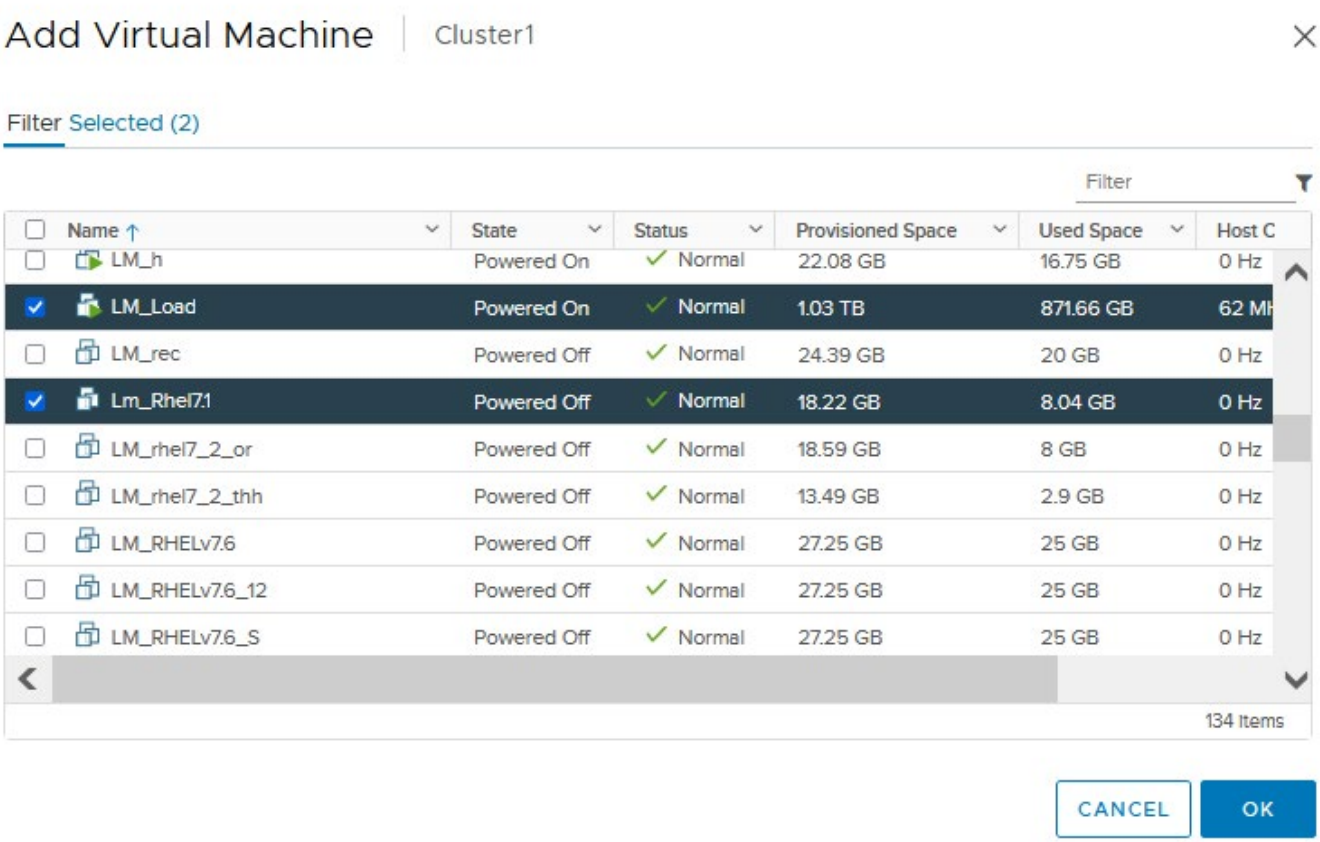


Figure 6.9

6. Now you can see the selected VMs for the *Keep Virtual Machines Together* affinity rule type (see *Figure 6.10*).

   Hit **OK** to save the affinity rule settings.

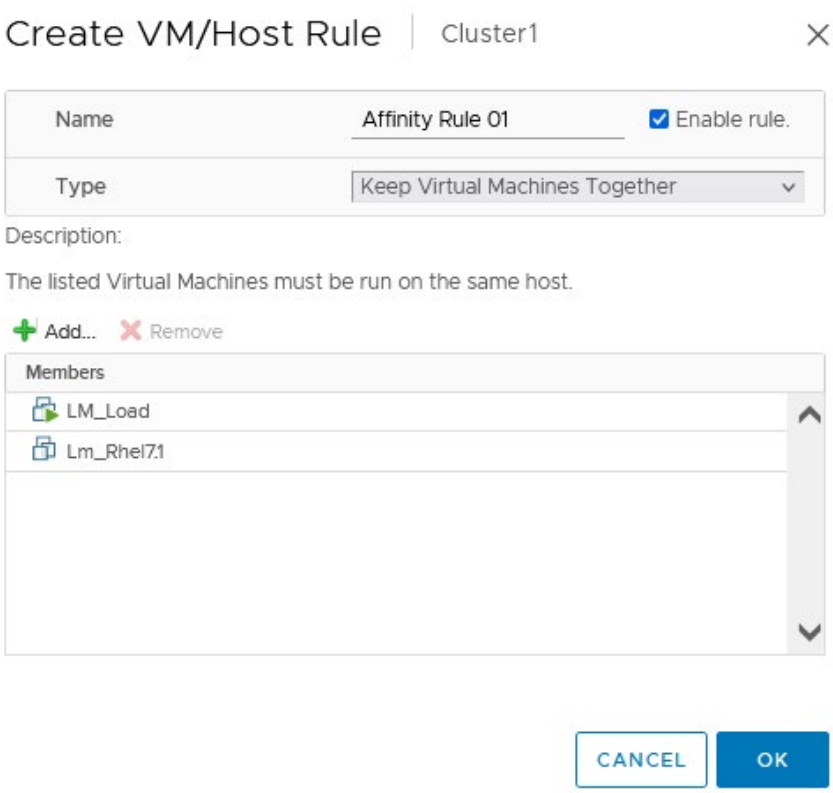


Figure 6.10

Other VM/Host rule types require creating VM groups and host groups first in VMware vSphere Client.

How to create VM/Host groups?

Follow these steps to create VM groups or host groups:

1. Select your cluster in the navigation pane and open the **Configure** tab.
2. Select **VM/Host groups** in the middle pane.
3. The *Create VM/Host Group* window is opened (see *Figure 6.11*).
4. Enter the group name, for example, *VM Group 1*.
5. Select the group type (**VM group** or **Host group**). Let's create a VM group first.
6. Click **+Add** to add virtual machines to the VM group.

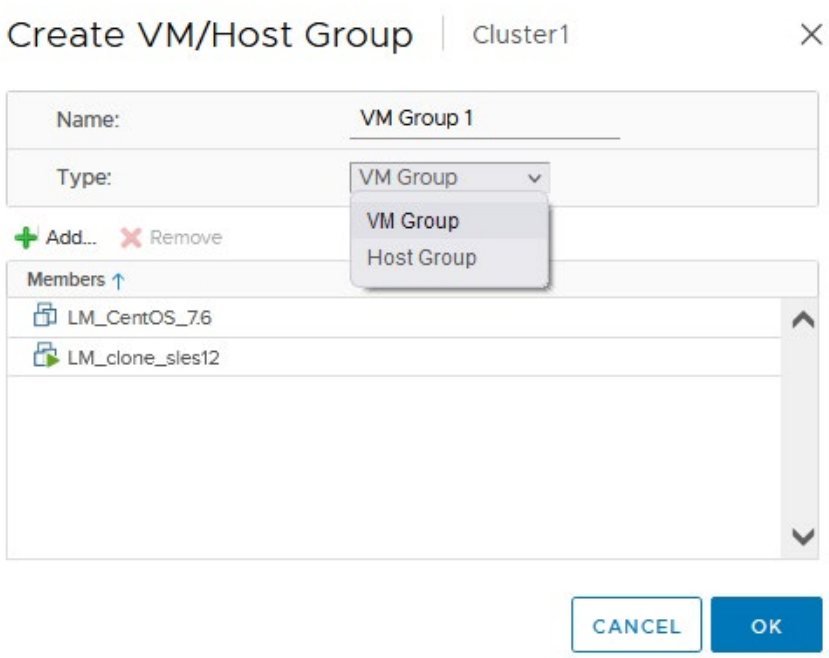


Figure 6.11

Similarly add ESXi hosts to host groups (see *Figure 6.12*).

When done, click **OK** to save the configuration and finish.

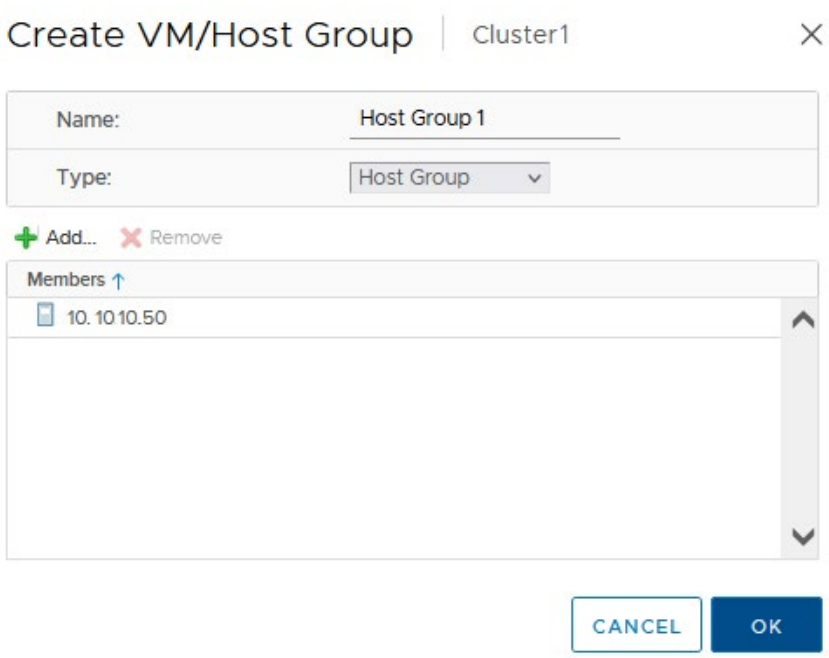


Figure 6.12

The created VM groups and host groups are now displayed in the appropriate section in the **Configure** tab (see *Figure 6.13*).

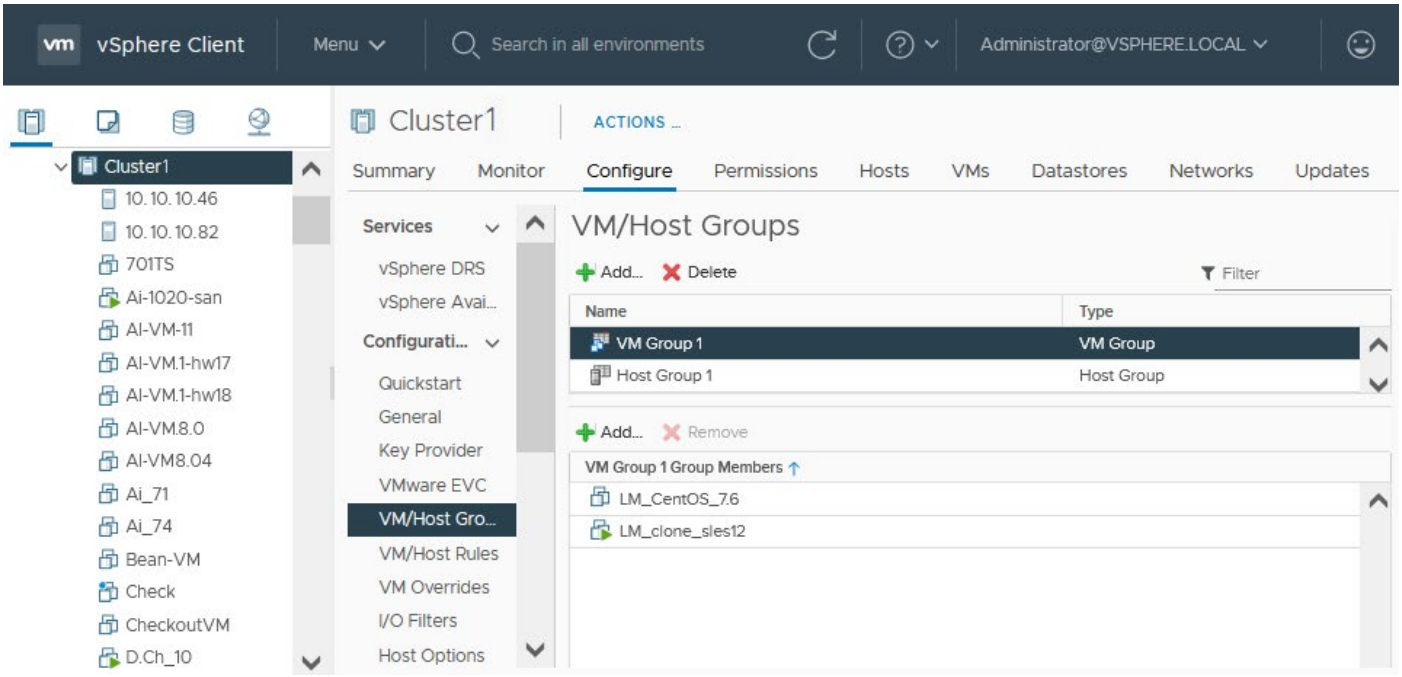


Figure 6.13

Virtual Machines to Hosts

Now, let's create a **Virtual Machines to Hosts** rule. Select the appropriate rule type in the drop-down list (see *Figure 6.14*). Select these parameters:

- **VM Group** – select VMs (a VM group) you want to run on the specified ESXi host or host group.
- Condition:
  - Must run on hosts in group
  - Should run on hosts in group
  - Must Not run on hosts in group
  - Should Not run on hosts in group
- **Host Groups** – select an ESXi host (host group) on which you want to run a VM group selected before.

In the case of configuring the **Virtual Machines to Hosts** affinity/anti-affinity rule and other rule types (except **Keep Virtual Machines Together**), you need to create VM groups and host groups first.

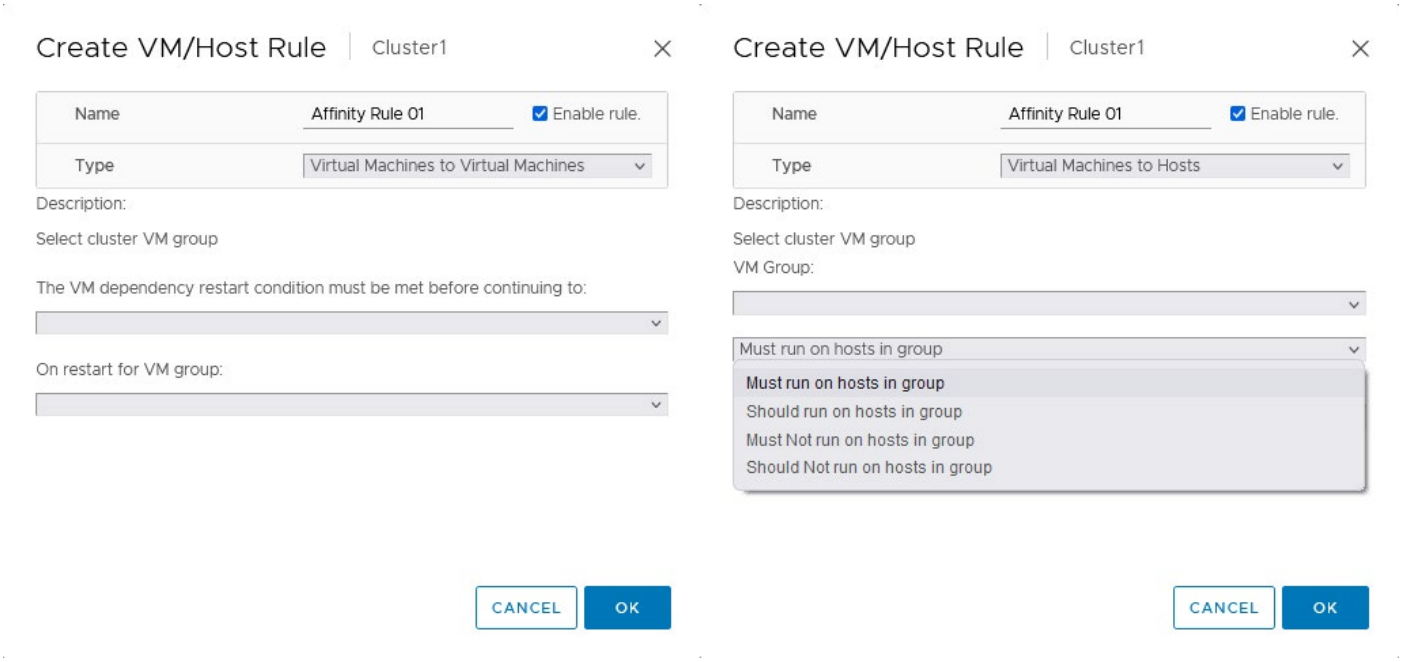


Figure 6.14

Once you have selected the needed parameter, you should see a screen like this (see *Figure 6.15*).

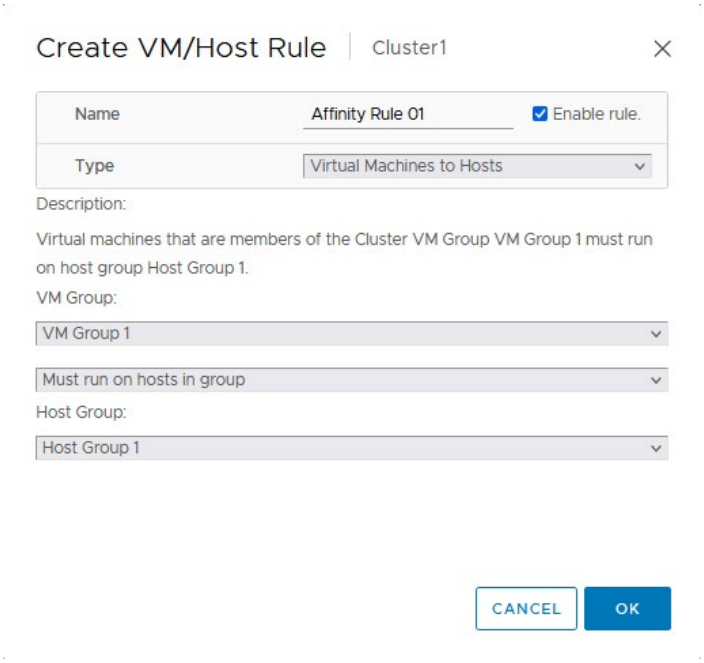


Figure 6.15

Virtual Machines to Virtual Machines

You can create this rule type by using a similar logic.

Select the **Virtual Machines to Virtual Machines** type in the drop-down list of the *Create VM/Host Rule* window (see *Figure 6.16*).

You need to have two VM groups created before you can configure this rule type.

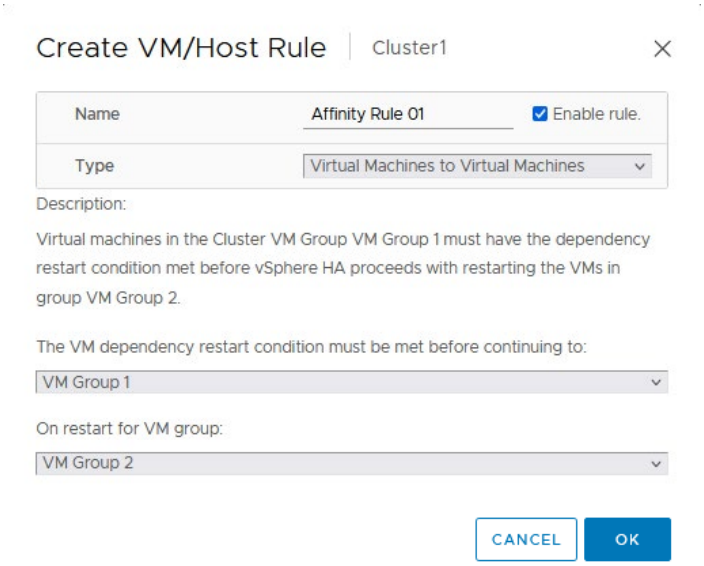


Figure 6.16

If you have not created VM groups, options in some drop-down menus may be empty (see *Figure 6.17*).

Remember to create VM/Host groups first.

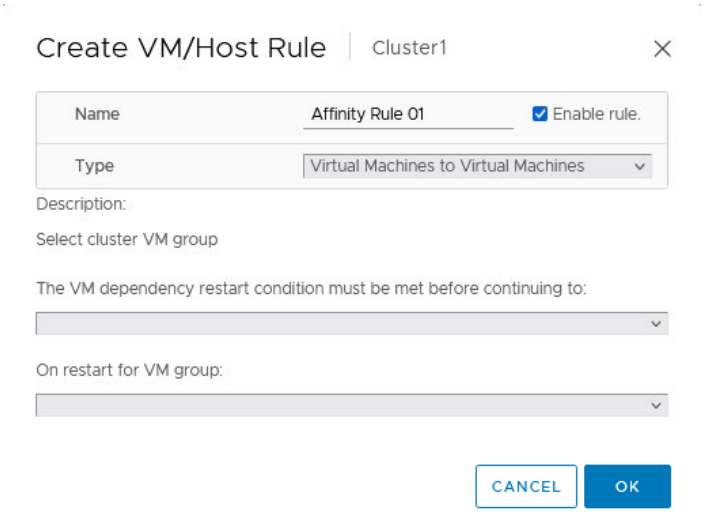


Figure 6.17

Hit **OK** to save affinity rule settings.

How to Configure a HA Cluster

Let's explore how to configure High Availability in a vSphere cluster.

1. Select the *Hosts and Clusters* view in VMware vSphere Client and in the navigation pane right-click your cluster name (see *Figure 7.1*).
2. Click **Settings** in the menu that opens.

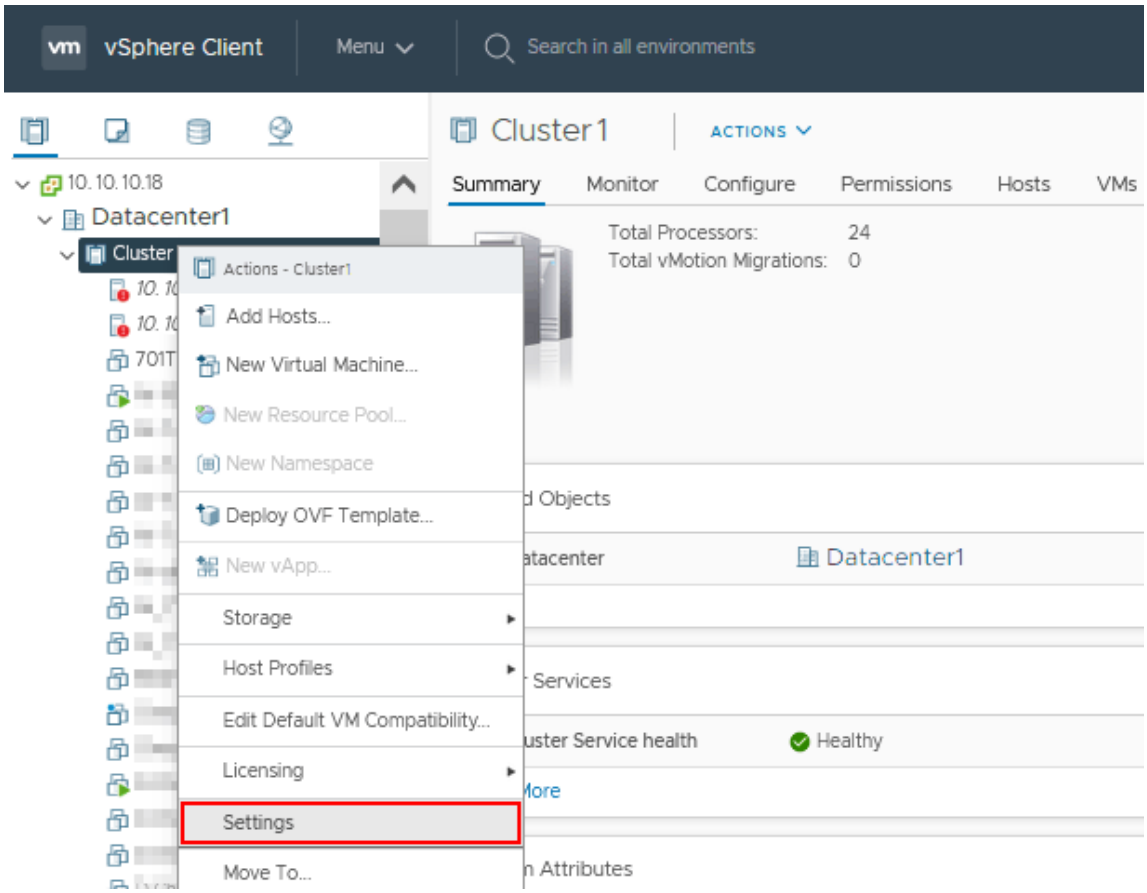


Figure 7.1

3. Now you should see the opened **Configure** tab and the selected **vSphere Availability** menu entry in the **Services** section.
4. Click **EDIT** in the top right corner of the window to enable, disable and configure VMware vSphere HA (see *Figure 7.2*).

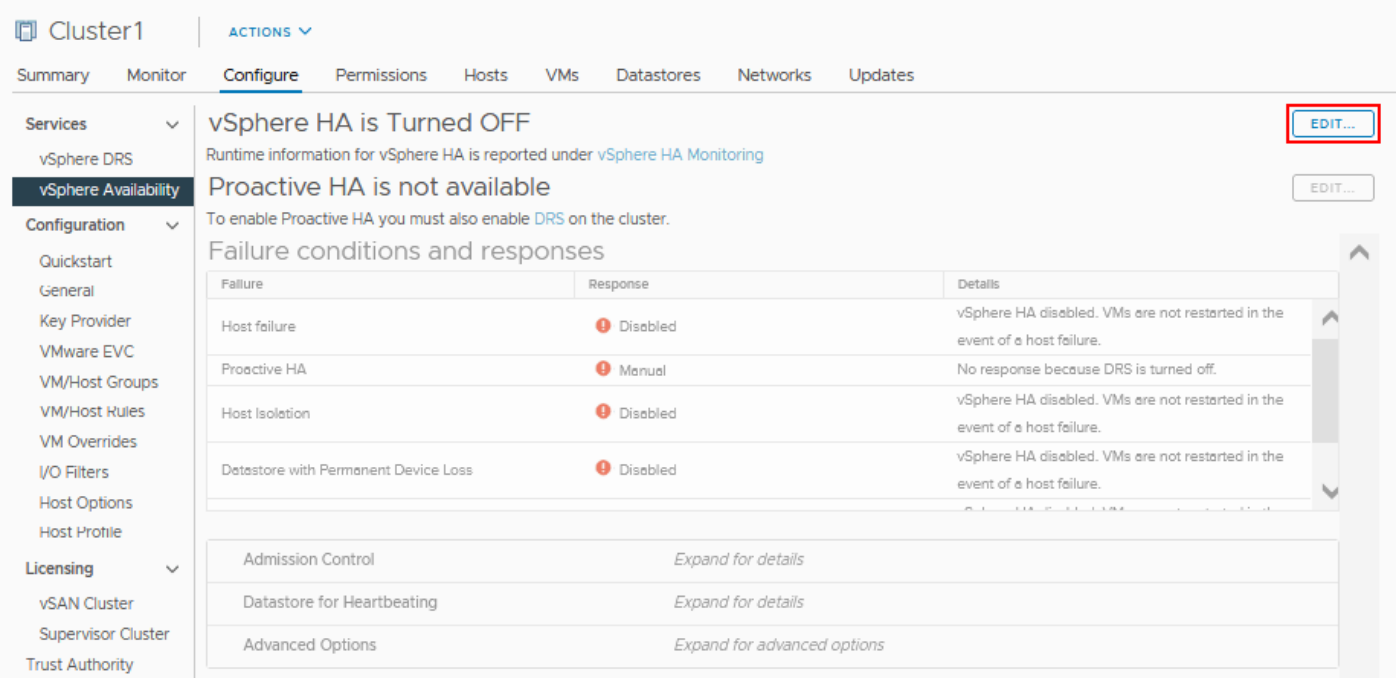


Figure 7.2

5. Click the **vSphere HA** switcher to enable or disable High Availability (see *Figure 7.3*).

   There are four tabs with HA options you can edit after enabling vSphere HA:

   - Failures and responses
   - Admission Control
   - Heartbeat datastores
   - Advanced Options

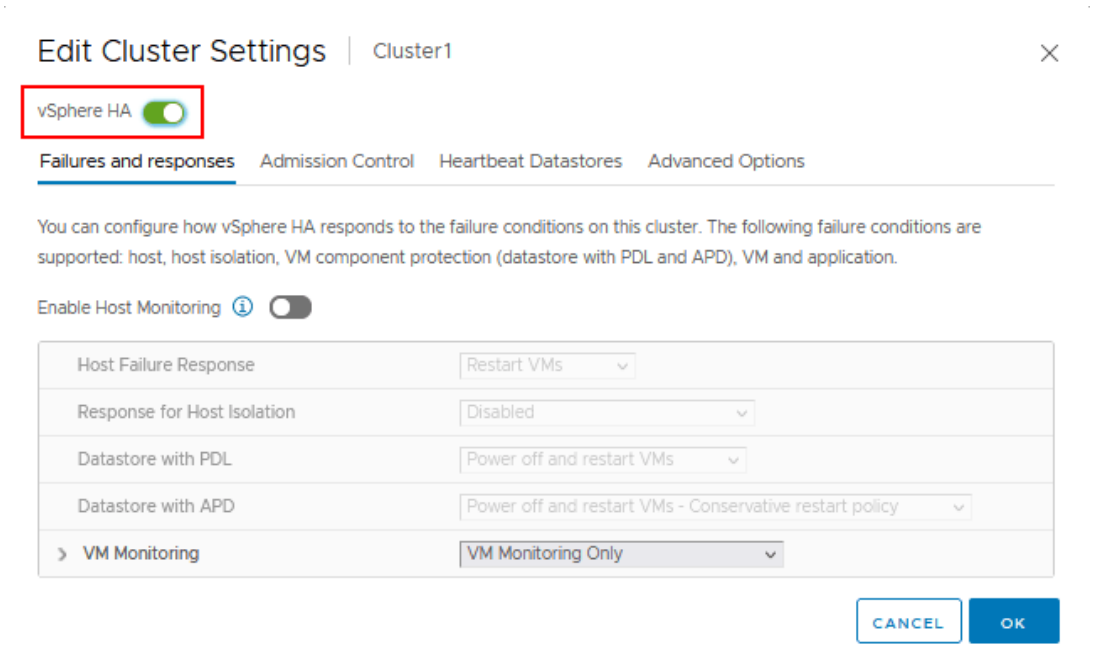


Figure 7.3

Let's look at each tab with the appropriate options.

Failures and responses

You can enable and configure Host Monitoring in this tab.

Host Monitoring: When this option is enabled, each ESXi host in the cluster is checked to ensure that this host is running. If a host failure occurs, virtual machines are restarted on another host. Host Monitoring is also required for the proper work of VMware Fault Tolerance recovery process. Remember to disable Host Monitoring when performing network maintenance. Host monitoring contains a set of options.

Host Failure Response

You can configure how VMware High Availability responds to ESXi host failure and isolation (see *Figure 7.4*).

Failure response allows you to configure ESXi host monitoring and VM failover in the cluster. Use this option to configure what vSphere HA must do in the case of an ESXi host failure. There are two options:

- Disabled
- Restart VMs

If you select the Restart VMs options, then the VM restart priority options are available for you.

Default VM Restart Priority determines the relative order in which virtual machines are restarted after a host failure. You can specify the priority needed: **Low, Medium, High**. You can set the main virtual machines, such as domain controller, database server, or email server, to restart with high priority.

VM Dependency Restart Condition. If the condition is met, then High Availability proceeds to restart the next VM according to the restart priority.

There are four restart conditions:

- Resource allocated
- Powered-on
- Guest heartbeats detected
- App heartbeats detected

Edit Cluster Settings | Cluster1

vSphere HA

Failures and responses | Admission Control | Heartbeat Datastores | Advanced Options

You can configure how vSphere HA responds to the failure conditions on this cluster. The following failure conditions are supported: host, host isolation, VM component protection (datastore with PDL and APD), VM and application.

Enable Host Monitoring

Host Failure Response

Failure Response	Allows you to configure host monitoring and failover on this cluster. ○ Disabled Host Monitoring is turned off. vCenter will not respond to host failures. ● Restart VMs When a Host failure is detected, VMs will be restarted in the order determined by their restart priority.
Default VM restart Priority	Medium
VM dependency restart condition	After the condition has been met, vSphere HA will proceed with the next VM restart priority. Resources allocated Additional delay: 0 seconds VM restart priority condition timeout: 600 seconds
Response for Host Isolation	Disabled
Datastore with PDL	Power off and restart VMs
Datastore with APD	Power off and restart VMs - Conservative restart policy
VM Monitoring	VM Monitoring Only

CANCEL OK

Figure 7.4

Response to Host Isolation

Sometimes you may have a situation when an ESXi host has not failed but is not responsive, for example, due to a network issue. Host isolation response allows you to configure what vSphere HA must do if a running ESXi host is not unreachable over the network (see *Figure 7.5*).

Host Isolation Response – three options are available here:

- **Disabled (Leave powered on)** – when network isolation occurs on the ESXi host, the state of virtual machines remains unchanged and the virtual machines on the isolated host continue to run, even if the host can no longer communicate with other hosts in the cluster. This setting also reduces the chances of a false positive.
- **Power off and restart VMs** – when network isolation occurs, all virtual machines are powered off and restarted on another ESXi host. This is a hard stop. A power-off response is initiated on the fourteenth second, and a restart is initiated on the fifteenth second.
- **Shut down and restart VMs** – when network isolation occurs, all virtual machines running on that host are shut down via VMware Tools and restarted on another ESXi host. This approach allows stopping the services and programs that are running on virtual machines correctly. If this is not successful within 5 minutes, the power-off response type is executed.

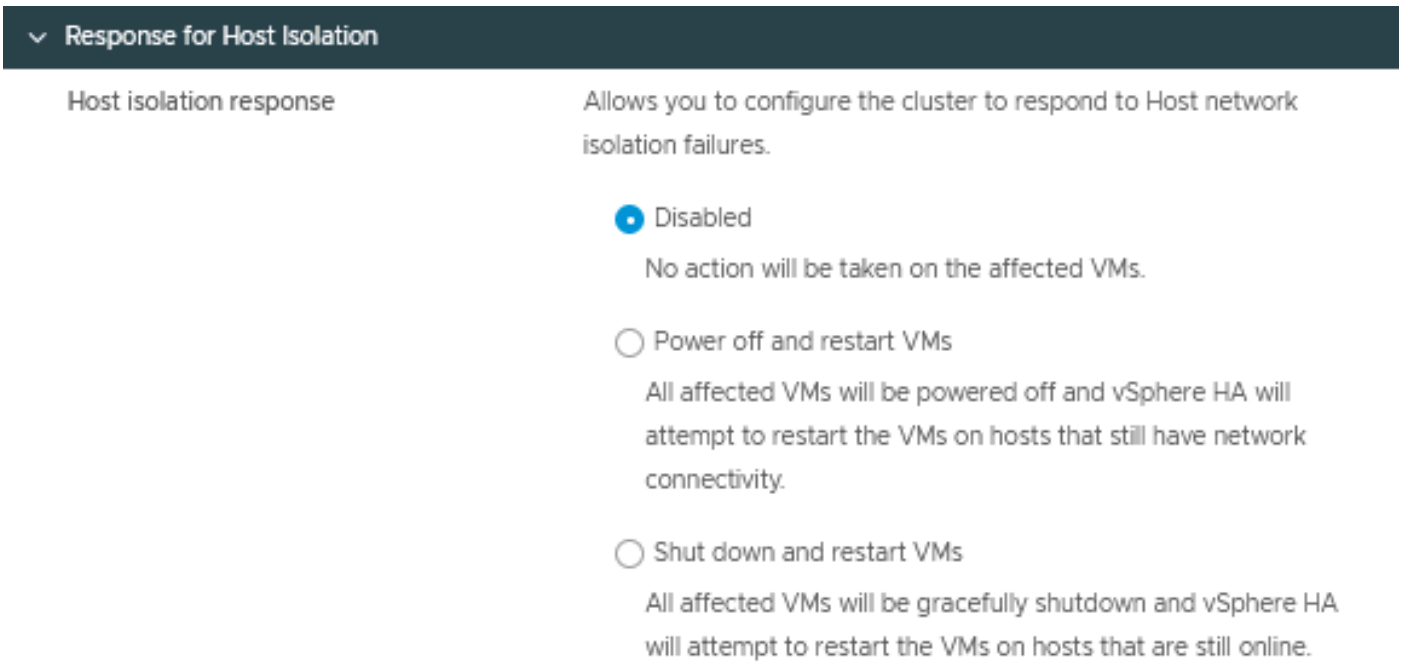


Figure 7.5

Datastore with PDL

Datastore with Permanent Device Loss (PDL) allows you to configure how the HA responds in situations when an ESXi host cannot communicate with a shared datastore even after multiple attempts (see *Figure 7.6*).

Datastore with PDL Failure has three options:

- Disabled
- Issue events
- Power off and restart VMs

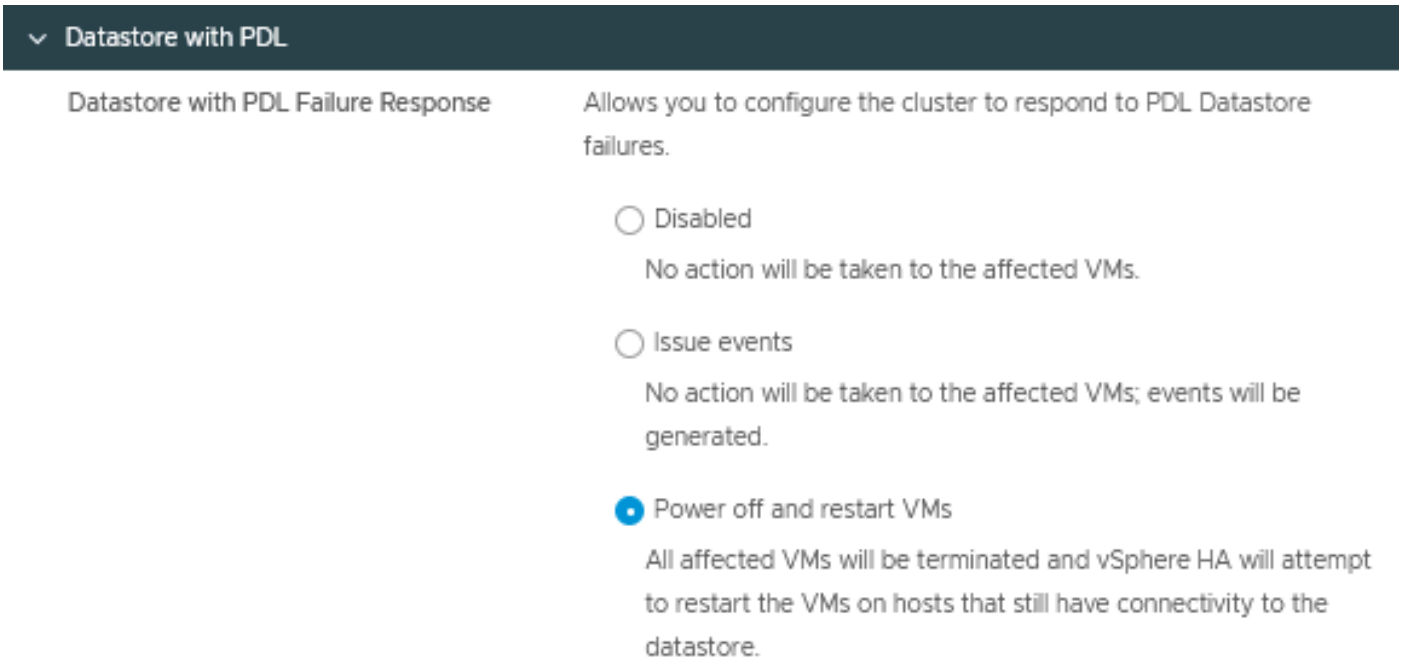


Figure 7.6

Datastore with APD

Datastore with All Paths Down (APD) is the option that allows you to configure the HA behavior in situations when all network paths to the datastore are offline (see *Figure 7.7*).

All Paths Down (APD) Failure Response has the following options:

- Disabled
- Issue events
- Power Off and restart VMs – Conservative restart policy
- Power Off and restart VMs – Aggressive restart policy

Response recovery is a parameter that determines how long to wait before taking action.

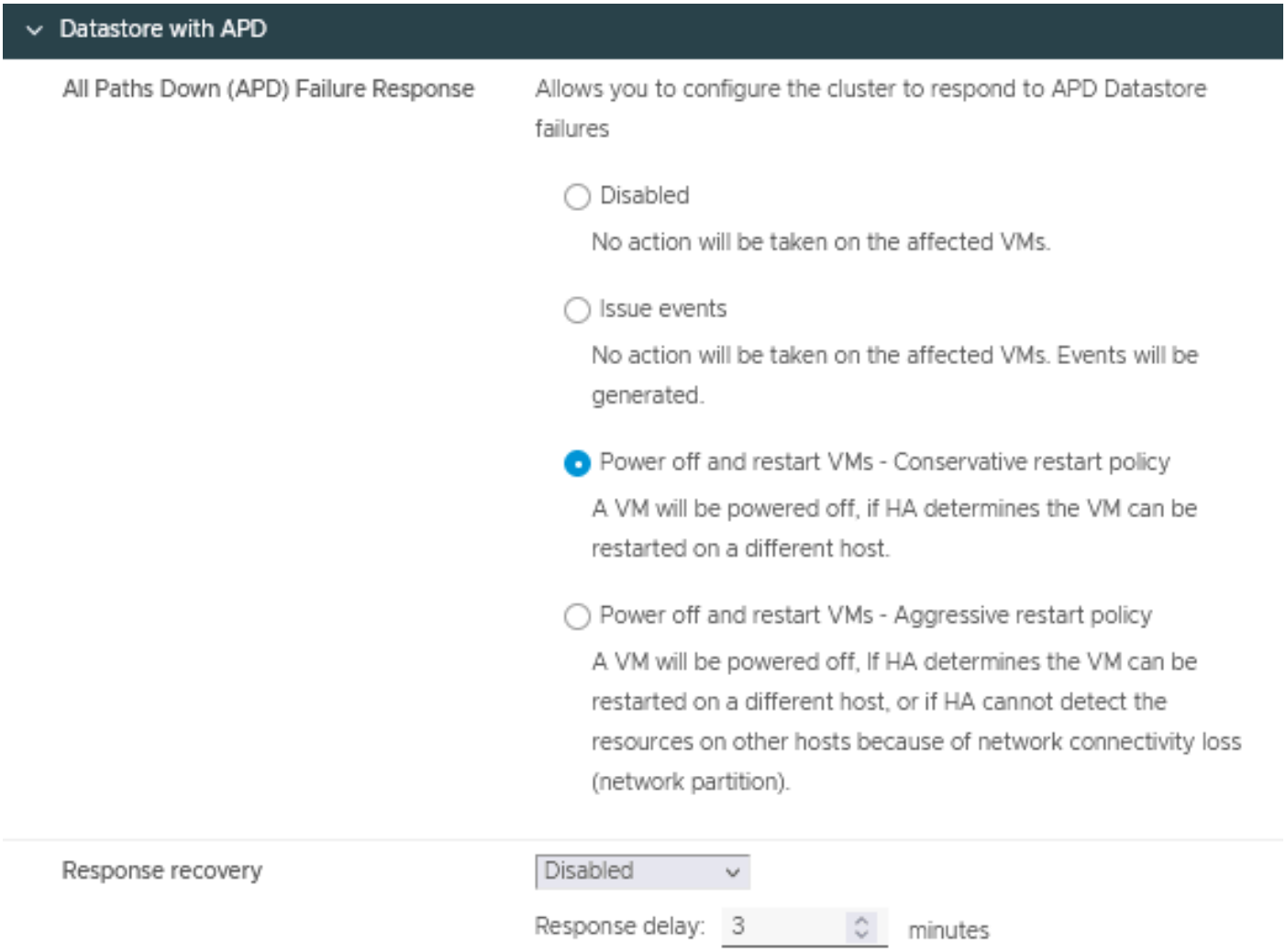


Figure 7.7

VM Monitoring

VM Monitoring allows VMware HA to monitor virtual machines and detect VM failures. You can determine what to do in case of VM failure and configure VM monitoring.

The VM monitoring service evaluates whether each virtual machine in the cluster is running by checking for regular heartbeats and input/output activity from the VMware Tools process running inside the guest. VM Monitoring is different from the host monitoring in that the item being watched is an individual virtual machine rather than an ESXi host. If vSphere can't detect VM heartbeats, the VM reboot happens. You can select the level of sensitivity using a preset or set the failure interval, the minimum uptime, and the maximum per-VM resets manually (see *Figure 7.8*).

The options for VM monitoring are:

- **Enable heartbeat monitoring:**
  - Disabled
  - VM monitoring only
  - VM and application monitoring
- **VM monitoring sensitivity:**
  - Preset
  - Custom

Edit Cluster Settings | Cluster1

vSphere HA

Failures and responses | Admission Control | Heartbeat Datastores | Advanced Options

VM Monitoring

Enable heartbeat monitoring

VM monitoring resets individual VMs if their VMware tools heartbeats are not received within a set time. Application monitoring resets individual VMs if their in-guest heartbeats are not received within a set time.

- Disabled
- VM Monitoring Only
  Turns on VMware tools heartbeats. When heartbeats are not received within a set time, the VM is reset.
- VM and Application Monitoring
  Turns on application heartbeats. When heartbeats are not received within a set time, the VM is reset.

VM monitoring sensitivity

- Preset
  Low — High
- Custom
  - Failure interval 30 seconds
  - Minimum uptime 120 seconds
  - Maximum per-VM resets 3
  - Maximum resets time window
    - No window
    - Within 1 hrs

CANCEL | OK

Figure 7.8

VM monitoring is disabled by default. Estimate risks of false-positive triggering before making the final decision to enable this feature.

Admission Control

Admission control is used by vCenter to ensure that sufficient resources are available in a cluster for failover protection. The cluster reserves resources to allow failover for all running virtual machines on the specified number of hosts. Admission control prevents you from powering on new VMs that can violate the *number of ESXi host failures to tolerate* within a cluster. The appropriate message is displayed if there is no free failover capacity and creating a new VM is not allowed.

Each Admission Control Policy has a separate Admission Control mechanism. Slots dictate how many virtual machines can be powered on before vCenter starts notifying you with the "Out of resources" message. The Admission Control process is a part of vCenter and not of the ESXi host (see *Figure 7.9*).

The percentage of Cluster Resources Reserved is the least restrictive and most flexible Admission Control Policy. 25% is the default reserving percentage, meaning that 25% of the total CPU and total memory resource across the entire cluster is reserved for the cluster.

Failover hosts are the ESXi hosts that are reserved for a failover situation. Failover hosts don't participate in DRS, and virtual machines can't run on these hosts in the regular mode.

Note
Remember to enable Admission Control because this option guarantees the ability of virtual machines to restart after a failure.

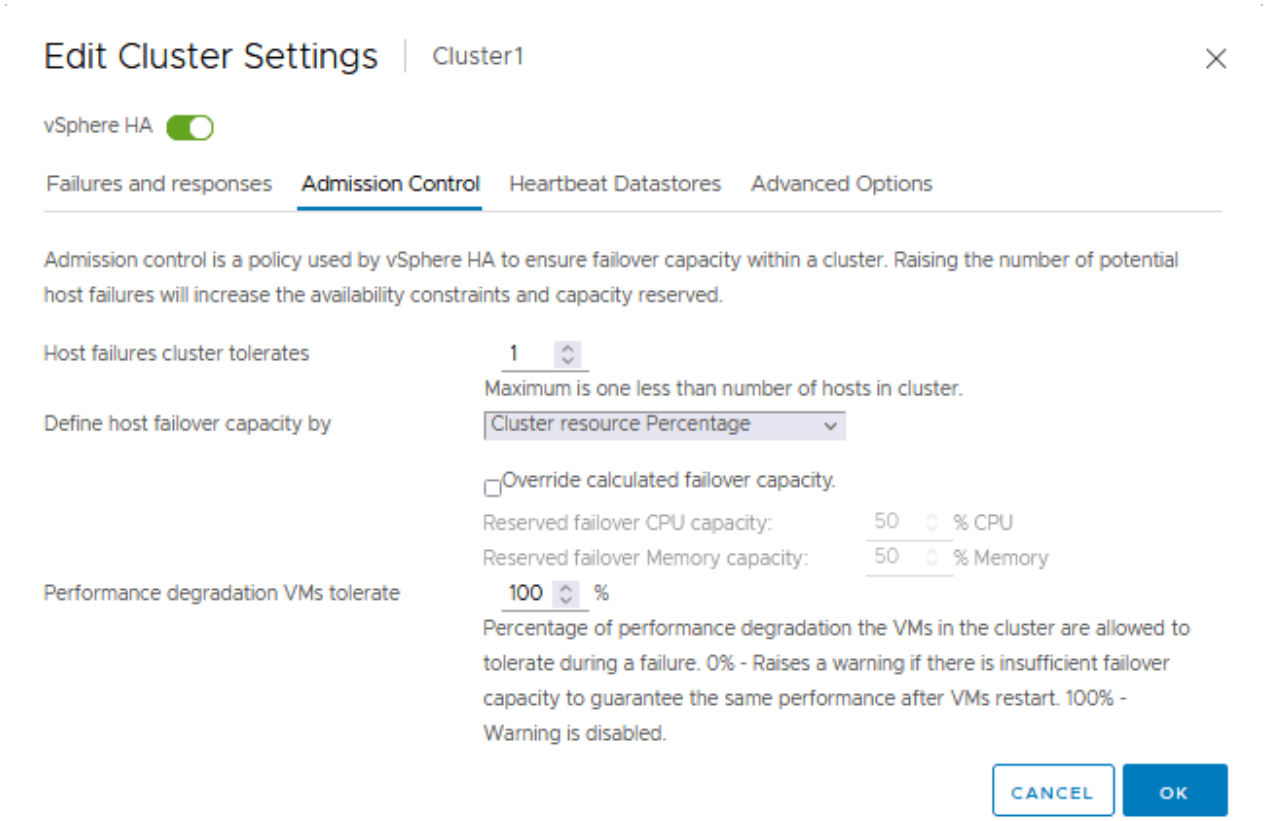


Figure 7.9

Heartbeat Datastores

Datastore heartbeating allows a master ESXi host in a cluster to distinguish the true state of a slave host. This situation may happen when ESXi hosts cannot communicate via the management network. If the management network of the ESXi host becomes isolated despite the virtual machines running, a restart signal is sent. Datastore heartbeating is used to determine more correctly a state of an ESXi host, even if the management network fails and, as a result, reduces the probability of false triggering of the virtual machine rebooting mechanism.

There are locking mechanisms to prevent concurrent usage of open files located on shared storage and avoid file corruption. HA manages the existing Virtual Machine File System (VMFS) locking mechanism that is also called a "Heartbeat Region", which is updated as long as the lock file exists. HA determines that at least one file is opened on the VMFS volume by checking files specially created for datastore heartbeating. These files have names like *VMname-hb, WindowsVM-hb*, *LinuxTest-hb, host1tst-hb,* etc. You can find them in the *.vSphere-HA* directory, which is located on a shared datastore with vSphere Client. Go to **Home -> Datastores -> DatastoreName -> Manage -> Files**. Don't delete or modify these files.

There are three options to configure Heartbeat Datastores (see *Figure 7.10*).

Heartbeat datastore selection policy:

- Automatically select datastores accessible from the hosts
- Use datastores only from the specified list
- Use datastores from the specified list and complement automatically if needed

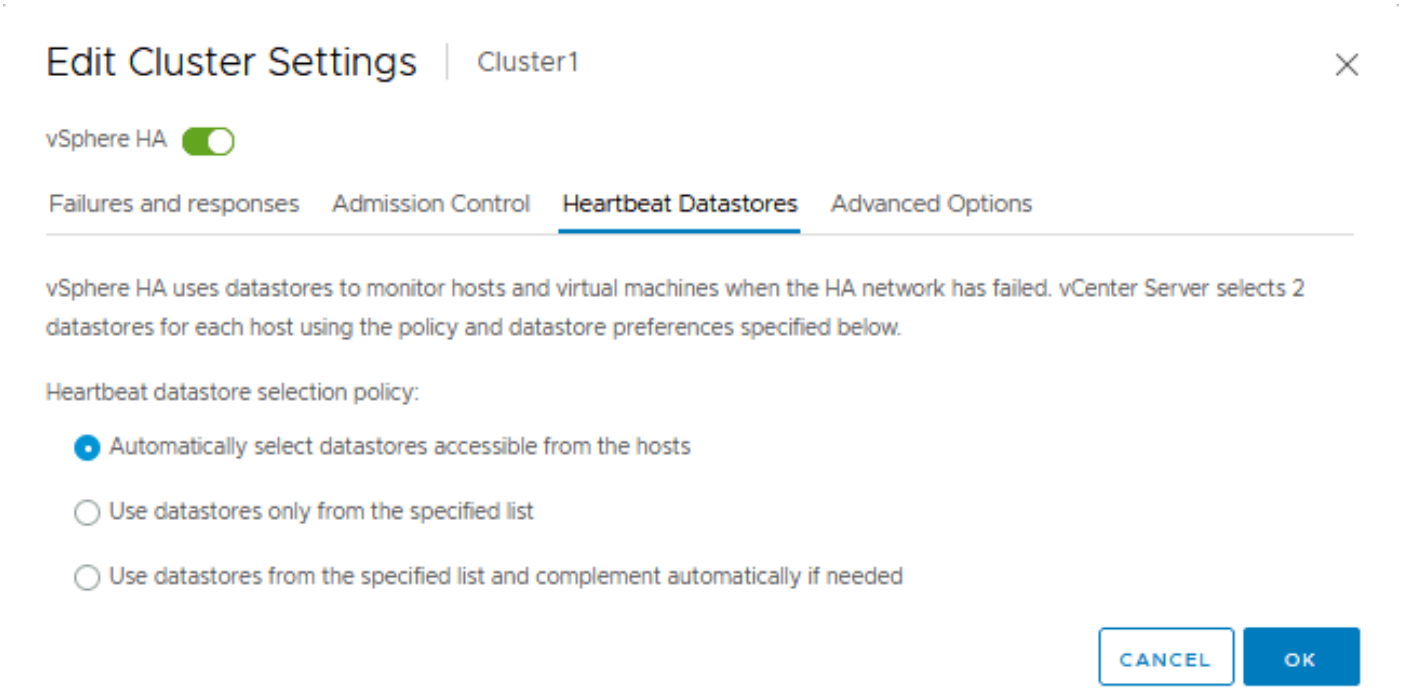


Figure 7.10

Advanced Options

You can manually set the advanced options to configure the behavior of your VMware High Availability cluster in an advanced manner (see *Figure 7.11*).

Examples of configuration parameters:

- das.isolationaddress
- das.usedefaultisolationaddress (true/false)
- das.isolationshutdowntimeout (300 is the default value)

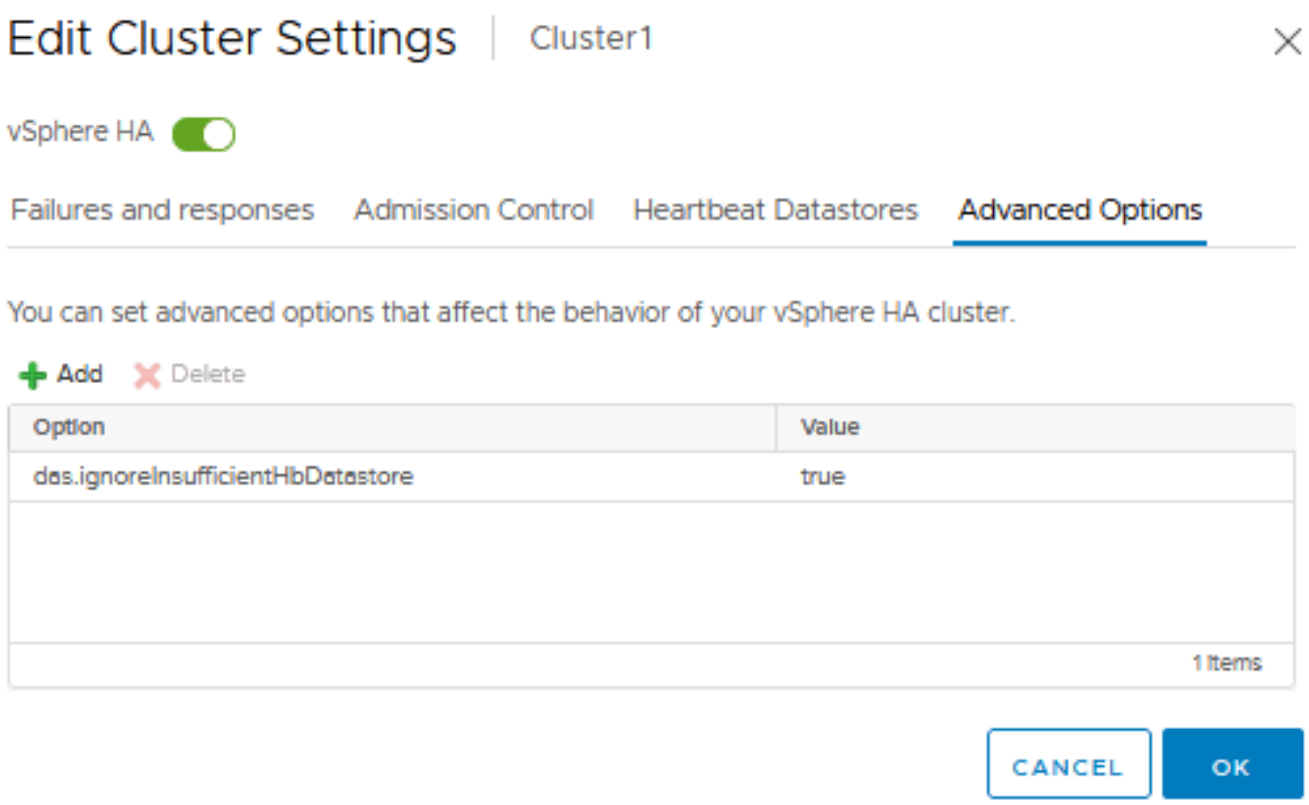


Figure 7.11

Proactive HA

Proactive HA is a feature that can help you avoid problems caused by hardware failures on ESXi hosts. Proactive HA monitors the ESXi hosts' hardware state, such as memory, storage, fan, network and power supply. If there is a monitored parameter pointing to the degraded health of the ESXi server hardware, Proactive HA generates a recommendation to migrate VMs from that host to other healthy hosts.

Thus you can configure Proactive HA to respond in situations when server failure may happen but has not happened yet. The main idea is to migrate VMs before a hardware failure occurs on an ESXi host and prevent the failure of VMs running on that host.

For example, if an ESXi server has two power supplies and one power supply fails, there is no redundancy from the power supply perspective. The ESXi server continues to work, but there is a single point of failure if the remaining power supply fails. In this case, vSphere Proactive HA initiates VM migration from a host with degraded hardware.

Proactive HA is not generally a feature of High Availability. This feature relied on vSphere DRS. For this reason, you need to enable DRS first, and then you can enable Proactive HA. Let's explain how to enable and configure Proactive HA.

1. Select your cluster in the *Navigation* pane by using VMware vSphere Client.
2. Click the **Configure** tab and select **vSphere Availability**.
3. Proactive HA is turned off by default. Click the **EDIT** button to change settings.

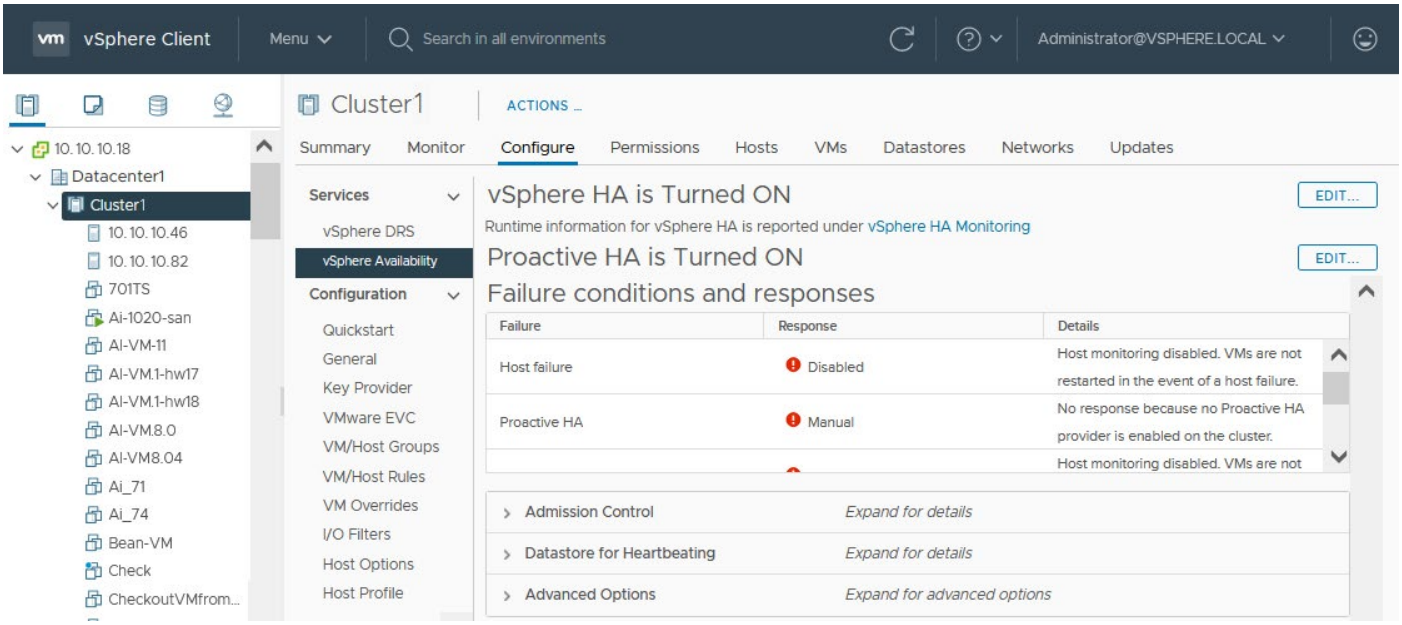


Figure 7.12

4. Click the **Status** switcher in the *Edit Proactive HA* window to turn on or turn off the feature.

Automation level:

- **Manual.** VMware vCenter provides suggestions and you should apply or reject them manually.
- **Automated.** VMs are automatically migrated to healthy ESXi hosts and a degraded host is entered into the maintenance mode.

Remediation allows you to select what to do with a partially degraded host:

- Quarantine mode for all failures
- Quarantine mode for moderate and Maintenance mode for severe failure (Mixed)
- Maintenance mode for all failures

Health providers are created in association with main server hardware vendors to get the hardware health data from sensors. The health provider reads, analyzes, and sends data to vCenter.

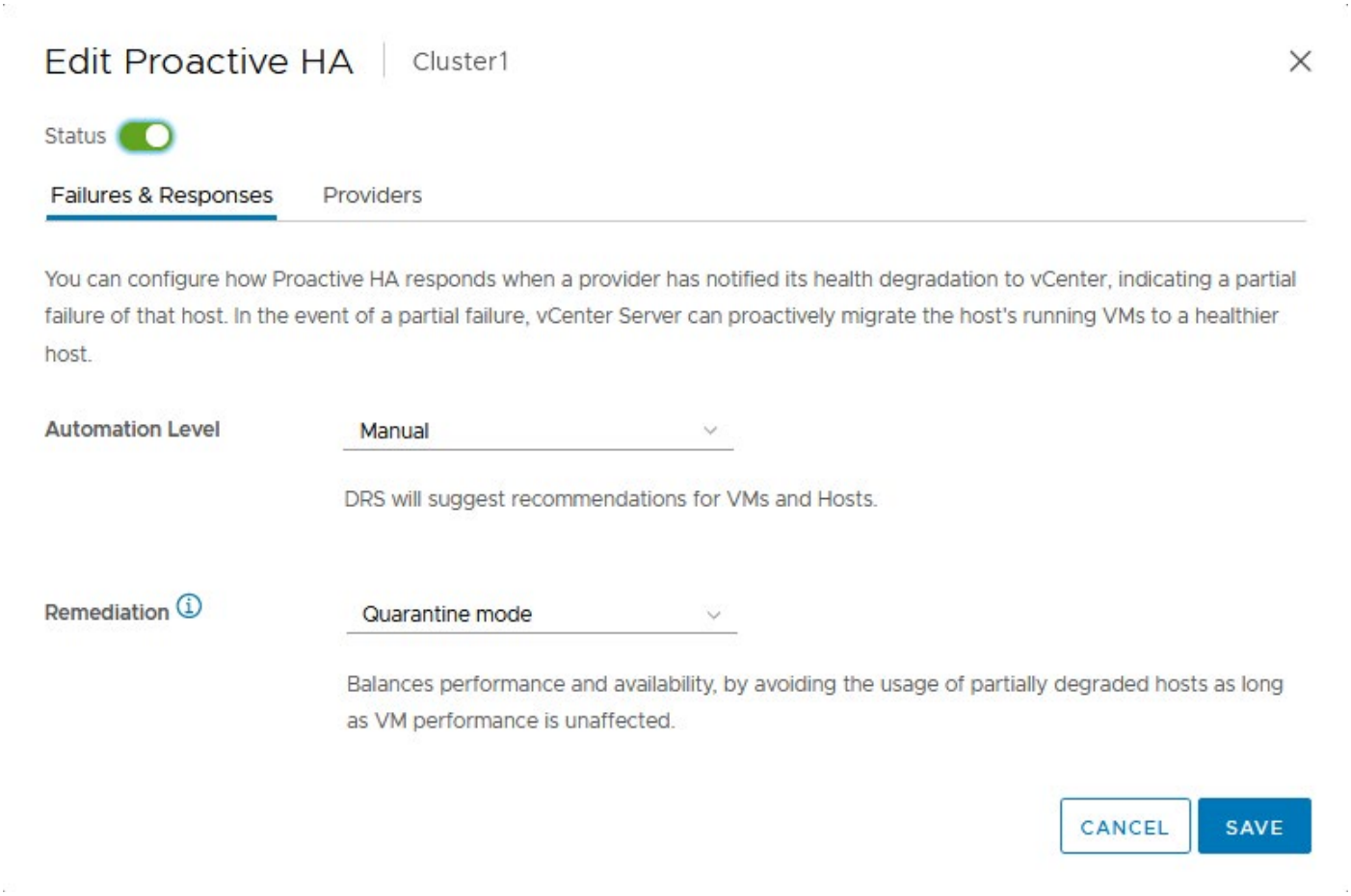


Figure 7.13

Once you have configured Proactive HA, the status is displayed on the *vSphere Availability* page in the *Configuration* tab.

Fault Tolerance: Purpose and Setup

Fault Tolerance (FT) provides continuous availability for virtual machines and enables a virtual machine to survive a physical server failure. This great feature allows the creation of an exact and continuously available replica on another ESXi host that can take over a virtual machine at the time of failure.

Fault Tolerance doesn't protect against software failures and operating system failures. Fault Tolerance replicates the primary VM state to the secondary VM. If, for example, a Blue Screen of Death (BSOD) is caused on the primary VM running windows, the secondary VM also crashes with BSOD.

In the High Availability mode, virtual machines need some time to load on another ESXi host after the ESXi host on which they were running fails. With Fault Tolerance, a virtual machine has a copy running on another ESXi host with a disabled VM network connection. If an ESXi host with a primary copy of VM fails, the secondary copy on another ESXi host needs only networking enabled. That is why the migration process looks seamless. If there are more than two ESXi servers included in the cluster, vSphere HA runs a new copy of the VM on the second ESXi Server at the moment of failure and creates a new VM replica on the third ESXi server.

You can use Fault Tolerance with DRS. It is recommended that you use three or more ESXi hosts in the cluster to provide better protection with Fault Tolerance.

Fault Tolerance Requirements

The Fault Tolerance requirements include the High Availability requirements and the requirements listed below:

- You need to have a dedicated 10-Gbit fault tolerance logging network configured for proper work of Fault Tolerance, as explained in the paragraph about network and storage configuration.
- A High Availability cluster must be configured.
- All ESXi hosts must be licensed for Fault Tolerance.

Fault Tolerance Limitations

There are some limitations for fault tolerance VMs you should take into account when enabling this feature in a HA cluster. The Fault Tolerance limitations in vSphere 7 are:

- VMware vSphere Standard and Enterprise licenses allow you to use up to 2 virtual CPUs on fault-tolerant VMs.
- VMware vSphere Enterprise Plus allows you to use up to 8 vCPUs on fault-tolerant VMs.

There are limitations that you can tune if the performance of your equipment used to run a cluster is enough:

- The maximum number of fault-tolerant VMs in the cluster (*das.maxftvmsperhost*). The default value is 4. To disable checking, set 0. This limit counts both primary and secondary VMs.
- The maximum number of virtual processors for all fault tolerance VMs on an ESXi host (*das.maxftvcpusperhost*). The default value is 8, but you can edit this parameter or disable checking. This limit counts both primary and secondary VMs.

Features that are not supported:

- VM snapshots
- Storage vMotion
- Disk encryption
- Linked clones
- Virtualization-based security
- Trusted Platform Module

There are unsupported devices for fault-tolerant VMs, including:

- CD-ROM and floppy devices that are backed to physical devices on ESXi servers.
- Physical raw device mapping (RDM)
- USB and sound devices
- Video devices with 3D enabled
- Virtual disks whose size is larger than 2 TB
- Parallel and serial ports

How to Enable Fault Tolerance

To enable FT for virtual machines, do the following actions in VMware vSphere Client:

Select your cluster and go to the list of virtual machines.

Right-click on the virtual machine you want to make fault-tolerant: **All vCenter Actions > Fault Tolerance > Turn On Fault Tolerance** (see *Figure 8.1*).

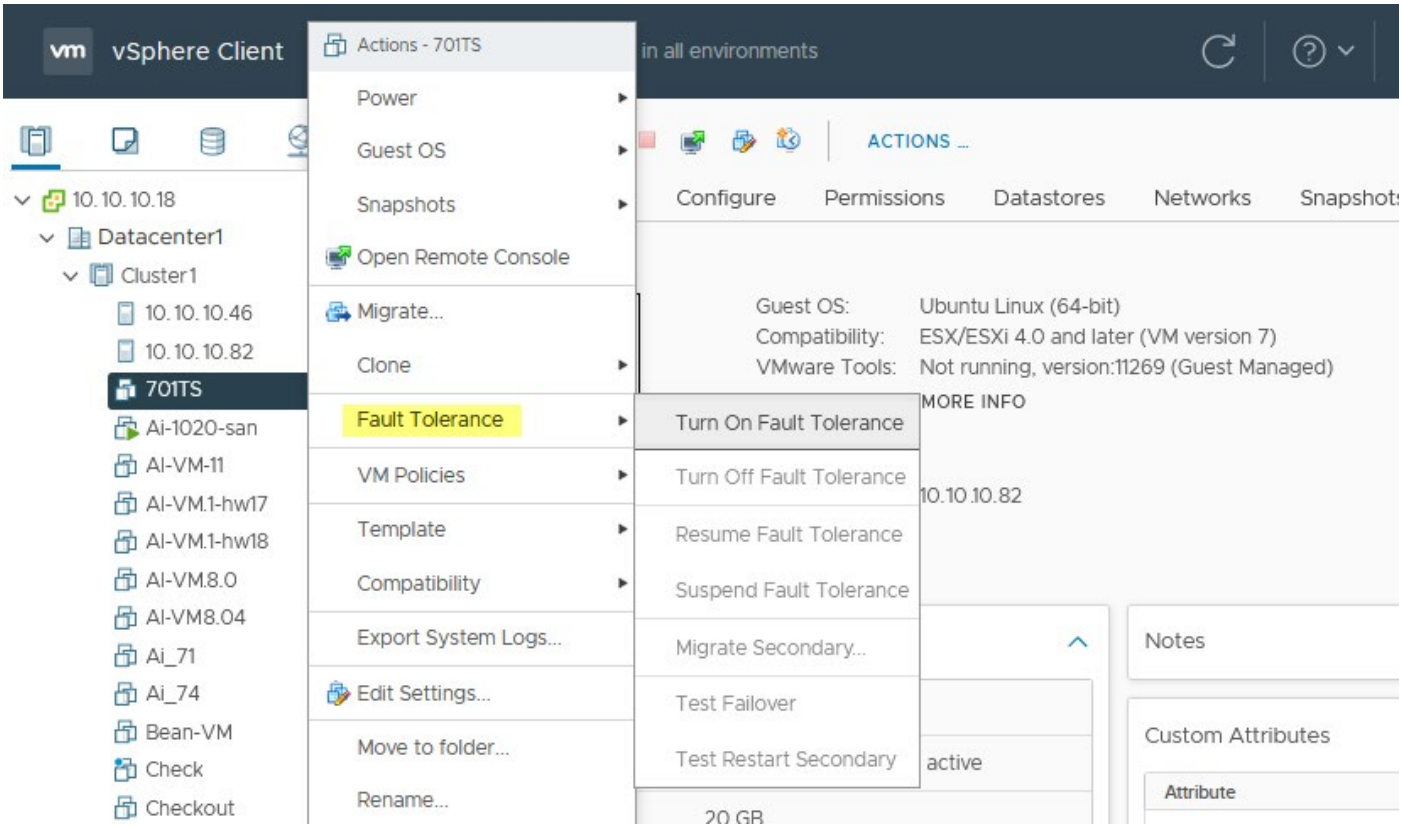


Figure 8.1

A three-step wizard is opened.

1. **Select datastores.** Select a shared datastore available for your cluster to store files of the secondary VM (see *Figure 8.2*). Click **Next** at each step to continue.

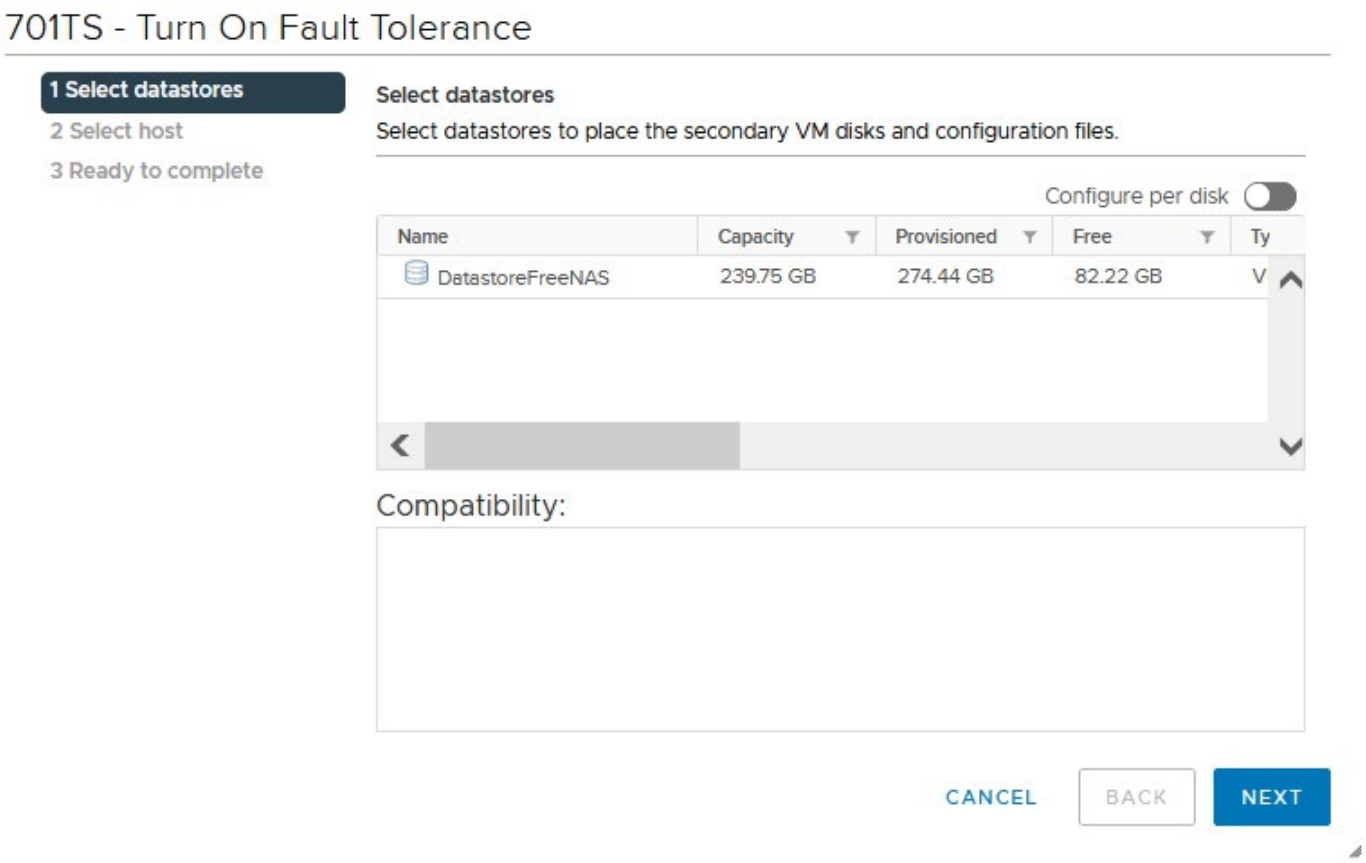


Figure 8.2

2. Select host. Select a host from your cluster to place the secondary VM (see *Figure 8.3*).

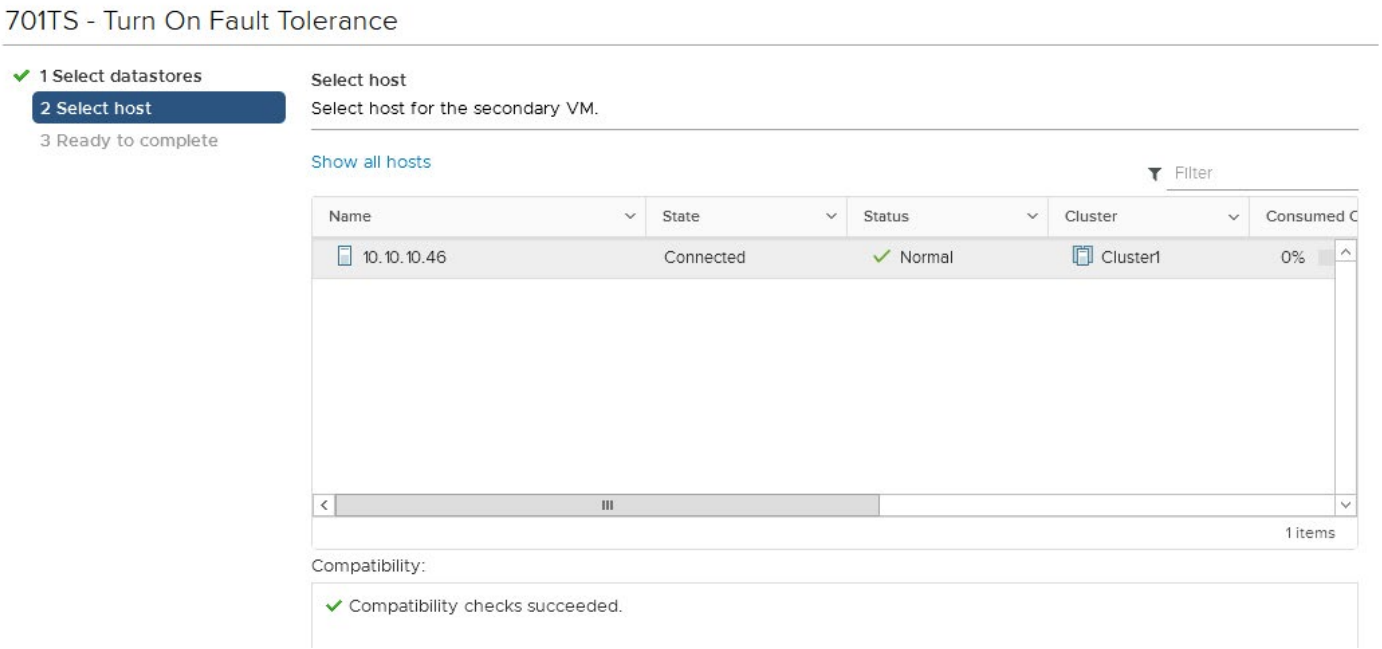


Figure 8.3

3. **Ready to complete.** Review your selections and hit Finish to enable fault tolerance for this VM (see *Figure 8.4*).

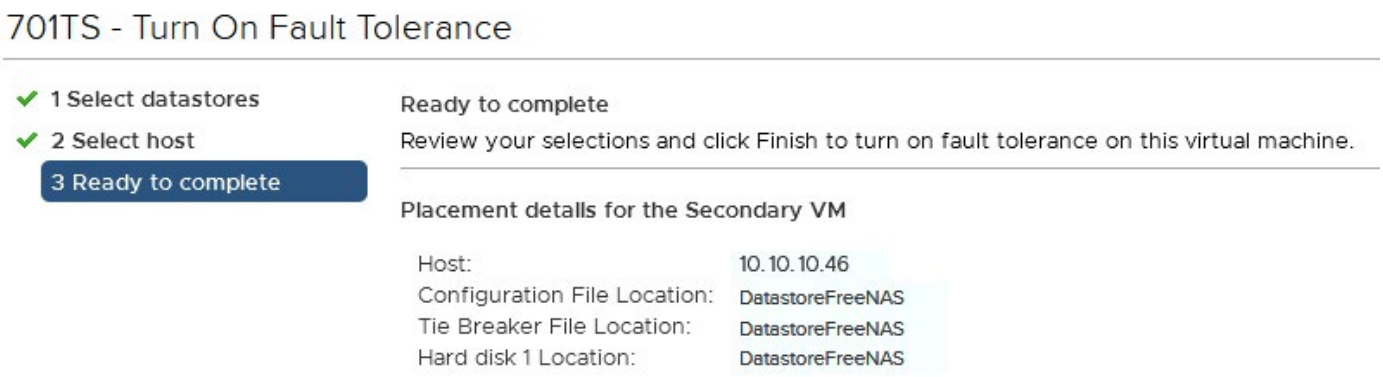


Figure 8.4

Now you can set up **Latency Sensitivity** (the high-performance mode of a virtual machine). Go to **VM > Manage > Settings > Advanced settings** (see *Figure 8.5*).

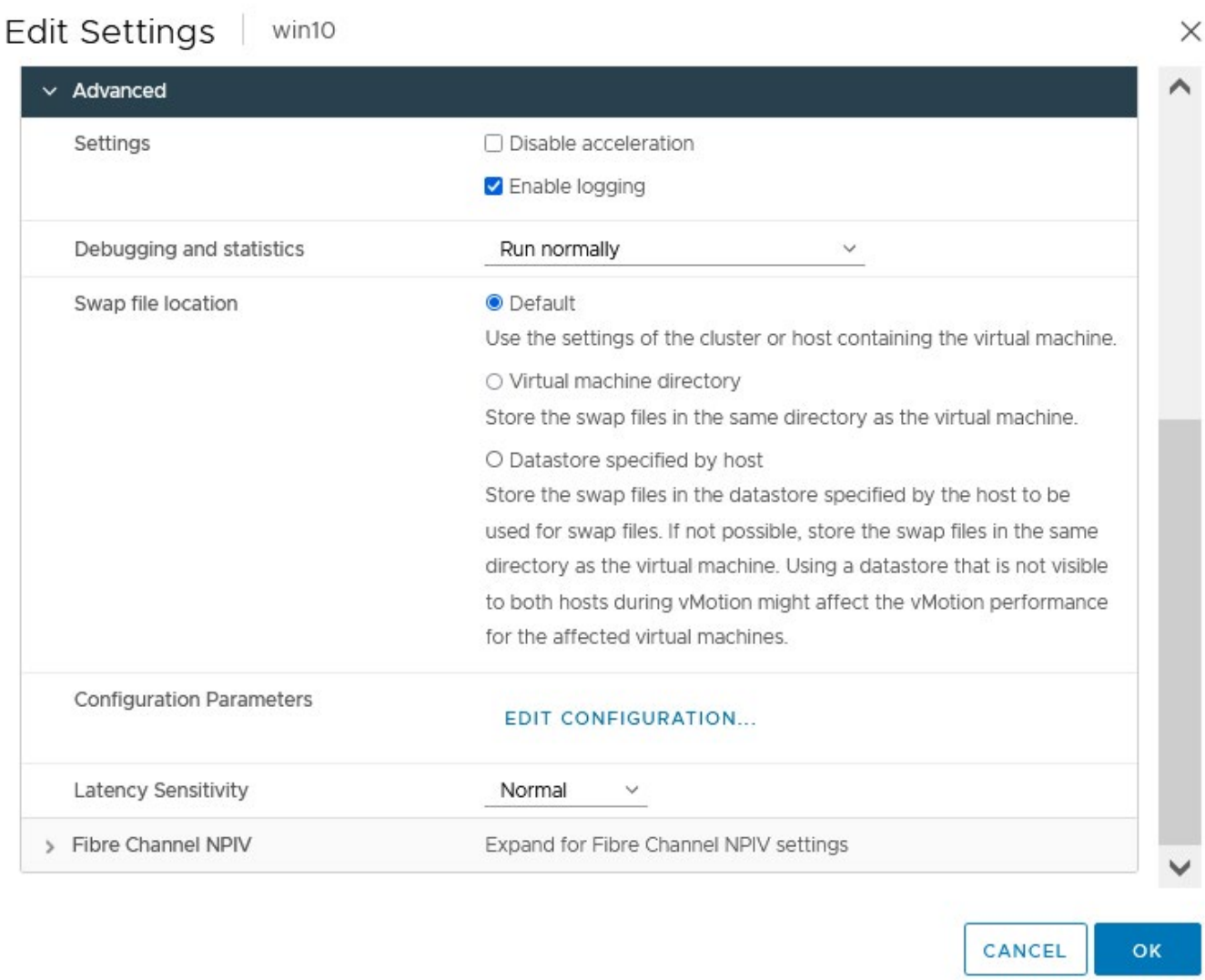


Figure 8.5

With "**High**" Latency Sensitivity, the ESXi host provides vCPU access to the physical CPU, while calculating the actual CPU load. With this option enabled, a virtual machine processor can directly interact with the physical processor without using the VMkernel scheduler. Thus, the Latency Sensitivity mode is useful for virtual machines demanding high performance.

How Fault Tolerance works

Here is an example of how High Availability and Fault Tolerance features work.

Both ESXi servers are running in a High Availability cluster. The virtual machine VM2 is running on ESXi Server 1 with the Fault Tolerance option enabled and has an exact replica with disabled networking running on ESXi Server2. VM1 is also running on ESXi Server 1, but the Fault Tolerance option is disabled for this virtual machine (see *Figure 8.6*):

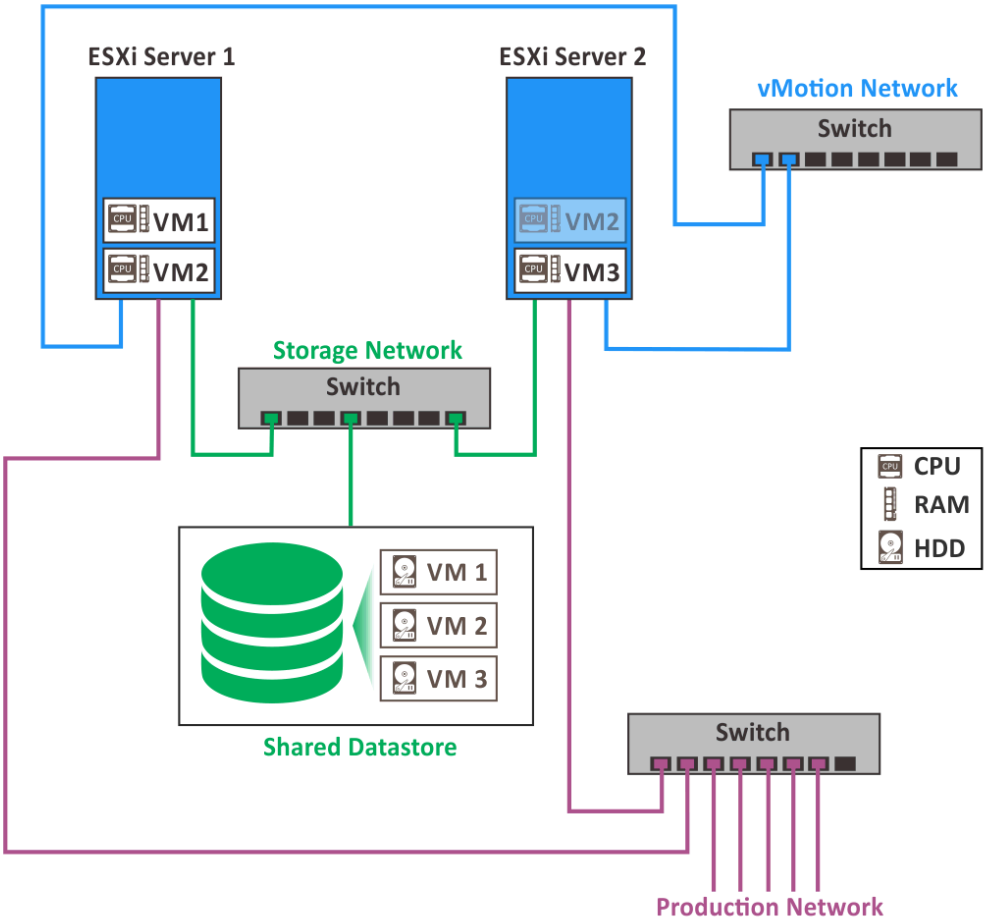


Figure 8.6

The ESXi Server 1 failure occurred. VM2 that was running on ESXi Server 1 also fails. However, the replica of VM2 that is still running on ESXi Server 2 becomes reachable in a moment by enabling networking and output operations. This is possible due to the automatic failover protection provided by the VMware Fault Tolerance feature. The VM2 failover is seamless and instant. At the same time, VM1 becomes unreachable and migration of this virtual machine to ESXi Server 2 has just started. Loading a virtual machine with the operating system and other services may take some time (see *Figure 8.7*):

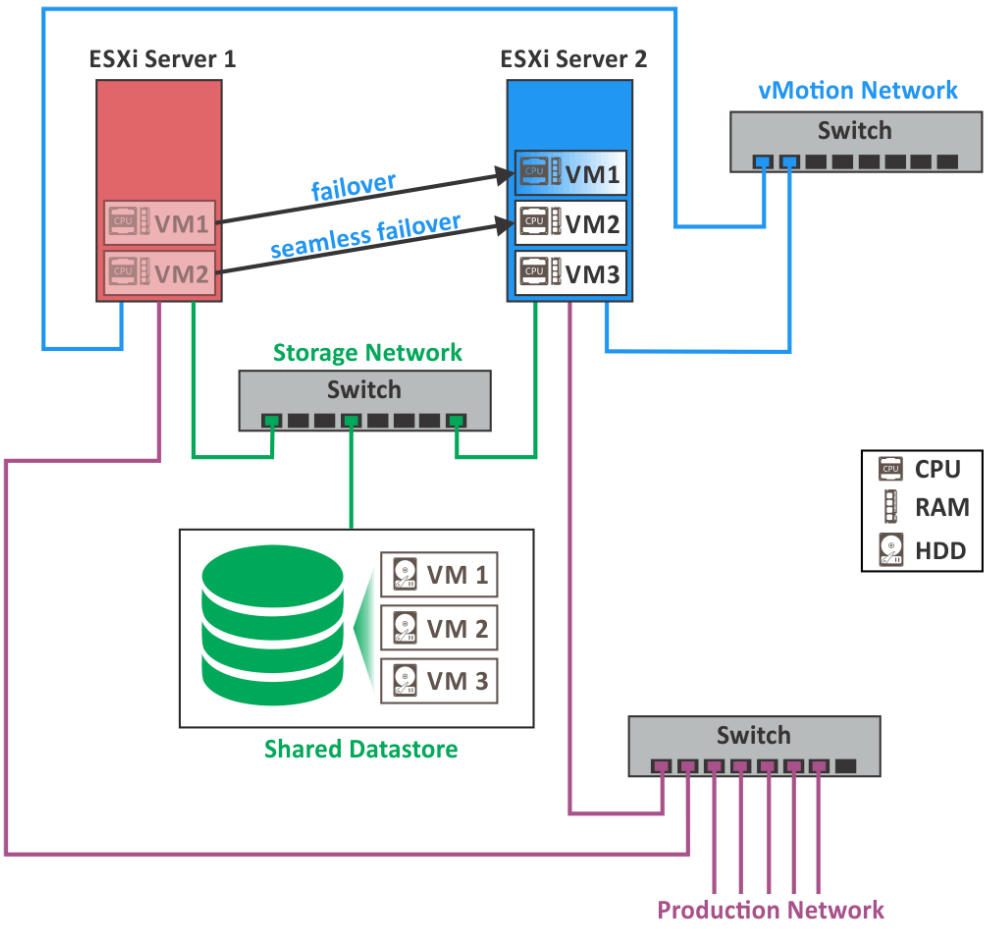


Figure 8.7

How to turn off and disable Fault Tolerance

If you decide to turn off Fault Tolerance, right-click on the respective virtual machine and select **Fault Tolerance > Turn Off Fault Tolerance** or **Fault Tolerance > Suspend Fault Tolerance**.

There is a difference between suspending (disabling) Fault Tolerance and turning off Fault Tolerance. If you disable FT, the secondary virtual machines are preserved with their configuration and history. Using this option allows you to re-enable FT in the future. Turning off VMware FT deletes all secondary virtual machines, their configuration, and the entire history. Use this option if you do not plan to re-enable VMware FT (see *Figure 8.8*).

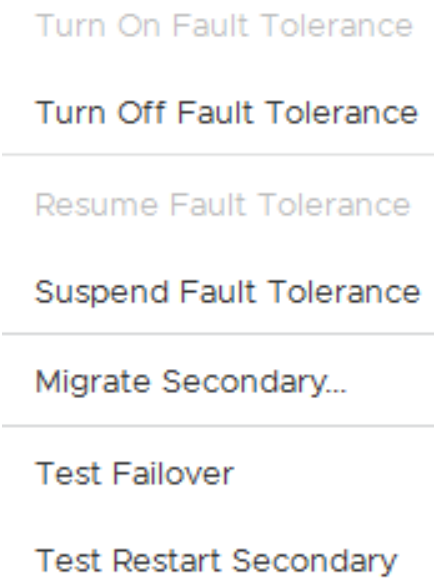


Figure 8.8

The explanation of other options in the Fault Tolerance menu:

- **Migrate Secondary** – use this option to migrate a secondary VM (ghost VM clone) to another ESXi host.
- **Test Failover** – test whether a secondary VM can fail over and become a primary VM.
- **Test Restart Secondary** – allows you to test whether a secondary VM can restart successfully.

Removing Hosts from the Cluster

If, for any reason, you decide to remove an ESXi host from the cluster, do the following:

1. Power off all virtual machines that are running on the host.
2. Right-click the ESXi host and select **Enter Maintenance Mode** (see *Figure 9.1*).

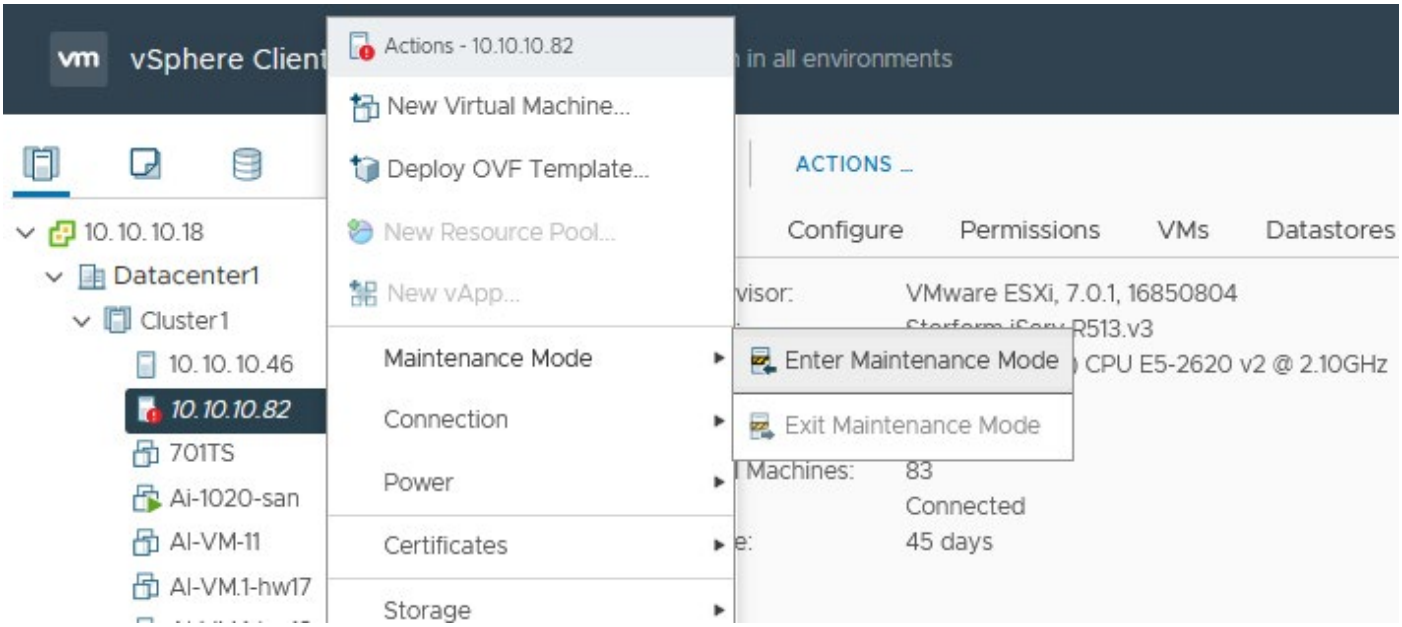


Figure 9.1

3. Right-click the ESXi host and select **Move To** (see *Figure 9.2*).

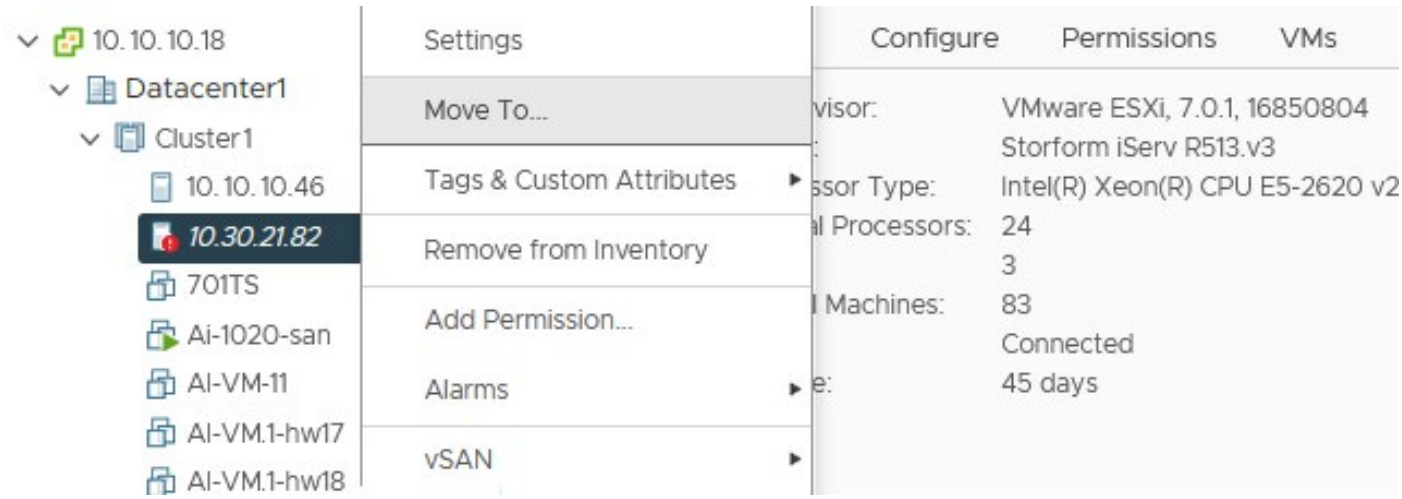


Figure 9.2

4. Select a new location for the host, for example, one of the available Datacenters (see *Figure 9.3*).

Figure 9.3

5. Right-click the ESXi host you have excluded from the cluster and select **Exit Maintenance Mode** (see *Figure 9.4*).

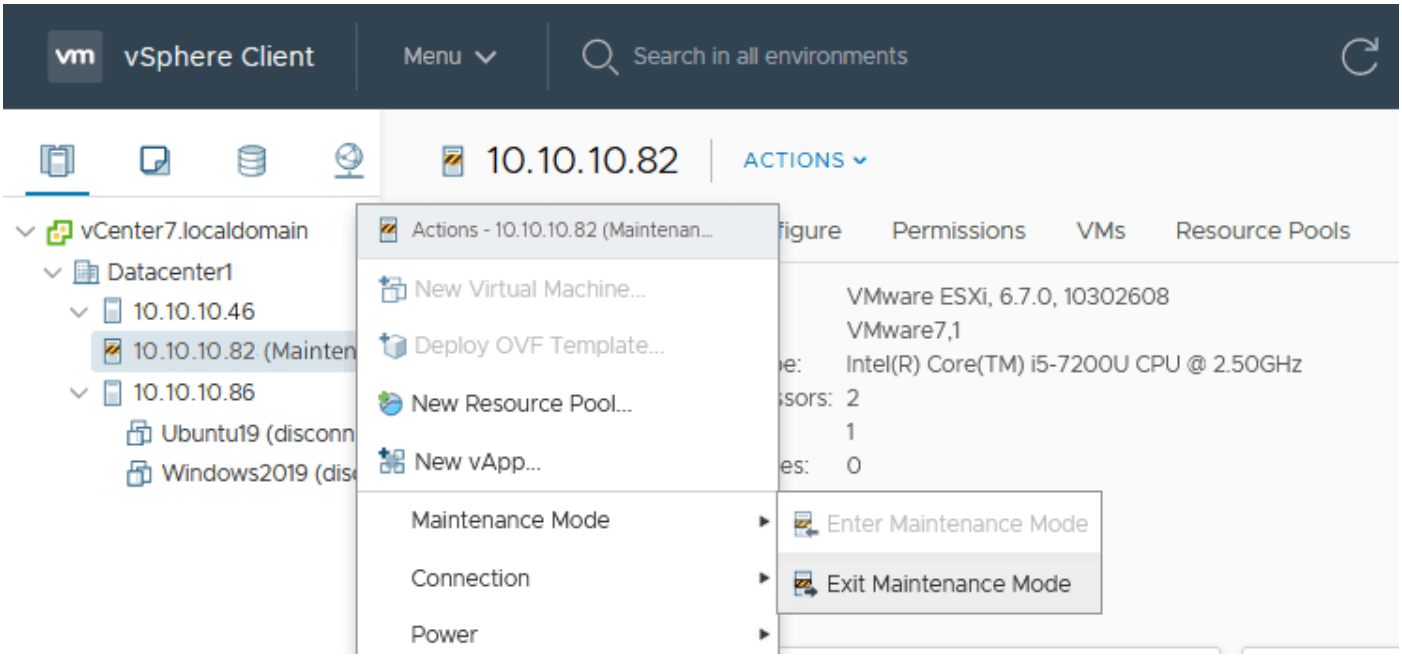


Figure 9.4

Similarly, you can remove other ESXi hosts from a vSphere cluster.

Conclusion

VMware vSphere is a virtual environment with a long list of features that help manage virtual machines, as well as provide great capability, reliability, and scalability. Clustering technologies are widely used in vSphere to connect servers over the network and achieve better performance in executing resource-intensive tasks. VMware supports creating Distributed Resource Scheduler (DRS) clusters and High Availability (HA) clusters.

Creating a DRS cluster helps improve performance through rational usage of computing resources. An HA cluster reduces the downtime of virtual machines in the event of failure by restarting VMs on another host via the redundant network. The Fault Tolerance feature of HA clusters ensures avoiding downtime and provides for the seamless migration of virtual machines from the failed host to the running host. That is vital for business-critical processes. Using vSphere HA and DRS together combines automatic failover with load balancing. This helps provide a more balanced cluster after vSphere HA moves virtual machines to different hosts.

High Availability and Fault Tolerance do not replace the need for data backup. In the event of cluster usage, virtual machines are stored on a shared datastore and should be backed up to another storage. A combination of VMware cluster features with backup ensures efficient resource management, increased reliability, and protection.

NAKIVO Backup & Replication at a Glance

NAKIVO Backup & Replication is a reliable data protection solution for all workloads. The solution offers backup, replication, instant granular recovery, ransomware protection, and IT monitoring for different IT infrastructures from a single pane of glass.

All-in-One Data Protection
Protect VMware vSphere, Microsoft Hyper-V, Nutanix AHV, Amazon EC2, Windows, Linux, Microsoft Office 365, NAS, and Oracle Database environments.

Flexible Installation Options
Install on Linux, Windows and NAS (such as Synology and QNAP), or deploy as a VMware vSphere VA, Nutanix AHV VA or Amazon Machine Image.

IT Monitoring
Get complete visibility through pie and line charts to visualize the virtual environment's performance and health metrics.

Excellent Support
Get free demos and deployment sessions for you and your clients to get you started with NAKIVO Backup & Replication. 24/7 tech support when needed to ensure the strictest SLAs.

Ransomware Protection
Immutable backups protected from deletion and encryption by ransomware in Amazon S3 and Linux-based repositories; air-gapped backup with offline storage and tape.

Backup Data Tiering
Backups and backup copies on onsite storage, CFS/NIFS shares, offsite, in the cloud (Amazon S3, Wasabi), and on tape.

Simple Administration
Features enterprise-grade functionality behind a user-friendly web interface for businesses of all industries and sizes.

Competitive Pricing Model
Offers flexible pricing models that let customers pay only for what they need and easily scale up to accommodate their growing infrastructure.

About NAKIVO

NAKIVO is a US-based corporation dedicated to delivering the ultimate backup, ransomware protection, and disaster recovery solution for virtual, physical, cloud, and SaaS environments. As one of the fastest-growing backup and ransomware recovery software vendors in the industry, NAKIVO boasts 24 consecutive quarters of double-digit growth, 5-star online community reviews, 98% customer satisfaction with support, and a network of over 7,000 partners worldwide. Over 22,000 customers in 171 countries trust NAKIVO with protecting their data, including major companies like Coca-Cola, Honda, Siemens, and Cisco.