Vmware Vsphere Lab Manual

Developed By: Waleed Adlan

Lab 1

ESXi Host Deployment

Objective:

Install and configure VMware ESXi 8 hypervisor as a virtual machine (VM) inside VMware Workstation Pro.

Lab Environment:

Host OS: Your Windows PC/laptop running VMware Workstation Pro (v16 or v17 recommended).

Guest OS (VM): VMware vSphere **ESXi 8.x** ISO.

Resources (recommended):

RAM: 8 GB minimum (12–16 GB ideal)

CPU: 4 vCPUs

Disk: 40 GB or more

Network Adapter: Bridged or Host-only (depending on your lab topology)

ISO: `VMware-VMvisor-Installer-8.x.x-xxxxxx.x86_64.iso`

Task 1: Create a new virtual machine on Vmware Workstation Pro

Open **VMware Workstation Pro** → click **Create a New Virtual Machine**.

Select **Typical (recommended)** → click **Next**.

Choose **Installer disc image file (ISO)** and **browse to your ESXi 8 ISO**.

Click **Next**.

Task 2: Specify the name and the location

Name: `ESXi8-Host01`

Location: Select or create a folder to store the VM files (e.g., `D:\VM-Labs\ESXi8`).

Click **Next**.

Task 3: Implement Disk Configuration

Maximum disk size: `40 GB (60 GB Recommended)`

Select **Store virtual disk as a single file.**

Click **Next**.

Task 4: Customize Hardware before finishing vm creation

Before finishing:

Memory: 8 GB minimum (if your system allows, give 12 GB), we will use 24 GB as we will host vcenter later.

Processors: 4 cores.

Network Adapter:

Choose "NAT" if you want to get ip address from vmware workstation Pro

Choose "Bridged" (so ESXi gets an IP from your LAN for browser access).

Or **Host-only** if you want to keep it isolated.

Add **another NIC** if you plan to practice management and VM networks separately.

We used "**Bridged Mode"** But ensure the physical network adapter of the host machine is physically connected to a switch or should be up.

Confirm the **ISO** is connected under the CD/DVD section.

Click **Close** → then **Finish**.

Task 5: Power On the VM

Power on the virtual machine.

It will boot from the **ESXi 8 ISO**.

Wait for the **"Welcome to the VMware ESXi installation"** screen.

Task 6: Follow ESXI Installation Wizard

Press **Enter** to start installation.

Accept the EULA → **F11**.

Select your 60 GB virtual disk → **Enter**.

Keyboard layout: **US Default** → **Enter**.

When prompted for **root password**, enter:

VMware#123

Confirm and press **Enter**.

Press **F11** to begin installation.

Installation takes about 3–5 minutes.

Task 7: Follow the Reboot Process

When installation completes, you'll see **"Installation Complete"**.

Press **Enter** to reboot.

The VM will restart and load the **ESXi Direct Console User Interface (DCUI)** (yellow-gray screen).

Task 8: Set Management Network IP

On the **DCUI**, press **F2** (Customize System / View Logs).

Log in as:

Username: root

Password: VMware#123

Go to:

Configure Management Network → IPv4 Configuration → Set Static

- Enter:

  IP Address: `192.168.1.100`

  Subnet Mask: `255.255.255.0`

  Default Gateway: `192.168.1.1`

Go to:

Configure Management Network → DNS Configuration → Use the following DNS server addresses.

- Enter:

  Primary DNS Server: `192.168.1.16`

  Hostname: `esxi01`

Press **Enter** → then **Esc** → confirm to **Restart Management Network**.

Task 9: Verify Web Access

On your host (Windows PC), open a browser and go to:

https://192.168.1.100

Accept the SSL warning.

Login with:

Username: root

Password: VMware#123

You should see the **ESXi Host Client Dashboard**:

Summary tab (Host name, CPU, Memory, Uptime)

Storage tab (Shows datastore1)

Networking tab (vmnic0 and management network)

Lab 2

Configure Datastores and Create a New Virtual Machine

Objective:

After completing this lab, students will be able to:

Add and verify a **local datastore** on an ESXi 8 host.

Create a **new virtual machine (VM)** and install an operating system (Windows/Linux).

Verify VM configuration and datastore usage from the ESXi web client.

Prerequisites

ESXi 8 host installed and accessible via browser (`https://<ESXi-IP>`).

You can log in as:

`root / VMware#123`

An ISO image for a guest OS (e.g., `ubuntu-22.04.iso` or `Windows Server 2022.iso`) uploaded to your PC.

Task 1:Access the ESXi Web Client

Open a browser → https://192.168.1.100

Login with:
`Username: rootPassword: VMware#123`

You will see the **ESXi Host Client Dashboard**.

Task2: Add the Local Datastore

Power off the ESXi VM.

In **VMware Workstation**:

Go to **VM → Settings → Add → Hard Disk → SCSI**

Choose **Create a new virtual disk**

Size: **100 GB** (or whatever fits your lab)

Finish → keep it as a new VMDK file

Repeat the above process for Size of **600 GB**. (It will be used for a second datastore for the next step).

Power on the ESXi VM.

In the **ESXi web UI**:

Go to **Storage → New Datastore → Create new VMFS datastore**

Select the **new 100 GB disk**

Name it, e.g. `datastore1`

Format it as **VMFS6**

Done

Now you'll see `datastore1` under **Storage**.

Task3: Create a New Datastore

If you want to simulate multiple datastores:

In the **Storage** section → click **New datastore**.

Choose:

Create new VMFS datastore

Name: `datastore2`

Select an available disk (if you added a second virtual disk to the ESXi VM).

Click **Next → Finish**.

Now ESXi shows two datastores: datastore1 and datastore2.

Task4: Upload an ISO Image

Click **datastore1 → Datastore browser → Upload**.

Choose the OS ISO (`ubuntu.iso`).

Upload progress appears at the bottom right.

Confirm the ISO is visible in the datastore browser.

This simulates your "installation media library."

Task5: Create a New Virtual Machine

From the main dashboard → click **Virtual Machines → Create/Register VM**.

Choose **Create a new virtual machine** → click **Next**.

Enter:

Name: `VM-Linux01`

Compatibility: ESXi 8.0 and later

Guest OS Family: Linux

Guest OS Version: Ubuntu Linux (64-bit)

Click **Next**.

Task6: Select Storage

Choose **datastore1**.

Click **Next**.

Task7: Configure Virtual Hardware

CPU: 2

Memory: 2048 MB (2 GB)

Hard disk: 20 GB

CD/DVD Drive:

Select **Datastore ISO file**

Browse → choose your uploaded ISO (e.g., `ubuntu.iso`)

Check **Connect at power on**

Network Adapter: VM Network

Leave others as default.

Click **Next → Finish.**

Task 8. Power On and Install OS

Select your VM → click **Power on.**

Click **Console → Open in new tab.**

You'll see the OS installation screen.

Follow the installation wizard (e.g., Ubuntu setup or Windows Server setup).

This demonstrates that ESXi can host guest VMs just like a bare-metal hypervisor.

Task 9. Verify Datastore Usage

After installation:

Go to **Storage → datastore1**.

Notice that the **used space has increased**, reflecting the new VM's files.

You can also explore the **Datastore browser** to see the VM folder (e.g., `VM-Linux01`).

Lab 3

Deploy and Configure VMware vCenter Server Appliance (VCSA) 8.0

Objective:

In this lab, students will:

Deploy **VMware vCenter Server Appliance (VCSA) 8.0** on an existing **ESXi 8 host**.

Complete both **Stage 1 (Deployment)** and **Stage 2 (Configuration)** of the vCenter installation wizard.

Access and verify the **vCenter web interface (vSphere Client)**.

Lab Environment:

Component	Specification
DNS Server on Windows Server	Eg.Windows Server 2016 or above
Host	VMware Workstation running ESXi 8 VM
Guest (vCenter)	VMware vCenter Server Appliance 8.0
ISO File	VMware-VCSA-all-8.x.x-xxxxxxx.iso
ESXi Host IP	192.168.1.100
vCenter VM Name	vCenter-Appliance
vCenter IP	192.168.1.110
Root Password	VMware#123

Hardware Requirement (for Workstation)

Minimum: 16 GB RAM, 6 CPU cores.

vCenter needs about **8–10 GB RAM** alone.

Preliminary Task : Add windows Server VM to Vmware Workstation and Enable DNS Server

ADD windows Server 2016 as VM in VMware Workstation with the below specs

```
40 GB disk

2 vCPUs

4 GB RAM

Bridged Network Adapter

Static IP (e.g. 192.168.1.16)
```

DNS Configuration

Step 1 — Log in and Set a Proper Hostname

Log in to your **Windows Server 2016 Desktop**.

Open **Server Manager** (it opens automatically).

In the left menu, click **Local Server**.

On the right, click the **Computer Name** (e.g., `WIN-XXXXXX`).

Click **Change**, and rename it to something like:

```
WIN-DC01
```

Click **OK**, then **Restart Now** when prompted.

Step 2 — Assign a Static IP Address

After reboot, open **Server Manager → Local Server** again.

Next to **Ethernet**, click the blue text (e.g., “IPv4 address assigned by DHCP”).

Right-click your network adapter → **Properties**.

Select **Internet Protocol Version 4 (TCP/IPv4) → Properties**.

Choose **Use the following IP address**, and fill:

Example (if using NAT network in VMware Workstation):

```
IP address: 192.168.1.16

Subnet mask: 255.255.255.0

Default gateway: 192.168.1.1

Preferred DNS server: 127.0.0.1
```

(Using 127.0.0.1 makes it use itself for DNS.)

Click **OK → Close → Close**.

Step 3 — Install the DNS Server Role

In **Server Manager**, click **Manage → Add Roles and Features**.

Click **Next** through the first screens until you reach **Server Roles**.

Check **DNS Server**.

Click **Add Features** when prompted.

Click **Next → Next → Install**.

Wait until it finishes, then click **Close**.

Step 4 — Open the DNS Manager Console

In **Server Manager**, click **Tools → DNS**.

The **DNS Manager** window opens — this is where you'll create your zones and records.

Step 5 — Create a Forward Lookup Zone

In DNS Manager, expand your server name (e.g., `WIN-DC01`).

Right-click **Forward Lookup Zones → New Zone**.

In the wizard:

Primary Zone → Next

Store zone in Active Directory → if AD not installed, this option is greyed out (just continue)

Zone name:

lab.local

Dynamic updates → choose **Do not allow dynamic updates** (simpler for labs).

Click **Finish**.

Step 6 — Add Host (A) Records

Expand **Forward Lookup Zones → lab.local**.

Right-click → **New Host (A or AAAA)**.

Add records for your lab systems:

Name: vcenterIP address: 192.168.1.110

Add host

Repeat for each:

esxi01 → 192.168.1.100

esxi02 → 192.168.1.101

You should now see them listed under the zone.

Step 7 — Create Reverse Lookup Zone

In DNS Manager, right-click **Reverse Lookup Zones → New Zone**.

Zone Type: **Primary Zone** → Next

Network ID:

192.168.1

Dynamic updates → **Do not allow**.

Click **Finish**.

Step 8 — Create PTR Records

If you checked "Create associated PTR record" earlier, they'll exist already.
Otherwise:

Expand **Reverse Lookup Zone** → right-click → **New Pointer (PTR)**.

Add each:

IP: 192.168.1.110 → Host: vcenter.lab.local

192.168.1.100 → esxi01.lab.local

192.168.234.101 → esxi02.lab.local

Step 9 — Test DNS Resolution

From **PowerShell or Command Prompt** on your Windows Server:

```
nslookup vcenter.lab.local

nslookup esxi01.lab.local

nslookup 192.168.1.100
```

You should get proper name ↔ IP resolution both ways.

Step 10 — Use It in Your Lab

Now:

In your **ESXi hosts** network settings, set **Primary DNS** = `192.168.1.16`

Later in **vCenter installer**, when asked for **System Name**, use:

```
vcenter.lab.local
```

and make sure it resolves successfully.

Task 1: Mount the VCSA ISO

On your **Windows host**, right-click the `VMware-VCSA-all-8.x.x.iso` → select **Mount** (or extract using WinRAR/7-Zip).

Open the mounted drive and navigate to:

```
\vcsa-ui-installer\win32\
```

Double-click **installer.exe**.

Task 2: Choose "Install"

The VCSA installer GUI will open.
You'll see 4 options:

Install

Upgrade

Migrate

Restore

Select **Install** to begin a new deployment.

Task 3: Stage 1 — Deploy vCenter Appliance

This stage deploys the VCSA VM to your ESXi host.

3.1. Introduction

Click **Next**.

Accept the license agreement → **Next**.

3.2. Connect to Target ESXi Host

Enter the details of your ESXi host from Lab 1:

Field	Value
ESXi Host	192.168.1.100
Username	root
Password	VMware#123

Click **Next** → accept the SSL certificate warning.

3.3. Set up vCenter VM

Field	Value
VM Name	vCenter-Appliance
root Password	VMware#123

3.4. Deployment Size

Select:

Deployment Size: Tiny

Storage Size: Default (Thin provisioned is fine in lab)

Click **Next**.

"Tiny" supports up to 10 hosts and 100 VMs — perfect for your classroom lab.

3.5 Select Datastore

Choose:

Datastore: `datastore1`

Tick **Enable Thin Disk Mode**
Click **Next**.

3.6. Configure Network Settings

Field	Value
Network	VM Network
IP Allocation	Static
IP Address	192.168.1.110
Subnet Mask	255.255.255.0
Gateway	192.168.1.1
DNS	8.8.8.8
FQDN	vcsa.lab.local

Click **Next → Finish**.

The installer will now **deploy the VCSA VM** onto your ESXi host — this takes about **10–15 minutes**.

3.7. Stage 1 Completion

When you see the message:

> “Deployment of the vCenter Server Appliance was successful.”

Click **Continue** to start **Stage 2** (configuration).

Task 4: Stage 2 — Configure vCenter Appliance

4.1. Appliance Configuration Wizard

Click **Next** to proceed.

4.2. Configure Time and NTP

Setting	Value
Time Synchronization	Synchronize time with the ESXi host
SSH Access	Enable (optional for labs)

Click **Next**.

4.3. Set SSO (Single Sign-On) Domain

Field	Value
SSO Domain Name	vsphere.local
Password	VMware#123
Site Name	LabSite01

Click **Next**.

This creates the identity source for vCenter logins (admin@vsphere.local).

4.4. Customer Experience

Uncheck the “Join CEIP” box → **Next**.

4.5. Summary and Finish

Review all settings → click **Finish**.
Click **OK** when prompted to start configuration.

Stage 2 takes around 10–15 minutes.

The appliance will configure:

vCenter services

vSphere Client

SSO

Databases

Task 5: Access the vCenter Web Interface

Once complete, you'll see:

"vCenter Server setup is complete."

Access using your browser:

https://192.168.1.110

LAUNCH VSPHERE CLIENT → login with:

Username: administrator@vsphere.localPassword: VMware#123

You'll enter the **vSphere Client Dashboard**, where you can:

Add ESXi hosts

Create Datacenters / Clusters

Manage VMs and resources

Lab 4

Add ESXi Host to vCenter and Create Datacenter

Objective:

In this lab, students will:

Connect to the vCenter Server Appliance (VCSA) web interface.

Create a **Datacenter** and **Cluster**.

Add an existing ESXi 8 host to vCenter.

Verify host and resource visibility under vCenter management.

Lab Environment:

Component	Configuration
vCenter IP	192.168.1.110
ESXi Host	192.168.1.100
vCenter Login	administrator@vsphere.local / VMware#123
ESXi Login	root / VMware#123

Task 1:Access vCenter Web Client

Open your browser →

```
https://192.168.1.110
```

Login with:

```
Username: administrator@vsphere.localPassword: VMware#123
```

You'll see the **vSphere Client (vCenter Dashboard)**.

Task 2: Create a New Datacenter

In the **Navigator pane (left)**, right-click your **vCenter Server name** (e.g., *vcenter.lab.local*).

Select:

`New Datacenter`

Enter a name:

`Datacenter-LAB`

Click **OK**.

You'll now see **Datacenter-LAB** under your vCenter hierarchy.

Task 3: Create a Cluster (optional but recommended)

Right-click **Datacenter-LAB** → choose:

`New Cluster`

Enter:

Name: `Cluster01`

Turn OFF DRS and HA for now (we'll enable them in later labs).

Manage all hosts in cluster with a single image: UNCHECKED

Click **Next** → **Finish**.

The cluster is created under **Datacenter-LAB**.

Task 4: Add the ESXi Host to vCenter

Right-click **Cluster01** → select:

`Add Hosts`

In the wizard:

Host name or IP: `192.168.1.100`

User name: `root`

Password: `VMware#123`

Click **Next.**

You'll see a **security certificate warning** — **Select** the Certificate, click **OK** to accept and continue.

Review the **host summary** → click **Next**.

Click → **Finish**

The process will take 2–3 minutes. You'll see:

```
Adding host to inventory...
Connected successfully.
```

Task 5: Verify Host Addition

Expand **Datacenter-LAB → Cluster01**.

You should now see your ESXi host (`192.168.1.100`).

Click on it to view:

Summary tab → CPU, memory, and uptime.

Networking tab → vmnic interfaces and vSwitches.

Storage tab → datastore1.

This confirms the host is successfully managed by vCenter.

Task 6: View Inventory Objects

In the **vSphere Client**, click **Hosts and Clusters** view.

Under **Datacenter-LAB**, confirm:

1 Cluster (Cluster01)

1 ESXi Host (192.168.1.100)

1 Datastore (datastore1)

1 Network (VM Network)

You can now manage VMs, networking, and storage centrally.

Lab 5

Creating and Managing vSwitches and Port Groups via vCenter

Objective:

Learn how to create and manage **standard virtual switches (vSwitches)** and **port groups** in vCenter, and understand how they connect virtual machines (VMs) to the physical and virtual network.

Lab Environment:

vCenter Server 8.0 (IP: `192.168.1.110`)

ESXi Host 8.0 (IP: `192.168.1.100`) — already added to vCenter

You have one or more **VMs** available to connect later.

VM Network Adapters: at least one physical NIC (`vmnic0`)

Goal: Create new vSwitches and port groups through the **vSphere Web Client**

Task 1: Create another VM from Vcenter

Open a browser → `https://192.168.1.110`

Log in as:

Username: `administrator@vsphere.local`

Password: `VMware#123`

Click **Menu → Hosts and Clusters** to enter the main inventory view.

In the left navigation pane:

Expand your hierarchy:

Datacenter-LAB → Cluster01 → 192.168.1.100

Right-click the ESXi host (or the cluster) →

Select **New Virtual Machine → New Virtual Machine**.

Choose:

Create a new virtual machine

Click **Next**.

Name: example → `WinServer-LAB` or `Ubuntu-Test`

Folder: leave default (Datacenter-LAB)

Click Next.

Choose the **ESXi host (192.168.1.100)** as the compute target.

Click **Next**.

Select a datastore, e.g. **datastore1**.

Keep storage format = **Thin Provision** (saves space).
Click **Next**.

Pick the default hardware compatibility (e.g. **ESXi 8.0 and later**).

Click **Next**.

Guest OS Family: Windows / Linux / Other

Version: Choose specific version (e.g. *Windows Server 2019 64-bit* or *Ubuntu 64-bit*).

Click **Next**.

Setting	Recommended for Lab
CPU	2 vCPUs
Memory	4 GB
Hard Disk	40 GB (Thin Provisioned)
Network Adapter	Connect to VM Network
CD/DVD Drive	Mount ISO (click “Datastore ISO file”) and select your OS image, then select at Power On

Click **Next**.

Review your VM summary.

Click **Finish**

You’ll see the VM appear under your host in **Inventory**.

Click **Menu → Hosts and Clusters**.

Power on the Virtual Machine and start the Installation of the VM OS

Task 2: View Existing Networking Configuration

Select the Host 192.1.68.1.100, Then Go to the **Configure** tab.

Under **Networking**, select **Virtual Switches**.

You will see the default **vSwitch0**, which contains the **Management Network** port group.

This is where the ESXi management interface resides.

Task 3: Create a New Standard vSwitch

Click **Add Networking** (or **Add Standard Virtual Switch**).

Choose **Virtual Machine Port Group for a Standard Switch → Next**.

Select **New Standard Switch**.

Assign a **physical adapter (vmnic1)** if available, or leave it unassigned for now → **Next**.

Name the switch:

```
vSwitch-LAB
```

Click **Finish**.

Step 4: Create a New Port Group

Select Virtual switches under Configure Tab

Choose Add Networking→ Choose Virtual Machine Port Group for a Standard Switch

Select an existing standard switch → **se**Select vSwitch1

Under **Networking → Port Groups → Add Port Group**.

Enter:

Network label: `VM-Network-LAB`

VLAN ID: `20` (for example, if you want to segment traffic)

Click **Finish**.

Explanation:

A *port group* defines how VMs connect to a specific VLAN or virtual switch.

Each port group can represent a different subnet or network segment.

Task 5: Assign a VM to the New Port Group

From Left Select the Host 192.168.1.11, Then Go to **Menu → VMs**

Right-click a virtual machine → **Edit Settings**.

Under **Network Adapter 1**, change the **Network label** to `VM-Network-LAB`.

Click **OK**.

Now your VM's virtual NIC is connected to the new port group on `vSwitch-LAB`.

Task 6: Validate Connectivity

Power on two VMs connected to the same port group.

Assign IPs in the same subnet (e.g., `192.168.1.11` and `192.168.1.12`).

Test connectivity using **ping** between them.

If successful — the vSwitch and port group are working correctly.

Task 7 (Optional): Add Another Uplink

If you have more physical NICs on your ESXi host:

Go back to **vSwitch-LAB → Edit Settings**.

Under **Uplinks**, click **Add Uplink** → select another `vmnic`.

This adds redundancy and bandwidth for the virtual switch.

Lab 6

Creating and Managing Distributed vSwitches (vDS) via vCenter

Objective:

Learn how to create, configure, and manage a vSphere Distributed Switch (vDS) using vCenter Server.
You'll understand how vDS simplifies centralized network management across multiple ESXi hosts.

Lab Environment:

vCenter Server 8.0 → IP: `192.168.1.110`

ESXi Host 8.0 → IP: `192.168.1.100`

Network Adapter: `vmnic1` (available for vDS uplink)

Goal: Create a vDS, add the ESXi host, and configure distributed port groups.

Preliminary Task:

Add a second network adapter to the first Esxi host

Add a Second ESXi host to the scenario with two network adapters and two Hard Disks

(60 GB,100 GB)

Add the ESXi to the Vcenter

Add a Linux VM to the second ESXi

Task 1: Log in to vCenter

Open a browser and go to `https://192.168.1.110`.

Log in with:

Username: `administrator@vsphere.local`

Password: `VMware#123`

Go to **Menu → Networking**.

Task 2: Create a New Distributed Switch

In the left pane, right-click your **Datacenter(Datacenter-LAB) → Networks → Distributed Switch → Actions → Distributed Switch → New Distributed Switch**.

Enter a name:

`DSwitch-LAB`

Choose **vDS Version** → select the latest (e.g., 7.0.0).

Click **Next**.

Task 3: Configure vDS Settings

Set:

Number of Uplinks: `2`

Create a Default Port Group: (check)

Name it `DVPortGroup-VMs`

Click **Next → Finish**.

1.

You now have a new **Distributed Switch** with one port group.

Task 4: Add ESXi Host to the Distributed Switch

Right-click **DSwitch-LAB → Add and Manage Hosts**.

Choose **Add Hosts → Next**.

Select your ESXi host (`192.168.1.100,192.168.1.101`) → **Next**.

Under **Manage physical adapters**, click **Assign uplink**.

Assign **vmnic1 → Uplink1**.

For **Network Adapters**, leave VMkernel and VM Network unselected for now → **Next** → **Finish**.

Task 5: Verify the vDS Uplink Connection

Select **DSwitch-LAB → Configure → Settings → Topology**.

You should see:

`vmnic1` attached to `Uplink1` on the ESXi host.

`DVPortGroup-VMs` available for virtual machine connections.

Task 6: Create Additional Distributed Port Groups

Right-click **DSwitch-LAB → Actions → Distributed Port Group → New Distributed Port Group**.

Enter name:

`DVPortGroup-Production`

Set **VLAN Type: VLAN**

VLAN ID: `30` (for example).

Click **Finish**.

Repeat to create another one, e.g.:

`DVPortGroup-TestingVLAN ID: 40`

Task 7: Connect a VM to the Distributed Port Group

Go to one of the VMs in Host1

Go to **Menu → VMs and Templates**.

Right-click a VM → **Edit Settings**.

Under **Network Adapter 1**, select:

Network: `DVPortGroup-Production`.

Click **OK**.

Repeat the same process for a VM in Host02

Make sure the two VM are assigned IP address in the same range

Task 8: Validate Network Connectivity

Power on two VMs connected to the same distributed port group.

Assign IPs in the same subnet (e.g., `192.168.30.10` and `192.168.30.11`).

Test **ping** connectivity.
Successful communication confirms the distributed switch is functioning.

Lab 7

Configuring VMkernel Adapters for vMotion and Management on a Distributed Switch (vDS)

Objective:

Learn how to create and configure VMkernel adapters (vmk interfaces) on a vSphere Distributed Switch (vDS) to handle vMotion and Management traffic.
This lab demonstrates how ESXi hosts use separate logical interfaces for different types of network services.

Lab Environment:

vCenter Server 8.0 → IP: `192.168.1.110`

ESXi Host 8.0 → IP: `192.168.1.100`

vDS: `DSwitch-LAB` (from Lab 6)

Ubuntu Server to work as separate NFS Server for Shared Storage

Available physical NIC on the ESXi host (vmnic1 or vmnic2)

Distributed Port Groups:

`DVPortGroup-vMotion` (VLAN 50)

Task 1: Log in to vCenter

Open browser → `https://192.168.1.110`

Login with:

Username: `administrator@vsphere.local`

Password: `VMware#123`

Go to **Menu → Networking → DSwitch-LAB**

Task 2: Create Distributed Port Groups for vMotion

Right-click **DSwitch-LAB → Distributed Port Group → New Distributed Port Group**.

Name: `DVPortGroup-vMotion`

VLAN ID: `50`

Click **Finish**

Now you have two new port groups for different network services.

Task 4: Create VMkernel Adapter for vMotion

On the same host → **Configure → Networking → VMkernel adapters → Add Networking**.

Choose:

Select an existing an existing network: `DVPortGroup-vMotion`

Enable only:

vMotion traffic

Assign IP:

IP address: `192.168.50.10`

Subnet mask: `255.255.255.0`

Click **Finish**.

You've created a dedicated VMkernel for vMotion.

Task 5: Verify the VMkernel Adapters

In **vCenter**, go to:

Host → Configure → Networking → VMkernel adapters

You should now see two adapters:

VMkernel Adapter	Port Group	Enabled Services	IP Address
vmk0	DVPortGroup-Management	Management	192.168.1.100
vmk1	DVPortGroup-vMotion	vMotion	192.168.50.10

Task 6: Shared Folder Configuration on Linux as NFS Server

Assign IP Address to the Linux VM on the Same Subnet (e.g 192.168.1.32)

Steps to Configure NFS on Linux:

Sudo apt update

Sudo apt install nfs-kernel-server -y

Sudo mkdir -p /nfsdata

Sudo chmod 777 /nfsdata

Sudo nano /etc/exports

#At the bottom line Type#

/nfsdata *(rw,sync,no_subtree_check,no_root_squash)

 # Save and exit (Ctrl+O, Enter, Ctrl+X)

Vverification of NFS Configuration

sudo exportfs -rav

sudo systemctl restart nfs-kernel-server

showmount -e

#You should see#

Export List for ubuntu-nfs:

/nfsdata *

Task 7: Mount The shared NFS Folder:

Right Click on Host -> Storage -> New Datastore -> Choose NFS -> Select NFS3 -> Fill the Data:

Name: Shared-NFS

Server: 192.168.1.32

Folder: /nfsdata

Task 8: Move VM to the Shared Datastore

Power off the VM

Right Click -> Migrate -> Change Storage Only

Select Shared—NFS

Finish -> Wait for the migration

Power on VM

Task 9: (Optional) Add vMotion to Another Host for Testing

If you have **two ESXi hosts**:

Repeat the above steps for the second host using:

Management IP: 192.168.1.11

vMotion IP: 192.168.50.11
Then test vMotion between them.

-

Task 10: Test vMotion (Optional but Recommended)

Go to **Menu → VMs**.

Right-click a powered-on VM → **Migrate**.

Choose **Change compute and storage resource** → select the other host.

vMotion should start and complete successfully without downtime.

Success means the vMotion VMkernel interface is functioning properly.

Lab 8

Configuring iSCSI Storage and Connecting ESXi to a Shared Datastore via vDS

Objective:

Learn how to **configure iSCSI storage networking** using a **vSphere Distributed Switch (vDS)**, connect the **ESXi host** to a **shared iSCSI storage target**, and create a **shared datastore** for VM deployment.

Lab Environment:

vCenter Server 8.0 → IP: `192.168.1.110`

ESXi Host → IP: `192.168.1.100`

vDS: `DSwitch-LAB` (created in Lab 6)

iSCSI Storage VM or Appliance: `192.168.60.100`

Dedicated VLAN for iSCSI traffic: VLAN 60

vCenter and ESXi host connected and operational

vDS is already created with at least two uplinks (vmnic1, vmnic2)

A working iSCSI target (FreeNAS, TrueNAS, StarWind, or another ESXi)

reachable via the iSCSI VLAN

You know the target IP address and IQN

Goal: Configure two VMkernel adapters for iSCSI multipathing and mount a shared datastore.

Task 1: Create Distributed Port Groups for iSCSI Traffic

In vCenter, go to **Menu → Networking → DSwitch-LAB**.

Right-click → **Distributed Port Group → New Distributed Port Group**.

Name: `DVPortGroup-iSCSI1`

VLAN ID: `60`

Click **Finish**

Repeat for the second one:

Name: `DVPortGroup-iSCSI2`

VLAN ID: `60`

These will be used for iSCSI multipathing.

Task 2: Add VMkernel Adapters for iSCSI

Go to **Menu → Hosts and Clusters** → select your **ESXi host**.

Configure → Networking → VMkernel adapters → Add Networking.

Choose:

Select existing distributed switch: `DSwitch-LAB`

Port Group: `DVPortGroup-iSCSI1` → **Next**

Enable:

vMotion: (uncheck)

Provisioning / Fault Tolerance / Management: (uncheck)

Assign static IP:

IP: `192.168.60.10`

Subnet: `255.255.255.0`

Click **Finish**

Repeat for the second one:

Port Group: `DVPortGroup-iSCSI2`

IP: `192.168.60.11`

You now have two VMkernel interfaces for iSCSI multipathing.

Task 3: Verify VMkernel Interfaces

Go to **Configure → Networking → VMkernel adapters.**

Ensure you have:

Adapter	Port Group	IP Address	Purpose
vmk2	DVPortGroup-iSCSI1	192.168.60.10	iSCSI path 1
vmk3	DVPortGroup-iSCSI2	192.168.60.11	iSCSI path 2

Task 4: Enable and Configure the iSCSI Software Adapter

Still under **Configure → Storage → Storage Adapters**.

Click **Add Software Adapter → iSCSI Adapter → OK.**

Select the newly created **iSCSI Software Adapter** (e.g., `vmhba65`).

Click **Enabled → Yes**.

Task 5: Bind VMkernel Adapters to iSCSI Adapter

Select the **iSCSI Software Adapter → Network Port Binding → Add.**

Add both:

`vmk2` (DVPortGroup-iSCSI1)

`vmk3` (DVPortGroup-iSCSI2)

Click **OK.**

This step enables multipathing for iSCSI traffic.

Task 6: Add iSCSI Target

Under **iSCSI Adapter → Dynamic Discovery → Add.**

Enter the iSCSI target IP (e.g., `192.168.60.100`).

Click **OK**, then **Rescan Adapter**.

You should see the **Target IQN** and available **LUNs** appear.

You should do the same for the second host as well

Task 7: Create a Datastore

First of All, Add extra Disk 300 GB to become as a shared storage

Go to **Menu → Storage**.

Right-click → **New Datastore → VMFS.**

Select:

Name: `iSCSI-Datastore01`

Device: choose the discovered iSCSI LUN

Click **Next → Finish.**

The shared iSCSI datastore is now available for all connected hosts.

Lab 9

Configuring High Availability (HA) and Distributed Resource Scheduler (DRS)

Objective:

Learn how to configure and verify **vSphere HA and DRS** in a cluster managed by vCenter Server.
By completing this lab, students will:

Create and configure a vSphere Cluster

Enable **vSphere HA** to automatically restart VMs upon host failure

Enable **DRS** to balance workloads dynamically across ESXi hosts

Lab Environment:

vCenter Server Appliance 8.0

Two ESXi hosts already added to vCenter:

`esxi01.lab.local` → 192.168.1.100

`esxi02.lab.local` → 192.168.1.101

Shared Datastore: `iSCSI-Datastore01`

Distributed Switch: `DSwitch-LAB`

Test VMs: at least two virtual machines

Both ESXi hosts are connected and visible in vCenter

Shared storage (iSCSI/NFS) is accessible by both hosts

vMotion and management networks are already configured (from previous labs)

Sufficient CPU, RAM, and disk resources available

Task 1: Create a New Cluster

In **vCenter → Menu → Hosts and Clusters**.

Right-click the **Datacenter** object → **New Cluster**.

Enter:

Name: `Cluster-LAB`

Check both:

Turn On vSphere HA

Turn On DRS

Click **OK** to create the cluster.

Task 2: Add Hosts to the Cluster

Right-click **Cluster-LAB → Add Hosts.**

Add `esxi01.lab.local` and `esxi02.lab.local` sequentially.

Confirm host connection and license status.

Once added, both hosts appear under **Cluster-LAB**.

Task 3: Configure Cluster Settings

Select **Cluster-LAB → Configure → vSphere Availability.**

Under **Failures and Responses:**

Host Failure Response: *Restart VMs*

Response for Datastore with PDL: *Power off and restart VMs*

Response for APD: *Restart VMs (Conservative)*

Enable **Admission Control**:

Policy: *Cluster Resource Percentage*

Reserve 25% of CPU and memory for failover capacity.

Click **OK** to apply.

Task 4: Configure DRS Settings

Select **Cluster-LAB → Configure → vSphere DRS.**

Set **Automation Level:**

Fully Automated (vCenter will migrate VMs automatically).

Set **Migration Threshold:**

Level 3 (Moderate – balanced between performance and migration frequency).

Click **OK**.

Task 5: Test DRS Load Balancing

Power on several test VMs on the same host (`esxi01`).

Monitor cluster performance under **Monitor → vSphere DRS.**

After a few minutes, vCenter may automatically migrate VMs to `esxi02` to balance resource load.

You can also trigger manual **DRS Recommendations**:

Click **Run DRS Now** → Accept recommendations.

Observe **vMotion migrations** happening automatically — no downtime.

Task 6: Test HA Failover

SSH into `esxi01.lab.local`.

Simulate host failure by putting it into **Maintenance Mode → Power Off** (or disconnect).

Observe behavior in vCenter:

VMs from `esxi01` will **restart automatically on esxi02**.

Once `esxi01` returns online, HA will rebalance workloads again.

HA verified when VMs restart automatically on surviving host.

Lab 11

Configuring vSphere Fault Tolerance (FT)

Objective:

Learn how to **configure and test vSphere Fault Tolerance (FT)** to achieve *zero downtime* for mission-critical virtual machines.

After this lab, students will:

Understand FT concepts and requirements

Enable Fault Tolerance on a VM

Simulate a host failure to verify seamless failover

Lab Environment:

vCenter Server 8.0 Appliance

Two ESXi Hosts:

`esxi01.lab.local` → 192.168.10.10

`esxi02.lab.local` → 192.168.10.11

Cluster-LAB with:

HA enabled

DRS enabled

Shared Datastore: `iSCSI-Datastore01`

vMotion Network: Configured on VLAN 50

VM for FT: `Critical-VM` (powered ON, 1 vCPU)

Cluster-LAB already configured with HA & DRS

Hosts share the same datastore and vMotion network

Licenses support Fault Tolerance (Enterprise Plus or evaluation)

VM has only 1 vCPU (FT for multi-vCPU requires advanced setup)

Task 1: Preliminary Task

Go to VMkernel adapter on Both Hosts and enable Fault Tolerance -Logging

Task 2: Enable Fault Tolerance on a VM

Right-click `Critical-VM` → **Fault Tolerance → Turn On Fault Tolerance.**

Select:

DataStore: Shared-service

Primary Host: `esxi01.lab.local`

Secondary Host: `esxi02.lab.local`

Click **Next → Finish.**

Wait for synchronization — vCenter will create a **Secondary VM** (shadow copy) on `esxi02`.

Task 3: Observe the Configuration

In the vCenter inventory, expand `Critical-VM` → you'll see:

Primary VM: running on `esxi01`

Secondary VM: automatically created on `esxi02`

Check under **Summary tab → Fault Tolerance Status:**

Protected

Secondary VM running

Task 4: Test Fault Tolerance Failover

Open a **continuous ping** to the VM's IP address.

On `esxi01`, simulate failure:

Right-click Host → Power Off (or disconnect it).

Observe:

The **ping continues without interruption**.

The **Secondary VM** on `esxi02` instantly becomes the **Primary VM**.

No downtime — zero packet loss.

Task 5: Restore Protection

Power on or reconnect `esxi01`.

In vCenter, the FT status will show **Degraded** temporarily.

Right-click the VM → **Fault Tolerance → Turn Off FT**, then **Turn On FT** again.

This recreates the Secondary VM and resynchronizes protection.

Lab 12

Configuring vSphere Lifecycle Manager (vLCM) and Patching ESXi Hosts

Objective:

In this lab, students will learn how to use **vSphere Lifecycle Manager (vLCM)**—the successor to Update Manager—to:

Check ESXi host compliance and available patches

Apply baseline updates and remediate hosts

Understand how vLCM manages host lifecycle operations such as patching, firmware, and driver consistency

Lab Environment:

vCenter Server 8.0 managing the environment

Cluster-LAB containing `esxi01` and `esxi02`

Shared Datastore: `iSCSI-Datastore01`

HA and DRS: Enabled

Internet connectivity (or offline patch repository if no Internet)

Both ESXi hosts are connected and healthy in vCenter

Cluster is already created with HA and DRS enabled

vCenter can access the Internet for VMware patch synchronization (or an offline bundle is available)

Maintenance window available for host reboot during remediation

Task 1: Access vSphere Lifecycle Manager

Login to `https://192.168.1.110` as `administrator@vsphere.local`.

Go to **Menu → Lifecycle Manager**.

Verify that the default VMware patch repositories are synced. If not, click **Actions → Sync Updates** to update the metadata.

Task 2: Attach Baseline or Create Image

vLCM supports two models: Baseline-based (legacy) and Image-based (recommended). This lab uses the Baseline-based approach for simplicity.

Go to **Menu → Hosts and Clusters → Cluster-LAB**.

Select the **Updates** tab.

Under **Attached Baselines**, click **Attach → Predefined → Critical Host Patches**.

Click **Attach → OK**.

Task 4: Scan Cluster for Updates

Still in the **Updates** tab, click **Check Compliance**.

Lifecycle Manager compares current ESXi builds with the latest patch levels.

Review results for each host:

Compliant → Host is up to date

Non-Compliant → Patches required

Task 4: Stage and Remediate Hosts

Click **Remediate All** or select one host.

In the wizard:

Choose the attached **Critical Host Patches** baseline.

Leave maintenance mode options default.

Enable "**Automatically disable FT VMs**" if Fault Tolerance is in use.

Click **Remediate** to begin.

Lifecycle Manager will:

Enter maintenance mode (vMotion VMs away using DRS)

Apply patches and reboot the host

Exit maintenance mode when complete

After both hosts are processed, their status shows **Compliant**.

Task 5: (Advanced) Create a Cluster Image

Select the Cluster → **Updates → Setup Image → New Image**.

Choose an ESXi base version (e.g., 8.0 Update 2).

Add vendor firmware and drivers if supported.

Click **Validate → Remediate All Hosts**.

An image ensures every host runs an identical software stack for firmware and drivers.

Lab 13

Configuring vSphere Alarms, Performance Monitoring, and Resource Management

Objective:

In this lab, you will learn how to configure and test **vSphere Alarms**, use **Performance Monitoring tools** in vCenter, and create **Resource Pools** to control CPU and memory allocation.

By completing this exercise, students will understand:

How to monitor ESXi hosts and VMs using vCenter charts

How to create and customize alarms for key events

How to manage compute resources using resource pools

Lab Environment:

This lab continues from the existing environment:

vCenter Server 8.0

Two ESXi hosts: `esxi01.lab.local` and `esxi02.lab.local`

Cluster: `Cluster-LAB` (with HA & DRS enabled)

Shared datastore: `iSCSI-Datastore01`

A few powered-on virtual machines (e.g., `VM-Test1`, `VM-Test2`)

vCenter and both ESXi hosts are connected and healthy.

DRS and HA already configured and active.

At least one VM is powered on for performance data visualization.

Task 1: Explore Performance Monitoring

Log in to **vCenter Server** → `https://192.168.1.110`.

Go to **Menu → Hosts and Clusters**.

Select any **VM** or **ESXi host**, then go to the **Monitor → Performance** tab.

Review the **Overview** graphs:

- CPU usage
- Memory consumption
- Network I/O
- Disk latency

Click **Advanced → Chart Options** to customize:

- Time range (past day, week, or month)
- Metrics (CPU Ready %, Memory Balloon, Disk Commands/sec)

This helps visualize workload trends and identify bottlenecks.

Task 2: Create a Custom Performance Chart

On a host or VM → **Monitor → Performance → Advanced → Chart Options**.

Add multiple counters (CPU, Memory, Disk).

Choose **Line Chart** or **Stacked Graph** visualization.

Save the chart configuration as a **custom view** for easy access later.

You can now switch between real-time and historical performance data.

Task 3: Create an Alarm for a Host

In **vCenter**, right-click your **Cluster-LAB → New Alarm → Alarm Definition**.

Set:

- **Name:** `High CPU Usage`

Monitor Type: *Hosts*

Event Type: *Monitor for specific conditions or state*

Click **Next → Add Condition.**

Condition: **Host CPU Usage (%)**

Warning threshold: `70%`

Critical threshold: `90%`

Under **Actions**, choose:

Send Email Notification (if SMTP configured)

Or **Trigger vCenter action** → e.g., *Power Off VM* (for demo)

Click **Finish.**

This alarm will now trigger if a host's CPU exceeds defined limits.

Task 4: Create an Alarm for a VM

Right-click a **Virtual Machine → New Alarm → Alarm Definition**.

Name it `VM Memory High`.

Monitor Type: *Virtual Machines*

Condition:

Virtual Machine Memory Usage (%)

Warning: 75%

Critical: 90%

Action: Display a **popup alert** or **email notification**.

Use **Monitor → Alarms → Triggered Alarms** to view when thresholds are crossed.

Task 5: Test an Alarm Trigger (Optional)

Power on multiple VMs and run CPU-intensive workloads (like stress-ng or looping scripts).

Observe:

Alarms tab shows Warning/Critical triggers.

vCenter sends alerts if configured.

You can right-click the alarm and **Acknowledge** once tested.

Task 6: Create a Resource Pool

Select your **Cluster-LAB → New Resource Pool.**

Name: `Training-Pool`.

Configure resource allocation:

CPU Shares: Normal

Memory Shares: Normal

Reservations: Leave 0 (default)

Limits: Unlimited (default)

Click **Finish.**

Drag and drop several VMs into `Training-Pool`.

Resource Pools group and prioritize workloads — ideal for labs or departments.

Task 7: Modify Resource Pool Settings

Right-click `Training-Pool` → **Edit Resource Settings**.

Increase **CPU Shares** to *High* and **Memory Shares** to *High*.

Apply changes and observe **Resource Allocation** under the cluster.

This ensures VMs in `Training-Pool` get more CPU/memory priority when contention occurs.

Lab 14

Configuring Templates, Cloning, and Snapshots

Objective:

In this lab, you will learn how to use **VM templates, cloning, and snapshots** to simplify virtual machine deployment and recovery.

By completing this exercise, students will understand:

The difference between a **template**, a **clone**, and a **snapshot**

How to create and deploy VMs quickly using templates

How to use snapshots to save and revert VM states

Lab Environment:

This lab continues from the same environment used in previous exercises:

vCenter Server Appliance 8.0

Two ESXi hosts: `esxi01.lab.local` and `esxi02.lab.local`

Cluster: `Cluster-LAB` (HA & DRS enabled)

Shared datastore: `iSCSI-Datastore01`

One base virtual machine: `WinServer-Base` (fully installed OS and powered off)

The base VM (`WinServer-Base`) must be powered off.

vCenter and both ESXi hosts must be healthy

Sufficient storage space available on the shared datastore.

Administrator privileges in vCenter.

Task 1: Convert a VM into a Template

In **vCenter**, go to **Menu → VMs and Templates**.

Locate your VM: `WinServer-Base`.

Right-click → **Template → Convert to Template**.

Confirm the conversion.

The VM icon changes to a *template symbol*.
Templates are read-only and cannot be powered on.

Purpose: Templates serve as "golden images" for rapid, consistent VM deployment.

Task 2: Deploy a New VM from Template

Right-click the template (`WinServer-Base`) → **New VM from This Template**.

Name it `WinServer-Lab01`.

Select:

Location: `Cluster-LAB`

Host: any available ESXi

Datastore: `iSCSI-Datastore01`

Network: `VM-Network`

Choose **Customize the operating system** (if you have a customization spec) or skip for now.

Click **Finish**.

A new VM is created and ready to power on.

Tip: Customization specs allow automated hostname, IP, and OS configuration.

Task 3: Clone a Running VM

Go to **Menu → Hosts and Clusters** → select a running VM (e.g., `VM-Test1`).

Right-click → **Clone → Clone to Virtual Machine**.

Enter name: `VM-Test1-Clone`.

Choose the **Cluster-LAB** as the destination.

Select the same datastore and network.

Click **Finish**.

The clone will be created — an identical copy of the original VM.

Cloning can be done from powered-on or powered-off VMs; snapshots are used if powered-on.

Task 4: Create a Snapshot

Select any VM → right-click → **Snapshots → Take Snapshot**.

Name: `Before Update`.

Description: "Snapshot before installing Windows updates."

Ensure **Snapshot the virtual machine's memory** is checked (optional).

Click **OK**.

The snapshot is created and listed under **Snapshots → Manage Snapshots**.

Snapshots capture the VM's disk, memory, and power state at a specific point in time.

Task 5: Revert to a Snapshot

After making changes (like installing updates), return to **Snapshots → Manage Snapshots**.

Select the snapshot `Before Update` → click **Revert to Snapshot**.

Confirm when prompted.

The VM returns exactly to its previous state — as if the updates never occurred.

Task 6: Delete Snapshots

Select the VM → **Snapshots → Manage Snapshots**.

Click **Delete All** to consolidate changes back into the base disk.

Always delete snapshots after testing — leaving them long-term can impact performance and storage usage.

Lab 15

vCenter Server Appliance (VCSA) Backup and Restore

Objective:

In this lab, you will learn how to **back up and restore the vCenter Server Appliance (VCSA)** using the built-in **VAMI (vCenter Appliance Management Interface)**.

By the end of this exercise, students will be able to:

Perform a **manual VCSA configuration backup**

Understand backup file structure and types

Restore the appliance from backup using the **VCSA installer ISO**

Lab Environment:

This lab continues from your current environment setup:

vCenter Server Appliance 8.0 — `192.168.1.110`

Two ESXi hosts — `esxi01.lab.local`, `esxi02.lab.local`

Cluster-LAB — with HA & DRS enabled

Backup repository: Network share or FTP/SFTP server (e.g. `192.168.1.90`)

Administrative access to the vCenter Appliance Management Interface (VAMI).

Accessible backup destination (FTP, FTPS, HTTP, HTTPS, SCP, or SFTP).

Enough storage on the backup target to hold database + configuration (~5–10 GB).

vCenter must be in a healthy state (no active tasks or maintenance).

Task 1: Access the Appliance Management Interface (VAMI)

Open your browser and go to:

`https://192.168.1.110:5480`

Log in using:

Username: `root`

Password: (the VCSA root password you configured during installation)

You are now in the **Appliance Management Console (VAMI)**.

Task 2: Enable SSH on Your Windows Machine (Target Backup)

- **Run PowerShell as Administrator**
  - Click **Start**, type `PowerShell`, right-click → **Run as administrator**.
- **Run this command:**

```
Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0
```

  - Wait for the install to finish.

You should see output like:

```
Path          :
Online        : True
RestartNeeded : False
```

  - **Start the SSH service:**

```
Start-Service sshd
```

And set it to start automatically:

```
Set-Service -Name sshd -StartupType Automatic
```

Enable Firewall on Windows and Antivirus to allow SSH

Task 3: Initiate a Manual Backup

In the left-hand menu, click **Backup**.

Click **Backup Now.**

Choose a **backup location**, for example:

`protocol://192.168.1.90/vcenter-backup`

Example: `sftp://192.168.1.90/vcsa_backup`

Enter authentication credentials for the remote storage.

Optional: Add a **backup description** like *"Pre-maintenance backup – 19 Oct 2025"*.

Choose to include:

Configuration and Inventory

Stats, Events, and Tasks (DB data)

Click **Start.**

Wait until the process completes. You'll see:

Backup Progress: 0–100%

Result: Success

The backup files are now stored on the external repository.

Task 3: Verify Backup Status

Go to **Backup → History.**

Confirm:

The new backup entry appears with **Status: Completed Successfully.**

Note the **Timestamp** and **Backup Size.**

-

Backup history is retained for quick validation and troubleshooting.

Task 4: Test the Backup Files (Optional)

Browse to your remote backup repository.

Confirm files such as:

`manifest.json`

`configFile.cfg`

`stats.tar`

`db.tar`

These files contain the VCSA configuration, database, and inventory.

Task 5: Simulate vCenter Failure (Optional for Demo)

Power off or disconnect the VCSA from vSphere inventory.

Try accessing `https://192.168.1.110` — it should fail.

This demonstrates how backup files are essential for disaster recovery.

Task 6: Restore vCenter from Backup

Mount the **vCenter Server Appliance ISO** on your local machine.

Launch the **installer** and choose:

Restore (instead of Install or Upgrade).

Enter the **backup details**:

Protocol: SFTP

Location: `sftp://192.168.1.90/vcsa_backup`

Credentials: same as backup step

Select a **target ESXi host** and datastore for deployment.

Complete the deployment wizard and power on the appliance.

After deployment, the restored vCenter comes online with all settings, inventory, and permissions intact.