



Information  
Technology  
Institute

# vSphere Project

VMware vSphere Infrastructure  
Implementation

## Team Members:

Islam Yasser  
Fatma Ahmed  
Tasneem Adel



## Supervisor:

Eng. Ekram AbdelWahab



## **Table Of Contents:**

<b>Project Overview</b>	<b>3</b>
<b>Infrastructure Setup</b>	<b>4</b>
<b>Cluster Configuration</b>	<b>5</b>
<b>Networking Configuration</b>	<b>6</b>
<b>Storage Infrastructure</b>	<b>11</b>
<b>Content Library</b>	<b>13</b>
<b>VMware Tools Installation and Verification</b>	<b>14</b>
<b>Virtual Machine Operations &amp; Lifecycle Management</b>	<b>15</b>
• Snapshot	15
• Cloning	15
• Template	16
• Migration	17
<b>vSphere High Availability (vSphere HA)</b>	<b>18</b>
<b>vSphere vSphere Distributed Resource Scheduler (DRS)</b>	<b>20</b>
<b>vSphere Fault Tolerance (vSphere FT)</b>	<b>21</b>
• FT Implementation in real lab environment	21
• Successful FT Implementation in HOL Environment	23
<b>Challenges Faced</b>	<b>25</b>
• VMkernel Network Connectivity Issue	25
• FT Deployment Challenges & Troubleshooting	26
• Linux Kernel Update Breaking VMware Workstation	28

# Project Overview

This documentation details the architecture, deployment, and configuration of a resilient, highly available, and automated VMware vSphere virtualization infrastructure. Designed to simulate an enterprise-grade datacenter environment.

The project focuses on resource optimization, continuous availability, and efficient virtual machine lifecycle management.

## Key Infrastructure Highlights:

- **Centralized Compute Cluster:** The core environment is built upon a cluster of three VMware ESXi hosts, centrally administered and monitored via a vCenter Server appliance.
- **Optimized Network Topology:** Network traffic is physically and logically segregated using Standard Virtual Switches (vSwitches). Management and virtual machine data are isolated on `vSwitch0`, while high-bandwidth storage (NFS) and live-migration (vMotion) traffic are dedicated to `vSwitch1`.
- **Shared Storage Backend:** A **Red Hat Enterprise Linux (RHEL 9.2)** server acts as the storage backend, providing centralized NFS datastores. This shared storage architecture enables cluster-wide features and hosts Content Libraries for standardized, rapid VM provisioning.
- **Automated Workload Balancing:** vSphere Distributed Resource Scheduler (DRS) is configured to ensure continuous, automated load balancing and optimal resource allocation across the cluster.
- **Disaster Recovery:** The infrastructure demonstrates robust fault tolerance and recovery capabilities. vSphere High Availability (HA) is enabled to automatically restart workloads in the event of a physical host failure. Furthermore, vSphere Fault Tolerance (FT) is implemented for critical workloads to ensure continuous availability with zero downtime.
- **Lifecycle Management:** The project covers end-to-end virtual machine administration, including live vMotion migrations, state preservation via snapshots, and the creation of standardized VM templates for scalable deployments.

# Infrastructure Setup

- **vCenter Server:** Configured with the management IP address **10.246.78.104**.
- **ESXi Hosts:** Three ESXi hosts were deployed to form the compute cluster.
  - **ESXi 1:** IP **10.246.78.50**.
  - **ESXi 2:** IP **10.246.78.100**.
  - **ESXi 3:** IP **10.246.78.169**.
- **Network Gateway:** All management interfaces share the default IPv4 gateway of **10.246.78.158** with a subnet mask of **255.255.255.0**.

Configure Management Network	IPv4 Configuration
Network Adapters VLAN (optional)	Manual  IPv4 Address: 10.246.78.50 Subnet Mask: 255.255.255.0 Default Gateway: 10.246.78.158  This host can obtain an IPv4 address and other networking parameters automatically if your network includes a DHCP server. If not, ask your network administrator for the appropriate settings.
Configure Management Network	IPv4 Configuration
Network Adapters VLAN (optional)	Manual  IPv4 Address: 10.246.78.50 Subnet Mask: 255.255.255.0 Default Gateway: 10.246.78.158  This host can obtain an IPv4 address and other networking parameters automatically if your network includes a DHCP server. If not, ask your network administrator for the appropriate settings.
Configure Management Network	IPv4 Configuration
Network Adapters VLAN (optional)	Manual  IPv4 Address: 10.246.78.169 Subnet Mask: 255.255.255.0 Default Gateway: 10.246.78.158  This host can obtain an IPv4 address and other networking parameters automatically if your network includes a DHCP server. If not, ask your network administrator for the appropriate settings.
IP Configuration	IP Address: 10.246.78.104 Subnet Mask: 255.255.255.0 Default Gateway: 10.246.78.158
IPv6 Configuration	
DNS Configuration	
Custom DNS Suffixes	

# Cluster Configuration

A cluster named **Cluster01** was established within the datacenter to pool the resources of the three ESXi hosts.

- Both **vSphere HA** and **vSphere DRS** were successfully enabled on **Cluster01**.

New Cluster | Datacenter X

Name	Cluster01
Location	Datacenter
(i) DRS	<input checked="" type="checkbox"/>
(i) vSphere HA	<input checked="" type="checkbox"/>
vSAN	<input type="checkbox"/>

These services will have default settings - these can be changed later in the Cluster Quickstart workflow.

CANCEL OK

vSphere Client | Menu Administrator@VSPHERE.LOCAL

Search in all environments

Cluster01 | ACTIONS ▾

Summary	Monitor	Configure	Permissions	Hosts	VMs	Datastores	Networks	Updates
---------	---------	-----------	-------------	-------	-----	------------	----------	---------

Services  
vSphere DRS  
vSphere Availability

Configuration  
Quickstart  
General  
Licensing  
VMware EVC  
VM/Host Groups  
VM/Host Rules  
VM Overrides  
Host Options  
Host Profile  
I/O Filters

More  
Alarm Definitions  
Scheduled Tasks

Nice work!

Your cluster is now successfully configured and ready to run workloads. In the future you can add more hosts to your cluster by doing Step 2: Add Hosts again.

1. Cluster basics	<input checked="" type="checkbox"/>
Selected services:	
• vSphere DRS	
• vSphere HA	
2. Add hosts	<input checked="" type="checkbox"/>
Hosts: 3	

# Networking Configuration

Network traffic was logically and physically separated using two Standard Virtual Switches (`vSwitch0` and `vSwitch1`) across all hosts.

## vSwitch0 (Management & VM Traffic)

- **Physical Uplink:** Assigned to physical adapter `vmnic0`.
- **Port Groups:**
  - **Management Network:** Configured on `vmk0` with the respective host management IPs (`10.246.78.50`, `10.246.78.100`, `10.246.78.169`).
  - **VM Network:** Dedicated to virtual machine data traffic.

## vSwitch1 (Storage & vMotion Traffic)

- **Physical Uplink:** Assigned to physical adapter `vmnic1`.
- **Port Groups & VMkernel Adapters:**
  - **NFS\_Storage (vmk2):** Used to connect to the shared storage backend. The IP addresses were set to `192.168.20.x` and subnet `255.255.255.0` (`192.168.20.50`, `192.168.20.100`, `192.168.20.169`). The MTU was set to the default `1500`.
  - **vMotion (vmk1):** Used for live migration of virtual machines. Configured on the `192.168.10.x` subnet `255.255.255.0` (`192.168.10.50`, `192.168.10.100`, `192.168.10.169`). Enabled services on this port included **vMotion** and **Provisioning**. The MTU was set to `1500`.

## **ESXI 10.246.78.50:**

### **vSwitch0:**

**Virtual switches**

- Standard Switch: vSwitch0
- Management Network (VMkernel Ports (1))
- VM Network (Virtual Machines (0))

Physical Adapters: vmnic0 10000 Full

### **vSwitch1:**

**Virtual switches**

- Standard Switch: vSwitch1
- NFS\_Storage (VMkernel Ports (1))
- vMotion (VMkernel Ports (1))

Physical Adapters: vmnic110000 Full

**ESXI 10.246.78.100:**

**vSwitch0:**

10.246.78.100 | ACTIONS ▾

Summary Monitor Configure Permissions VMs Datastores Networks Updates

Virtual switches

Standard Switch: vSwitch0 ADD NETWORKING EDIT MANAGE PHYSICAL ADAPTERS ...

Management Network  
VLAN ID: --  
VMkernel Ports (1)  
vmk0 : 10.246.78.100

VM Network  
VLAN ID: --  
Virtual Machines (0)

Physical Adapters  
vmnic0 10000 Full

**vSwitch1:**

10.246.78.100 | ACTIONS ▾

Summary Monitor Configure Permissions VMs Datastores Networks Updates

Virtual switches

Standard Switch: vSwitch1 ADD NETWORKING EDIT MANAGE PHYSICAL ADAPTERS ...

NFS\_Storage  
VLAN ID: --  
VMkernel Ports (1)  
vmk2 : 192.168.20.100

vMotion  
VLAN ID: --  
VMkernel Ports (1)  
vmk1 : 192.168.10.100

Physical Adapters  
vmnic1 10000 Full

## ESXi 10.246.78.169:

### vSwitch0:

The screenshot shows the vSphere Web Client interface for ESXi 10.246.78.169. The left sidebar is expanded, showing the 'Networking' section with 'Virtual switches' selected. The main pane displays 'Standard Switch: vSwitch0' with two configured networks: 'Management Network' (VLAN ID: --, VMkernel Ports: 1, vmk0 : 10.246.78.169) and 'VM Network' (VLAN ID: --, Virtual Machines: 1). A physical adapter, 'vmnic0 10000 Full', is connected to the Management Network. The 'MANAGE PHYSICAL ADAPTERS' tab is selected.

### vSwitch1:

The screenshot shows the vSphere Web Client interface for ESXi 10.246.78.169. The left sidebar is expanded, showing the 'Networking' section with 'Virtual switches' selected. The main pane displays 'Standard Switch: vSwitch1' with two configured networks: 'NFS\_Storage' (VLAN ID: --, VMkernel Ports: 1, vmk2 : 192.168.20.169) and 'vMotion' (VLAN ID: --, VMkernel Ports: 1, vmk1 : 192.168.10.169). A physical adapter, 'vmnic1 10000 Full', is connected to the NFS\_Storage network. The 'MANAGE PHYSICAL ADAPTERS' tab is selected.

### vMotion:

The two vSwitches have been configured for all ESXI's hosts.

#### vmk1 - Edit Settings

Port properties

IPv4 settings

IPv6 settings

VMkernel port settings

TCP/IP stack	Default
MTU	1500

Available services

Enabled services

vMotion

Provisioning

Fault Tolerance logging

Management

vSphere Replication

vSphere Replication NFC

vSAN

CANCEL OK

### NFS\_Storage:

The two vSwitches have been configured for all ESXI's hosts.

#### vmk2 - Edit Settings

Port properties

IPv4 settings

IPv6 settings

VMkernel port settings

TCP/IP stack	Default
MTU	1500

Available services

Enabled services

vMotion

Provisioning

Fault Tolerance logging

Management

vSphere Replication

vSphere Replication NFC

vSAN

CANCEL OK

# Storage Infrastructure

The environment relies on a centralized NFS server to provide shared storage capabilities necessary for cluster services like HA and DRS.

- **NFS Server Details:** A Red Hat Enterprise Linux (RHEL) 9.2 virtual machine (**NFS\_RHEL**) was deployed as the storage backend. It was assigned the IP address **192.168.20.170**.
- **Connectivity Verification:** Successful ICMP ping tests were executed from the RHEL server to the **vmk2** IP interfaces of all three ESXi hosts, confirming routing and connectivity on the storage network.
- **Datastores Mounted:**
  - **Islam datastore:** A local datastore utilized to host the RHEL 9.2 installation ISO (**rhel-9.2-x86\_64**, sized at approximately 9.3 GB).
  - **NFS\_Shared\_Storage:** The primary shared datastore mounted across the cluster.

## Uploading the RHEL .Iso image into the Datastore

Name	Size	Modified	Type	Path
.sdd.sf		02/03/2026, 1:07:07...	Folder	[Islam datastore].sd...
.vSphere-HA		02/07/2026, 6:10:25...	Folder	[Islam datastore].vS...
NFS_RHEL		02/07/2026, 9:38:2...	Folder	[Islam datastore] NF...
vCenterVM		02/07/2026, 7:22:0...	Folder	[Islam datastore] vC...
rhel-9.2-x86_64...	9,371,072 KB	02/07/2026, 9:21:08...	ISO Image	[Islam datastore] rhel...

## RHEL Installation for NFS Storage:

Guest OS:	Linux 5.14.0-284.11.1.el9_2.x86_64 Red Hat Enterprise Linux 9.2
Compatibility:	ESXi 6.7 and later (VM version 14)
VMware Tools:	Running, version:12325 (Guest Managed) <a href="#">More info</a>
DNS Name:	localhost.localdomain
IP Addresses:	192.168.20.170

## Pinging from NFS server to all ESXI's hosts

```
[islam@localhost ~]$ ping 192.168.20.100
PING 192.168.20.100 (192.168.20.100) 56(84) bytes of data.
64 bytes from 192.168.20.100: icmp_seq=1 ttl=64 time=8.73 ms
64 bytes from 192.168.20.100: icmp_seq=2 ttl=64 time=6.33 ms
64 bytes from 192.168.20.100: icmp_seq=3 ttl=64 time=11.8 ms
64 bytes from 192.168.20.100: icmp_seq=4 ttl=64 time=9.92 ms
^C
--- 192.168.20.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 6.332/9.201/11.819/1.988 ms
[islam@localhost ~]$ ping 192.168.20.169
PING 192.168.20.169 (192.168.20.169) 56(84) bytes of data.
64 bytes from 192.168.20.169: icmp_seq=1 ttl=64 time=0.734 ms
64 bytes from 192.168.20.169: icmp_seq=2 ttl=64 time=0.547 ms
64 bytes from 192.168.20.169: icmp_seq=3 ttl=64 time=0.741 ms
64 bytes from 192.168.20.169: icmp_seq=4 ttl=64 time=0.512 ms
^C
--- 192.168.20.169 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3054ms
rtt min/avg/max/mdev = 0.512/0.633/0.741/0.104 ms
[islam@localhost ~]$ ping 192.168.20.50
PING 192.168.20.50 (192.168.20.50) 56(84) bytes of data.
64 bytes from 192.168.20.50: icmp_seq=1 ttl=64 time=5.40 ms
64 bytes from 192.168.20.50: icmp_seq=2 ttl=64 time=28.6 ms
64 bytes from 192.168.20.50: icmp_seq=3 ttl=64 time=8.75 ms
64 bytes from 192.168.20.50: icmp_seq=4 ttl=64 time=9.58 ms
^C
--- 192.168.20.50 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 5.398/13.090/28.637/9.111 ms
[islam@localhost ~]$
```

## Shared DataCenter:

Name	State	Status	Cluster	Consumed CPU %	Consumed Memory %	HA State
10.246.78.100	Connected	Warning	Cluster01	2%	26%	Connected
10.246.78.169	Connected	Warning	Cluster01	15%	91%	Running (Mi)
10.246.78.50	Connected	Warning	Cluster01	0%	21%	Connected

## Content Library:

- A content library named **Project\_ISOs** was created.
- A Photon OS installation medium (**photon-minimal-5.0...**, sized at 537.48 MB) was uploaded to this library for rapid VM deployment.

## Content Library Creation:

The screenshot shows the vSphere Client interface with the 'Content Libraries' tab selected. A single entry named 'Project\_ISOs' is listed in the main pane. The details for this library are shown in a table below:

Name	Type	Published	Passwor...	Automati...	Templates	Other Lib...	Storage ...	Creation ...	Last Mod...	Last Sync
Project_ISOs	Local	No		No	0	0	0 B	Feb 12, 20...	Feb 12, 20...	

## Uploading Iso image to content library

The screenshot shows the details page for the 'Project\_ISOs' content library. Under the 'Other Types' tab, a table lists the uploaded ISO file:

Name	Type	Stored Lo...	Guest OS	Size	Last Modified Date	Last ...	Content Library	UUID
photon-minimal-5.0-dde71ec57x86_64	iso	Yes		537.48 MB	Feb 14, 2026 5:16 PM		Project_ISOs	urn:vapi:co

## Install Photon OS:

The screenshot shows the summary page for the 'PhotonOS\_VM'. The VM is currently powered on, as indicated by the 'Powered On' status and the green 'Powered On' icon. Key details listed include:

- Guest OS: VMware Photon OS (64-bit)
- Compatibility: ESXi 6.7 and later (VM version 14)
- VMware Tools: Not running, not installed
- DNS Name: (not explicitly shown)
- IP Addresses: (not explicitly shown)
- Host: 10.246.78.50

Actions available include 'Launch Web Console' and 'Launch Remote Console'.

## VMware Tools Installation and Verification

Ensuring VMware Tools (or Open VM Tools) are installed and running is a critical step in VM deployment, as it provides essential drivers and enables vCenter to manage the guest operating system effectively. The project involved checking and verifying these tools across different guest operating systems.

### Red Hat Enterprise Linux 9.2 (NFS Server):

For the basic virtual machine running RHEL 9.2, which serves as the NFS storage backend, an explicit installation and verification process was documented.

- **Installation Attempt:** The package manager was utilized via the command line to attempt the installation using the command `sudo dnf install open-vm-tools`.
- **Service Verification:** To ensure the toolset was actively functioning in the background, the command `sudo systemctl status vmtoolsd` was executed.
- **Service Status:** The output confirmed that the `vmtoolsd.service` (Service for virtual machines hosted on VMware) was loaded, `enabled`, and active (`running`).

```
[islam@localhost yum.repos.d]$ sudo dnf install open-vm-tools
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use subscription-manager to register.

RHEL 9 Base OS                                     5.2 MB/s | 1.7 MB   00:00
RHEL 9 AppStream                                  32 MB/s | 6.3 MB   00:00
Package open-vm-tools-12.1.5-1.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!

[islam@localhost yum.repos.d]$ sudo systemctl status vmtoolsd
● vmtoolsd.service - Service for virtual machines hosted on VMware
  Loaded: loaded (/usr/lib/systemd/system/vmtoolsd.service; enabled; preset: enabled)
  Active: active (running) since Thu 2026-02-12 15:26:06 EET; 32min ago
    Docs: http://github.com/vmware/open-vm-tools
    Main PID: 867 (vmtoolsd)
      Tasks: 4 (limit: 10973)
     Memory: 5.1M
        CPU: 3.055s
       CGroup: /system.slice/vmtoolsd.service
               └─867 /usr/bin/vmtoolsd

Feb 12 15:26:06 localhost systemd[1]: Started Service for virtual machines hosted on VMware.
[islam@localhost yum.repos.d]$
```

### VMware Photon OS Virtual Machines:

- Because Photon OS is optimized for vSphere, VMware Tools are baked into the OS automatically.
- Immediately following the deployment of PhotonOS\_VM, the vSphere Client briefly showed the tools as `Not running`, `not installed` prior to the OS completing its first full boot sequence.
- Shortly after, the status automatically updated to `Running`, without any manual installation commands being executed.

# Virtual Machine Operations & Lifecycle Management

Extensive testing of VM lifecycle and mobility features was conducted using the lightweight **PhotonOS\_VM**.

- **Snapshots:** A snapshot named **Photon\_VM\_Snapshot** was subsequently taken, capturing 1.05 GB of disk usage and preserving the virtual machine's memory state.

Manage Snapshots | PhotonOS\_VM

Name	Photon_VM_Snapshot
Created	02/14/2026, 8:22:20 PM
Disk usage	1.05 GB
Snapshot the virtual machine's memory	Yes
Quiesce guest file system	No

**DELETE ALL** **DELETE** **REVERT TO** **EDIT**

**DONE**

- **Cloning:** The VM was successfully cloned to a new instance named **PhotonOS\_VM\_Clone**.

PhotonOS\_VM\_Clone

Summary Monitor Configure Permissions Datastores Networks Updates

Powered Off

Guest OS: null  
Compatibility:  
VMware Tools: N/A  
More info

DNS Name:  
IP Addresses:  
Host:

Launch Web Console  
Launch Remote Console

Recent Tasks Alarms

Task Name	Target	Status	Initiator	Queued For	Start Time
Fetch Content of a Library	photon-minimal-5.0-dde71...	Completed	vsphere.loc...	132 ms	02/14/2026, 8:40:08 PM
Clone virtual machine	PhotonOS_VM	43% (X)	VSPHERE.L...	12 ms	02/14/2026, 8:38:56 PM

- **Templates:** To standardize future deployments, the VM was cloned into a template named **PhotonOS\_VM\_Template**. This template was stored on **NFS\_Shared\_Storage** utilizing Thin Provisioning.

### PhotonOS\_VM - Clone Virtual Machine To Template

✓ 1 Select a name and folder      Ready to complete  
 ✓ 2 Select a compute resource  
 ✓ 3 Select storage  
**4 Ready to complete**

Click Finish to start creation.

Provisioning type	Clone virtual machine to template
Source virtual machine	PhotonOS_VM
Template name	PhotonOS_VM_Template
Folder	Datacenter
Cluster	Cluster01
Datastore	NFS_Shared_Storage
Disk storage	Thin Provision

CANCEL   BACK   FINISH

The screenshot shows the VMware vSphere interface. On the left is a navigation tree for the Datacenter. In the center, the details for the virtual machine "PhotonOS\_VM" are displayed, including its summary, hardware, and related objects. On the right, there are performance metrics for CPU, memory, and storage usage. At the bottom, a table titled "Recent Tasks" shows the current task: "Clone virtual machine" for target "PhotonOS\_VM", which is 40% complete and started at 02/14/2026, 8:56:55 PM.

Task Name	Target	Status	Initiator	Queued For	Start Time	Completion Time	Server
Clone virtual machine	PhotonOS_VM	40%	VSPHERE...	3 ms	02/14/2026, 8:56:55 PM		10.246.78.104

- **Live Migration (vMotion):** PhotonOS\_VM was successfully live-migrated from host **10.246.78.50** to host **10.246.78.100**. The migration priority was set to "High".

**PhotonOS\_VM** | ACTIONS ▾

**Summary** Monitor Configure Permissions Datastores Networks Updates

**Powered On**

**Guest OS:** VMware Photon OS (64-bit)  
**Compatibility:** ESXi 6.7 and later (VM version 14)  
**VMware Tools:** Running, version:12352 (Guest Managed) [More info](#)  
**DNS Name:** photon-b6279b613d9c  
**IP Addresses:** 10.246.78.137 [View all 2 IP addresses](#)  
**Host:** 10.246.78.50

[Launch Web Console](#) [Launch Remote Console](#)

**PhotonOS\_VM - Migrate**

✓ 1 Select a migration type Ready to complete  
✓ 2 Select a compute resource Verify that the information is correct and click Finish to start the migration.  
✓ 3 Select networks  
✓ 4 Select vMotion priority  
**5 Ready to complete**

Migration Type	Change compute resource. Leave VM on the original storage
Virtual Machine	PhotonOS_VM
Cluster	Cluster01
Host	10.246.78.100
vMotion Priority	High
Networks	No network reassignments

**CANCEL** **BACK** **FINISH**

**PhotonOS\_VM** | ACTIONS ▾

**Summary** Monitor Configure Permissions Datastores Networks Updates

**Powered On**

**Guest OS:** VMware Photon OS (64-bit)  
**Compatibility:** ESXi 6.7 and later (VM version 14)  
**VMware Tools:** Running, version:12352 (Guest Managed) [More info](#)  
**DNS Name:** photon-b6279b613d9c  
**IP Addresses:** 10.246.78.137 [View all 2 IP addresses](#)  
**Host:** 10.246.78.100

[Launch Web Console](#) [Launch Remote Console](#)

# vSphere High Availability (vSphere HA)

During the validation phases, several specific behaviors and simulated failures were tested:

- After Enabled vSphere HA The election finished and there is the master and slaves.

Name	State	Stat...	Cluster	Consumed CPU %	Consumed Memori...	HA State
10.246.78.100	Connected	W...	Cluster01	2%	25%	Connected (Slave)
10.246.78.169	Connected	W...	Cluster01	7%	91%	Running (Master)
10.246.78.50	Connected	W...	Cluster01	0%	21%	Connected (Slave)

- **HA Failover Test and VM Downtime:** To test HA functionality, a crash was simulated on host **10.246.78.100**, which was actively running **PhotonOS\_VM**.

PhotonOS\_VM

Guest OS:	VMware Photon OS (64-bit)
Compatibility:	ESXi 6.7 and later (VM version 14)
VMware Tools:	Running, version:12352 (Guest Managed) <a href="#">More info</a>
DNS Name:	photon-b6279b613d9c
IP Addresses:	10.246.78.137 <a href="#">View all 2 IP addresses</a>
Host:	10.246.78.100

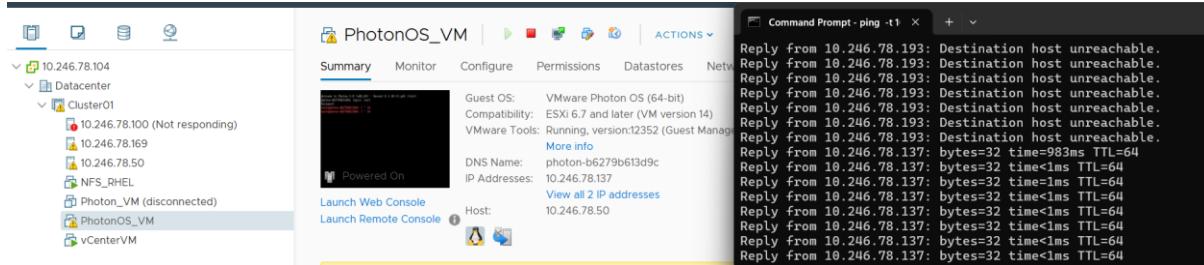
- The host entered a **Not responding** state. During this time, continuous ping requests to the VM dropped, showing **Request timed out** and **Destination host unreachable**.

PhotonOS\_VM

VMware Tools is not installed on this virtual machine.

```
Reply from 10.246.78.137: bytes=32 time=7ms TTL=64
Reply from 10.246.78.137: bytes=32 time=6ms TTL=64
Reply from 10.246.78.137: bytes=32 time=21ms TTL=64
Reply from 10.246.78.137: bytes=32 time=53ms TTL=64
Reply from 10.246.78.137: bytes=32 time=6ms TTL=64
Reply from 10.246.78.137: bytes=32 time=11ms TTL=64
Reply from 10.246.78.137: bytes=32 time=17ms TTL=64
Reply from 10.246.78.137: bytes=32 time=12ms TTL=64
Reply from 10.246.78.137: bytes=32 time=30ms TTL=64
Reply from 10.246.78.137: bytes=32 time=11ms TTL=64
Reply from 10.246.78.137: bytes=32 time=6ms TTL=64
Reply from 10.246.78.137: bytes=32 time=8ms TTL=64
Reply from 10.246.78.137: bytes=32 time=82ms TTL=64
Reply from 10.246.78.137: bytes=32 time=68ms TTL=64
Reply from 10.246.78.137: bytes=32 time=77ms TTL=64
Reply from 10.246.78.137: bytes=32 time=86ms TTL=64
Request timed out.
Request timed out.
Request timed out.
Reply from 10.246.78.193: Destination host unreachable.
Reply from 10.246.78.193: Destination host unreachable.
```

- vSphere HA detected the host failure and automatically restarted **PhotonOS\_VM** on a surviving host (**10.246.78.50**).
- Once the VM rebooted, ping replies stabilized, confirming successful automated disaster recovery.

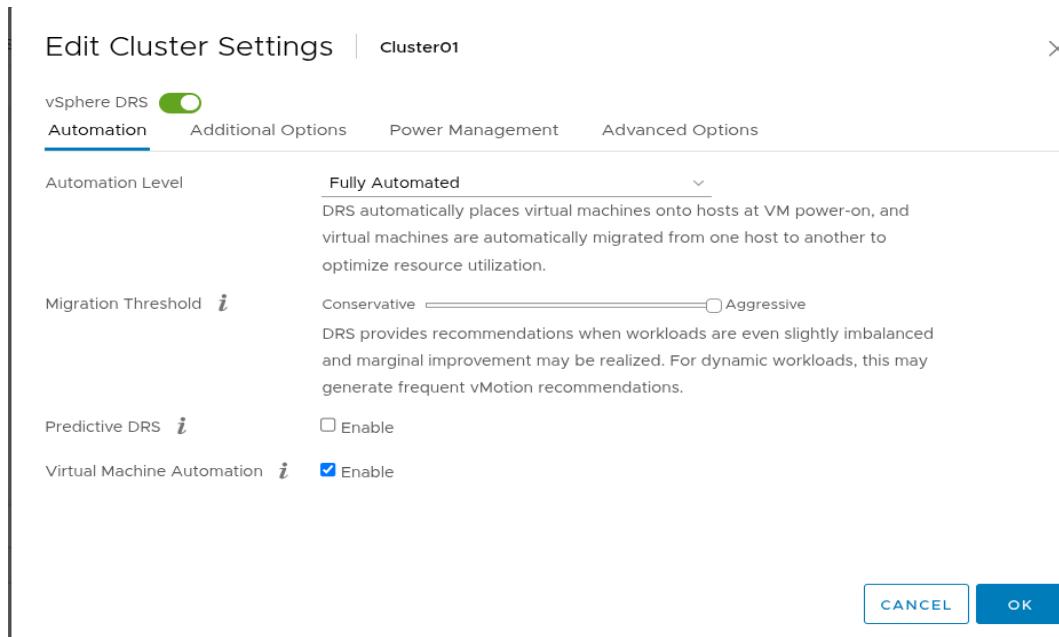


## vSphere Distributed Resource Scheduler (DRS)

vSphere DRS was enabled during the initial configuration of **Cluster01**, alongside vSphere HA, to ensure balanced resource allocation and optimize the performance of the virtual machines across the ESXi hosts.

The cluster settings were specifically tuned to handle workload distribution without requiring manual intervention:

- **Automation Level:** The automation level was set to Fully Automated. Under this setting, DRS handles operations without manual prompts in two key ways:
  - **Initial Placement:** It automatically places virtual machines onto the most optimal hosts when they are powered on.
  - **Load Balancing:** It automatically live-migrates virtual machines from one host to another to continuously optimize resource utilization across the cluster.
- **Migration Threshold:** The migration threshold was configured to be **Aggressive**. With this high-sensitivity setting, DRS provides recommendations when workloads are even slightly imbalanced and marginal improvement may be realized. For dynamic workloads, this aggressive stance may generate frequent vMotion recommendations.
- **Additional Automation Options:**
  - **Virtual Machine Automation:** This setting was explicitly enabled to allow DRS to apply automation at the individual VM level.
  - **Predictive DRS:** This advanced feature was left disabled.



# vSphere Fault Tolerance (vSphere FT)

- **On the real lab environment we were working on:**
  - Zero-downtime protection was configured for **PhotonOS\_VM**. Prior to turning on vSphere FT, the Fault Tolerance logging service was explicitly enabled on the **vmk1** VMkernel adapter on vSwitch1.

vmk1 - Edit Settings

The screenshot shows the 'vmk1 - Edit Settings' dialog. On the left, there's a sidebar with 'Port properties' selected, followed by 'IPv4 settings' and 'IPv6 settings'. The main area is titled 'VMkernel port settings' and shows 'TCP/IP stack' set to 'Default' and 'MTU' set to '1500'. Below this is a section titled 'Available services' with a list of checkboxes. The checked boxes are: 'vMotion', 'Provisioning', and 'Fault Tolerance logging'. The unchecked boxes are: 'Management', 'vSphere Replication', 'vSphere Replication NFC', and 'vSAN'. At the bottom right are 'CANCEL' and 'OK' buttons.

- Immediately after enabling vSphere FT, the **Primary** VM remained on its host while the **Secondary** VM was placed on **10.246.78.106**.

PhotonOS\_VM - Turn On Fault Tolerance

The screenshot shows the 'PhotonOS\_VM - Turn On Fault Tolerance' dialog. On the left, a sidebar lists steps: '1 Select datastores' (green checkmark), '2 Select host' (green checkmark), and '3 Ready to complete' (blue button). The main area has a title 'Ready to complete' with the sub-instruction 'Review your selections and click Finish to turn on fault tolerance on this virtual machine.' Below this is a section titled 'Placement details for the Secondary VM' with the following table:

Host:	10.246.78.106
Configuration File Location:	NFS_Shared_Storage
Tie Breaker File Location:	NFS_Shared_Storage
Hard disk 1 Location:	NFS_Shared_Storage

At the bottom right are 'CANCEL', 'BACK', and 'FINISH' buttons.

The screenshot shows the vSphere Web Client interface. On the left, a navigation tree displays a datacenter named "10.246.78.104" containing a cluster named "Cluster01". Inside Cluster01, several VMs are listed: "10.246.78.100" (warning), "10.246.78.106" (warning), "10.246.78.169" (warning), "10.246.78.50" (warning), "NFS\_RHEL" (green checkmark), "Photon\_VM" (green checkmark), "PhotonOS\_VM (primary)" (selected and highlighted in blue), and "vCenterVM" (green checkmark). On the right, a detailed summary card for "PhotonOS\_VM" is shown. The card includes the following information:

Summary	Monitor	Configure	Permissions	Datastores	Networks	Updates
Guest OS: VMware Photon OS (64-bit)						
Compatibility: ESXi 6.7 and later (VM version 14)						
VMware Tools: Not running, not installed						
<a href="#">More info</a>						
DNS Name:						
IP Addresses:						
Host: 10.246.78.50						
<a href="#">Launch Web Console</a> <a href="#">Launch Remote Console</a>						

Below the summary card, there is a "Recent Tasks" table:

Task Name	Target	Status	Initiator	Queued For	Start Time	Completion Time	Server
Start Fault Tolerance	PhotonOS_VM	100%	System	4 ms	02/14/2026, 10:36:57 PM		10.246.78.104

On the far right, performance metrics are displayed:

- CPU USAGE: 0 Hz
- MEMORY USAGE: 0 B
- STORAGE USAGE: 1.06 GB

- The vSphere client displayed a yellow warning: **Fault Tolerance status: Not protected**. This was verified to be a normal transitional state. The interface noted the status was **Starting** because the powered-on primary virtual machine was actively **Synchronizing** its state to the newly deployed secondary VM.

Fault Tolerance

Fault Tolerance status

**⚠️ Not protected**

Starting

The virtual machine is powered on and has at least one secondary VM that is synchronizing its state with the primary VM.

Secondary VM location	10.246.78.106
Log bandwidth usage	N/A

- Successful Implementation via VMware Hands-on Labs**
  - To conclusively validate our Fault Tolerance deployment methodology and overcome the local physical bottlenecks, we transitioned our execution to the **VMware Hands-on Labs (HOL)** environment.

#### The following steps were successfully implemented:

- Dedicated EVC Cluster Creation:** A new, separate cluster named **EVC-en** was created specifically for Fault Tolerance testing. Enhanced **vMotion Compatibility (EVC)** was successfully enabled on this dedicated cluster to guarantee a uniform CPU baseline.

TinyLinux2

Fault Tolerance

Status: Protected

Metro Cluster status: Disabled

Host Group: Not Applicable

Recent Tasks: Start Fault Tolerance Secondary VM

Task Name	Target	Status	Details	Initiator	Queued For	Start Time
Start Fault Tolerance Secondary VM	TinyLinux2	Completed	Migrating Virtual Machine active state	System	4 ms	02/16/2026, 1:40:04 PM

- **Isolating the Management Plane:** Two ESXi hosts (`esx-01a.vcf.sddc.lab` and `esx-02a.vcf.sddc.lab`) were added to this new cluster. Crucially, these hosts were deliberately selected because they were not hosting the vCenter Server. This successfully bypassed the **vCenter paradox**, allowing the EVC baseline to be applied without disrupting the management plane.
- **Overcoming Admission Control Limits:** After configuring the required FT Logging networks, the allocated RAM for our test virtual machine (`TinyLinux2`) was strategically reduced. This memory tuning ensured the VM's footprint complied with vSphere's strict Admission Control policies within the lab.

Name	Reservation (MB)	Limit (MB)	Type	Shares	Shares Value	% Shares
/vmfs/volumes/5a905bc6-8f09cc8-9e77-00505601dfd/a/vCLS-095509b5-dca1-402 (inaccessible)	0	0	Fixed	0	0	0

- The cluster only had 190 MB of memory available.
- Enabling Fault Tolerance on a 512 MB VM required more memory than the cluster had left.
- Therefore, we decreased the VM's memory from 512 MB to 128 MB. This allowed it to easily fit into the 190 MB limit and power on successfully.

**Successful Execution and Protection:** With the CPU architecture unified and resources optimized, Fault Tolerance was initiated. The vCenter task **Start Fault Tolerance Secondary VM** completed successfully.

**Verification:** The vSphere Client confirmed the synchronization, transitioning the FT Status to Protected. Both the **TinyLinux2 Primary** and **Secondary** virtual machines operated simultaneously under a **Normal** status, successfully demonstrating a fully functional FT configuration.

	Name	State	Status	Provisioned Space	Used Space	Host CPU	Host Mem
☐   ::	<a href="#">TinyLinux2 (secondary)</a>	Powered On	✓ Normal	180.01 MB	80.01 MB	0 Hz	0 B
☐   ::	<a href="#">TinyLinux2 (primary)</a>	Powered On	✓ Normal	100.55 MB	18.09 MB	21 MHz	156 MB

# Challenges Faced

- **VMkernel Network Connectivity Issue**

One of the main challenges faced during the project was related to **VMkernel network connectivity** used for NFS storage.

All ESXi hosts were configured with VMkernel adapters in the same subnet **192.168.20.x**.

Communication between ESXi hosts was successful.

However, the NFS virtual machine was unable to communicate with one specific host **192.168.20.50**.

This caused issues in accessing shared storage and prevented proper datastore functionality.

→ **Root Cause**

After extensive troubleshooting of:

- vSwitch configuration
- Port Groups
- VMkernel adapters
- Physical uplinks

It was found that the issue was not related to the ESXi configuration itself, but rather to the **host operating system environment**.

The ESXI's Host was initially running on a Linux-based system using VMware Workstation, which caused unexpected network behavior and instability in the virtual networking layer.

→ **Solution**

- The host operating system was changed from Linux to Windows
- The environment was rebuilt using VMware Workstation on Windows
- All ESXi hosts and networking configurations were recreated

**After switching to Windows:**

- Network connectivity became stable
- VMkernel adapters were reachable from the NFS VM
- NFS datastore was successfully configured

→ **Lesson Learned**

The issue highlighted that:

- The underlying host OS can significantly impact virtualization performance
- Some networking issues may originate outside the ESXi configuration
- Using a stable and fully supported host environment is critical for lab setups

- **FT Deployment Challenges & Troubleshooting**

Deploying Fault Tolerance in this simulated enterprise environment presented significant hardware and resource challenges, which required strategic workarounds.

- **Challenge 1:**

### **Enhanced vMotion Compatibility (EVC) and Hardware Mismatch**

vSphere FT requires the Primary and Secondary VMs to execute in strict lockstep, which demands identical CPU instruction sets and hardware compatibility between the physical hosts. During the initial FT deployment across the original cluster hosts, an EVC/hardware compatibility error occurred.

- **Solution:** To bypass this hardware mismatch and fulfill FT's strict CPU requirements, a completely new nested ESXi host was provisioned with the IP address **10.246.78.106**. This new host was specifically created on the **same physical PC** hosting the primary VM. By running both the primary and secondary hosts on the exact same underlying physical hardware, the CPU architecture was guaranteed to be identical, resolving the EVC conflict and allowing the FT configuration to proceed.

**The Optimal Enterprise Solution (Enabling VMware EVC):** The best practice to resolve this in a production environment is to enable Enhanced vMotion Compatibility (EVC) on **Cluster01**. EVC establishes a baseline CPU feature set across all hosts in the cluster, masking hardware differences and allowing features like vMotion and FT to function seamlessly across heterogeneous processors.

### **Steps to Enable EVC:**

1. Navigate to **Cluster01** in the vSphere Client.
2. Select the **Configure** tab.
3. Under the **Configuration** menu, select **VMware EVC**.
4. Click **EDIT** and change the EVC state to **Enable** for the respective CPU vendor (Intel or AMD).
5. Select an EVC mode (baseline) that corresponds to the oldest processor architecture present among the ESXi hosts.

**The vCenter Paradox & Our Lab Workaround:** To apply a lower EVC baseline successfully, any running virtual machines utilizing advanced CPU instructions must be powered off. However, because our **vCenter VM** resides inside the cluster it manages, powering it off creates a paradox where we lose the management interface required to enable EVC. Resolving this normally requires complex migrations to hosts outside the cluster.

To bypass this paradox without disrupting the nested environment, we provisioned a completely new nested ESXi host (**10.246.78.106**) on the same physical PC that was hosting the primary VM. Running both the primary and secondary hosts on the exact same underlying physical hardware guaranteed identical CPU architecture out-of-the-box, allowing us to seamlessly proceed with the FT configuration.

---

- **Challenge 2:**

#### **Severe Resource Contention**

Because the environment utilizes nested virtualization running multiple ESXi hosts, a vCenter Server appliance, and an NFS server, the hardware's compute and memory resources were heavily oversubscribed.

- When FT was enabled, it attempted to deploy and power on the Secondary VM on the new host (**10.246.78.106**). However, due to the severe lack of available physical resources, the Secondary VM could not fully instantiate or complete its initial synchronization.

- As a result of this resource exhaustion, the vSphere client displayed a yellow warning: Fault Tolerance status: **Not protected**. The interface remained stuck in the **Starting** state, noting that the VM was attempting to synchronize its state to the newly deployed secondary VM, but could not complete the process due to the hardware bottleneck.

Fault Tolerance status	
<b>⚠️ Not protected</b>	
<b>Starting</b>	
The virtual machine is powered on and has at least one secondary VM that is synchronizing its state with the primary VM.	
Secondary VM location	10.246.78.106
Log bandwidth usage	N/A

- Linux Kernel Update Breaking VMware Workstation**

Another major issue occurred after updating the host operating system. A Linux kernel update caused VMware Workstation to stop working properly. Virtual machines failed to start due to incompatibility between VMware modules and the updated kernel.

→ **Root Cause**

VMware Workstation relies on kernel modules that must be compatible with the running Linux kernel.

After the update, these modules were no longer supported.

→ **Solution**

- Reverted to a previous, compatible Linux kernel version
- Ensured VMware modules could compile correctly
- Avoided automatic kernel updates during the project timeline