Higher Diploma Project Final Report

COHDCN19.1F

Project Title	Upgrade of the Network Design for the ITIT institute
Student Names	Rusiru Munasingha Sakunthala Ellawala Sudeera Seneviratne
Registration No. & Index No.	
Supervisor's Name	
For Office Use Only	

Upgrade of the Network Design for ITIT institute

A dissertation submitted for the
Higher Diploma in Computer Networks

R.T. Munasingha
S.H.Seneviratna
S.P. Ellawala

National Institute of Business Management
School of Computing
2020

Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.
To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Students Names:

1.Rusiru Munasingha Signature: …….

2.Sakunthala Ellawala Signature: …….

3.Sudeera Seneviratne Signature: …….

Date: ……..

This is to certify that this project is based on the work of …… under my supervision. The report has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name:

_____

Signature: Date:

ITIT institution Upgraded network design

NIBM

HDCN19.1| 15/09/2020

Abstract

In the Vastly increasing World of technology, Education plays a fundamental role in preparing the students to participate in the arising opportunities through innovative technology provided within the learning system. Furthermore, ITIT University of higher Education provides the highest standards of information sharing through its robust network infrastructure to deliver world class learning experience to its student body and faculty alike.

Through the upgradation of the previous Network infrastructure the proposed upgradations includes newly introduced facilities and avant-garde technologies to ensure the learning curve of the university inhabitants are an always enriching experience that delivers insights towards ever changing the industry platforms.

Furthermore, in terms of availability of information, the proposed upgrade for the network infrastructure contains major technologies that assures redundancy, accessibility and infrastructure flexibility for growth in the years to come.

Moreover, this proposed design of infrastructure focuses on the latest framework of high availability, integrity and security where the internal eco systems are protected from external threats.

In conclusion, in this project scope the integrations of upcoming technologies such as azure cloud, virtualization technologies and Linux technologies will be used to provide a seamless learning experience to the higher education institute of ITIT

Acknowledgements

This Project would not have been possible for all the time and effort put in by our course Director Mr. Milan Maduranga who guided us and taught us all the necessary things to make this project a successful reality.

In addition, special thanks goes out to all our HND lecturers for instilling us with all the knowledge and practical expertise to take this project a one-step further in terms of innovative means and create truly unique beneficial project content

Moreover, much thanks goes out to all our seniors, who supported us by sharing their invaluable advice and experience on the subject matters, and to our batch mates, for all the time-spent during brainstorming sessions, coming up with different troubleshooting methods.

Furthermore, much gratitude goes out to the community forums that aided us in tackling many technical issues and with trouble shooting processes with expert guidance and insightful analysis of the root problems and system errors.

In addition, to our parents, for giving us their unstinted support and encouragement, to help us pursue and push this project into what it is today.

And Finally our thanks goes out to google, for all the information, online designing tools, and accurate data, for without it this project won't be as half cutting edge and accurate as it is now

Contents

Figures and tables

Introduction

With the increasing demand of information and communication that is being a daily necessity for day to day transactions, the delivery of both aspects are of utmost importance within the environment of education .
There in, in such situations providing fast and always available resources that are accessible to the student body and staff is one of the most important factor in an education facility providing the innovative ways to learn and vocationally develop .

Here in , the ITIT is an institute that consists with roughly 3000 students and with around 100 lectures with more than 100 other employees, have a supporting network infrastructure that has the ability to provide all its uses with a highly availability , Infrastructural flexibility , self-healing, layered protection security to the internal system for all daily activities
Furthermore, the proposed upgrade to the infrastructure will prove to provide a more efficient way to connect the overall ecosystems of the campus

The network is heavily redundant due to the usage of duplicated devices in both layers and also due to the availability of many redundant links through cloud and failover technologies to provide the users with a reliable network connection and access to the required systems
The proposed upgrade network is a collapsed core hierarchical model.

- The network is divided into subnetworks though VLANs
  - IT centre
  - Student WIFI
  - Lab
  - Guest WIFI
  - Academic staff
  - CCTV Surveillance
  - Exams
  - Finance
  - AP Management
  - HR
  - Internal Servers
  - Device Management
  - ITS Dept.
  - Marketing
  - Stuff
  - Storage

The specific upgrades to the network infrastructure are as follows ;

- Virtualization of servers
- Addition of failover servers and DMZ servers

- Firewall redundancy
- Network scripting

Up-graded Network Topology

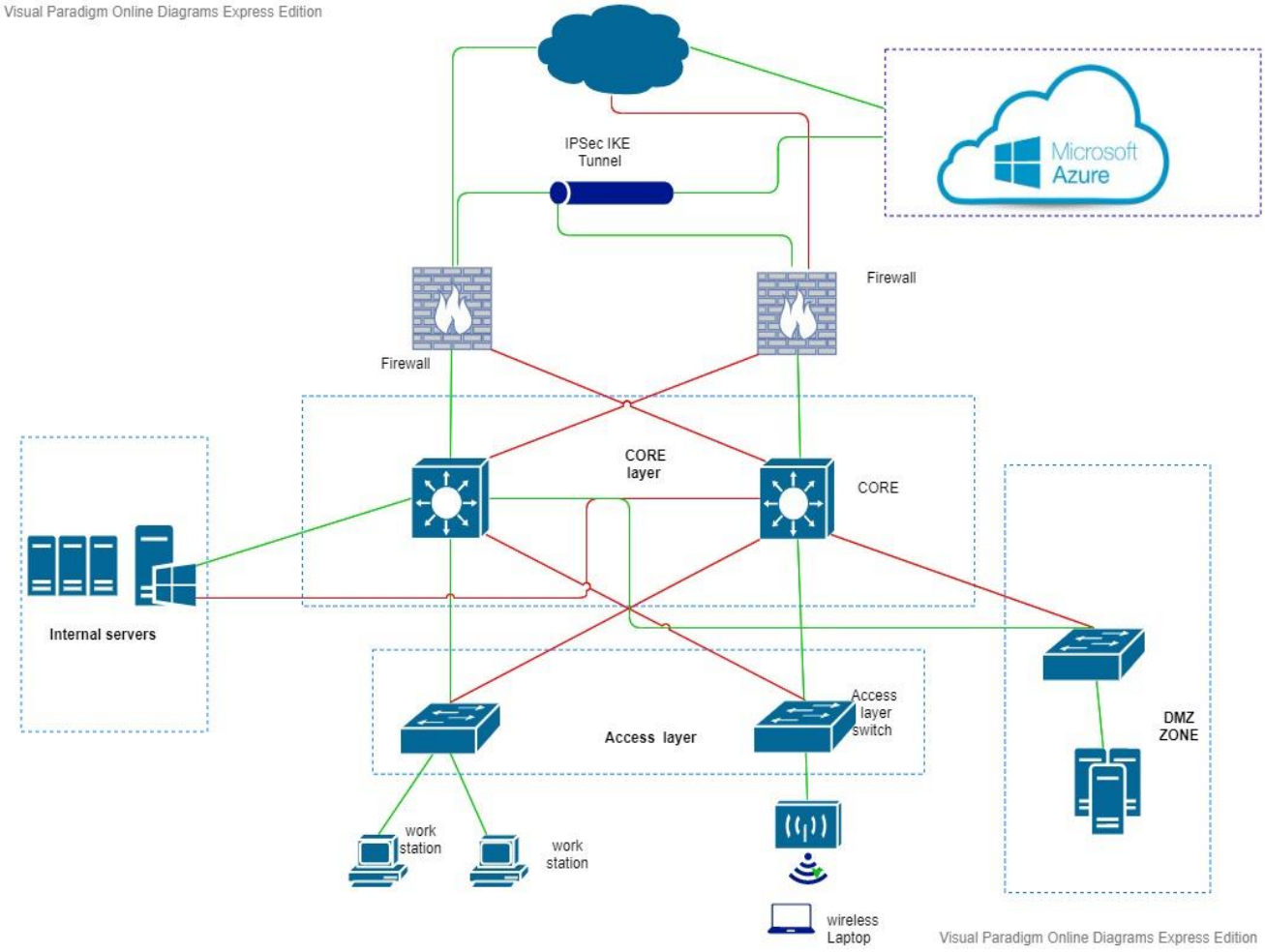


Figure 1.0

AZURE Infrastructure Topology

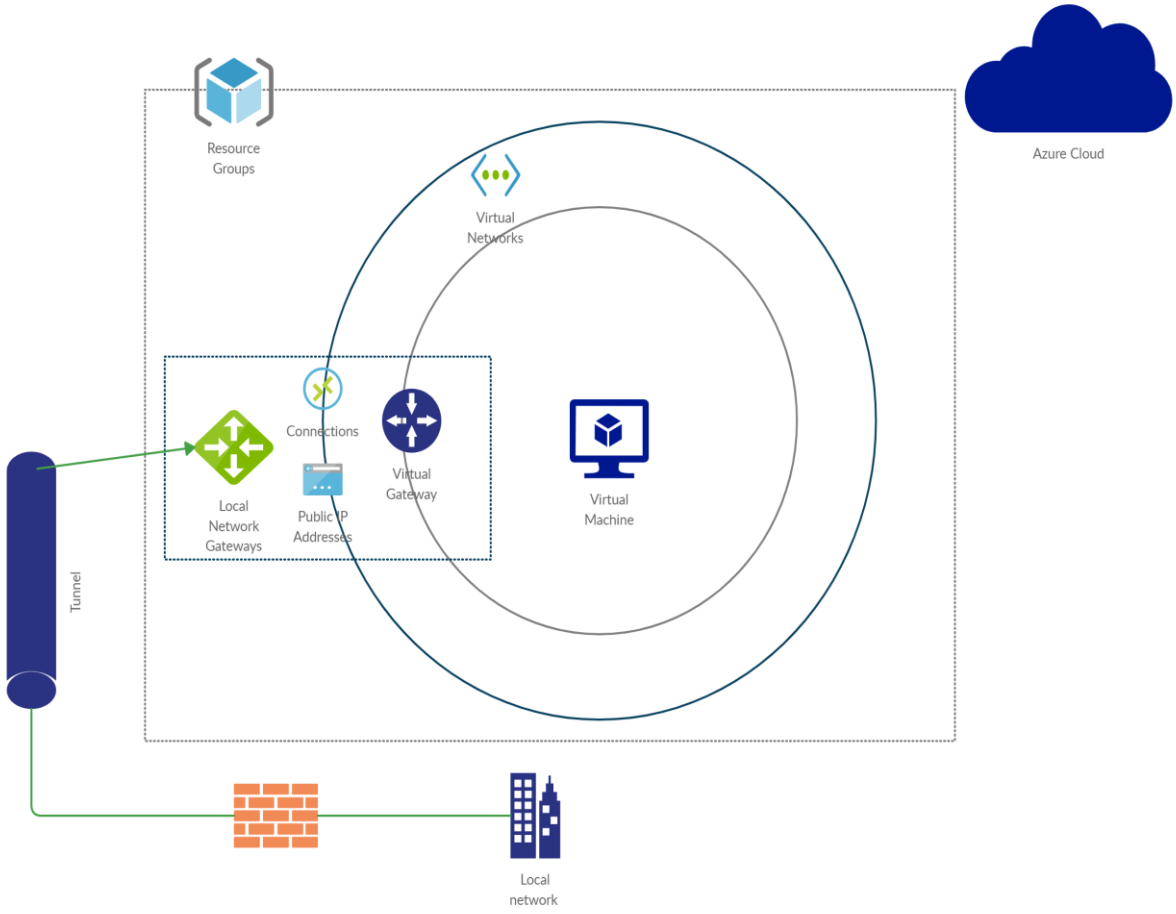


Figure 1.1

Up-graded Sever Rack Design

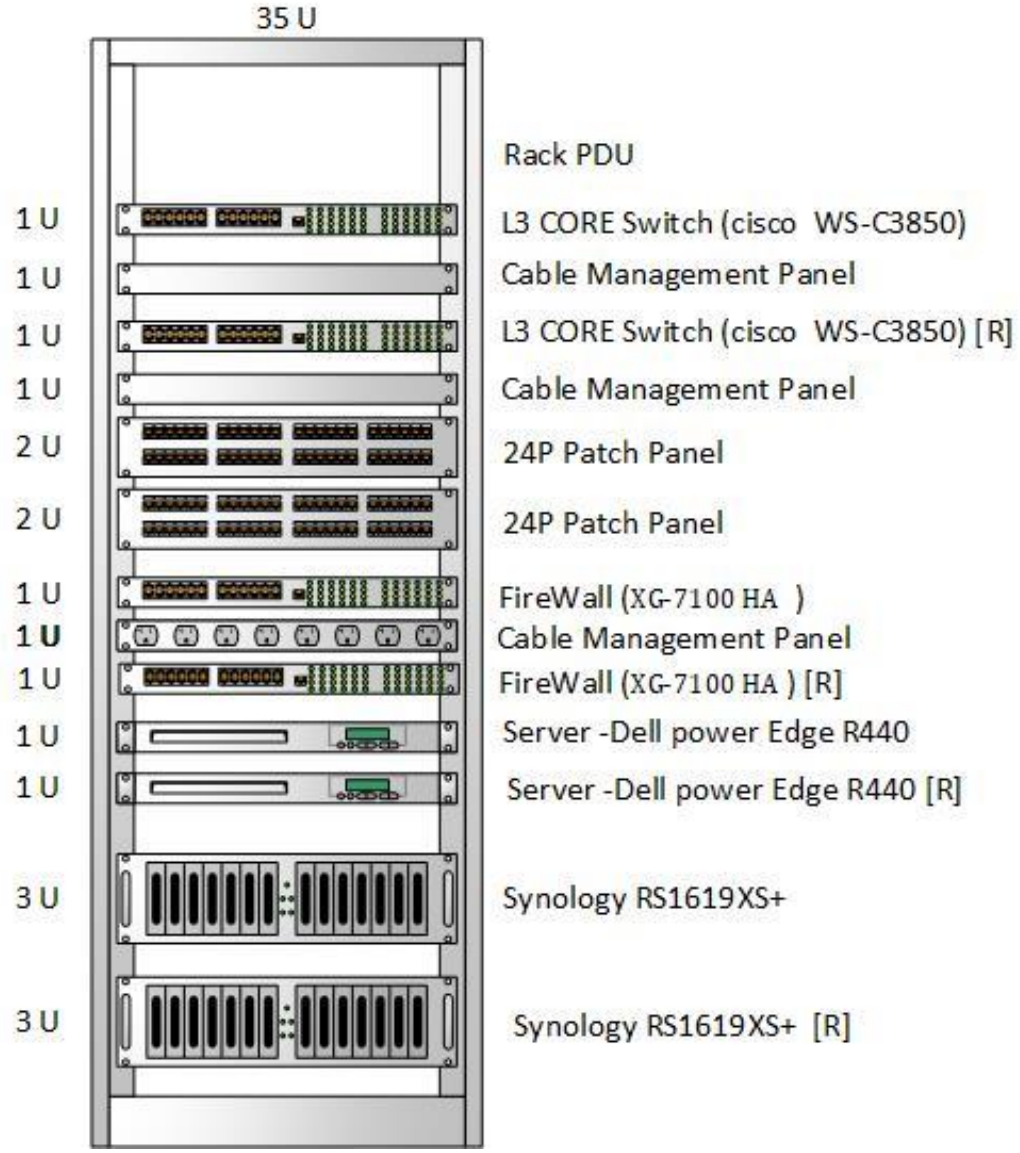


Figure 1.2

Note: [R] is used to the indicate the redundancy (fail-over) devices

Floor plans

Ground Floor

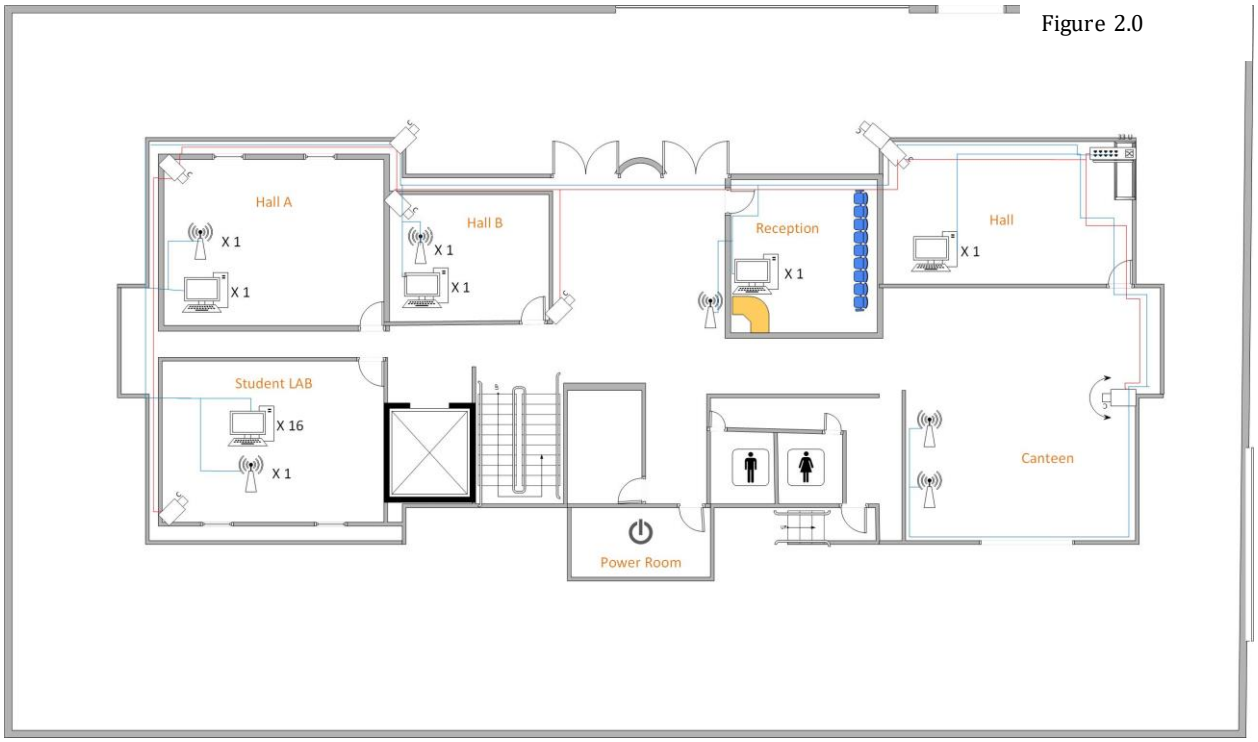


Figure 2.0

Switches	1 x 48
PCs	20
Ap	5
Ip cameras (Sony sncem602rc-indoor/outdoor)	8
Wall cabinet	6U (closed)

Table 1.0

Floor 1

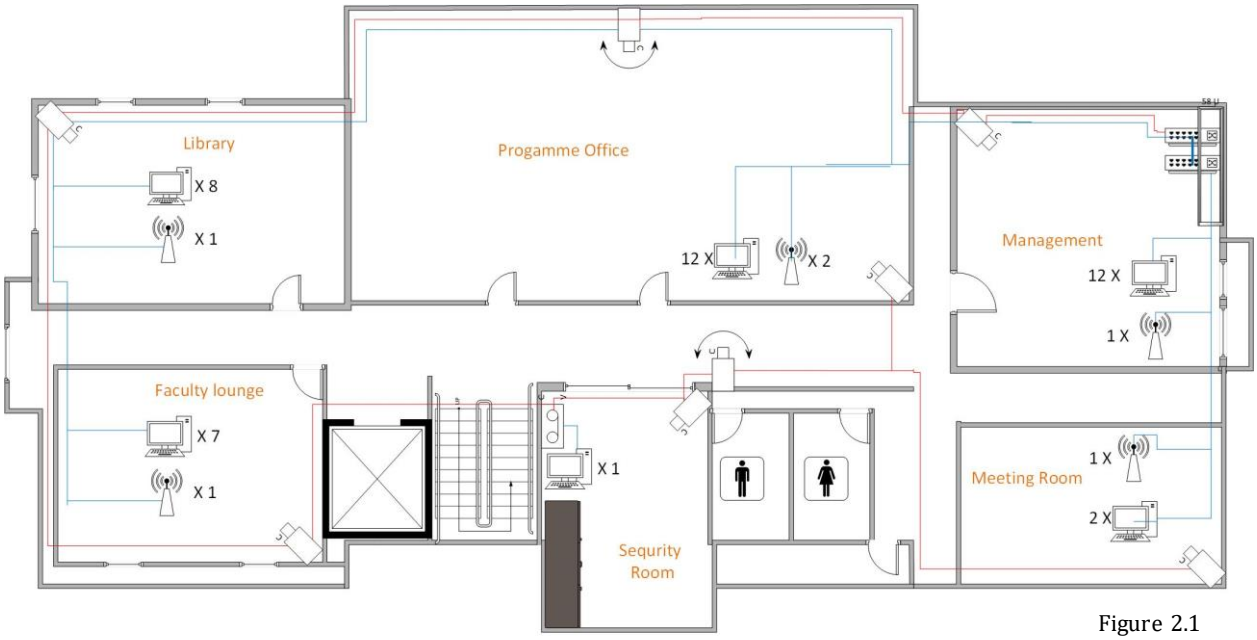


Figure 2.1

Switches	1 x 24 1 x 16
PCs	42
Ap	6
Ip cameras (Sony sncem602rc-indoor/outdoor)	8
Wall cabinet	8U (closed)

Table 1.1

Floor 2

Figure 2.2

Switches	1 x 24
	1 x 16
PCs	20
Ap	4
Ip cameras (Sony sncem602rc-indoor/outdoor)	6
Wall cabinet	6U (closed)

Table 1.3

Floor 3

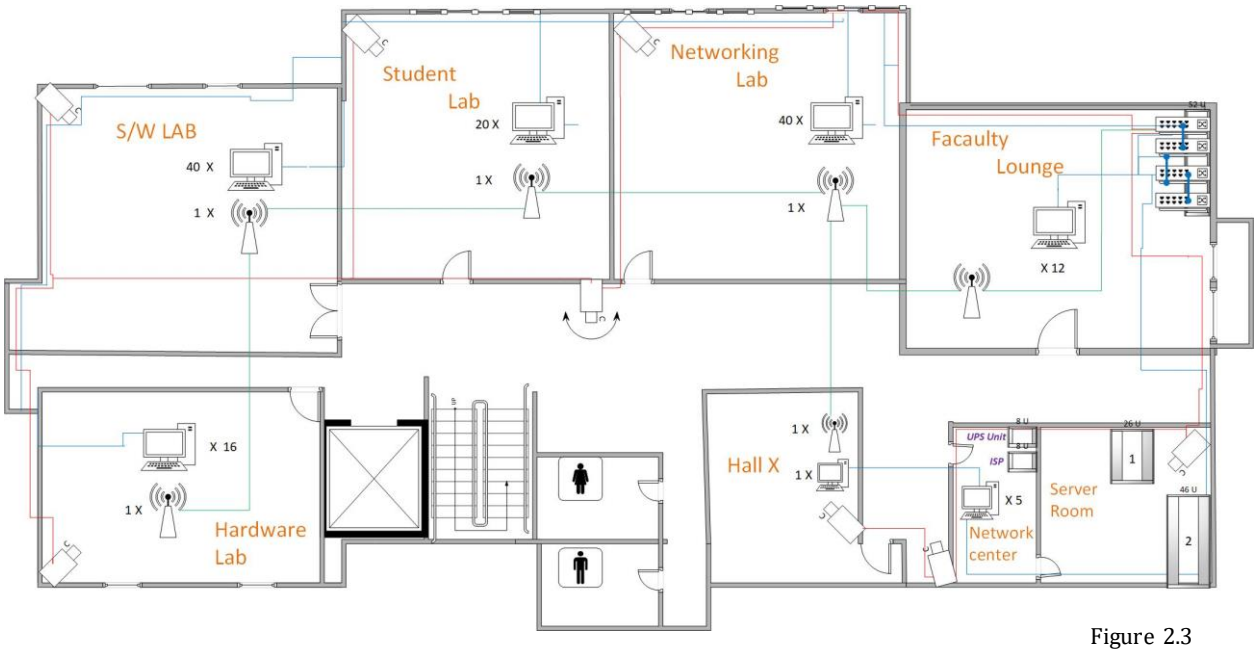


Figure 2.3

Switches	3 x 48 1 x 16
PCs	134
Ap	6
Ip cameras (Sony sncem602rc-indoor/outdoor)	7
Wall cabinet	8U (closed)
Sever room	2 sever racks (48 u), (floor)

Table 1.4

Floor 4

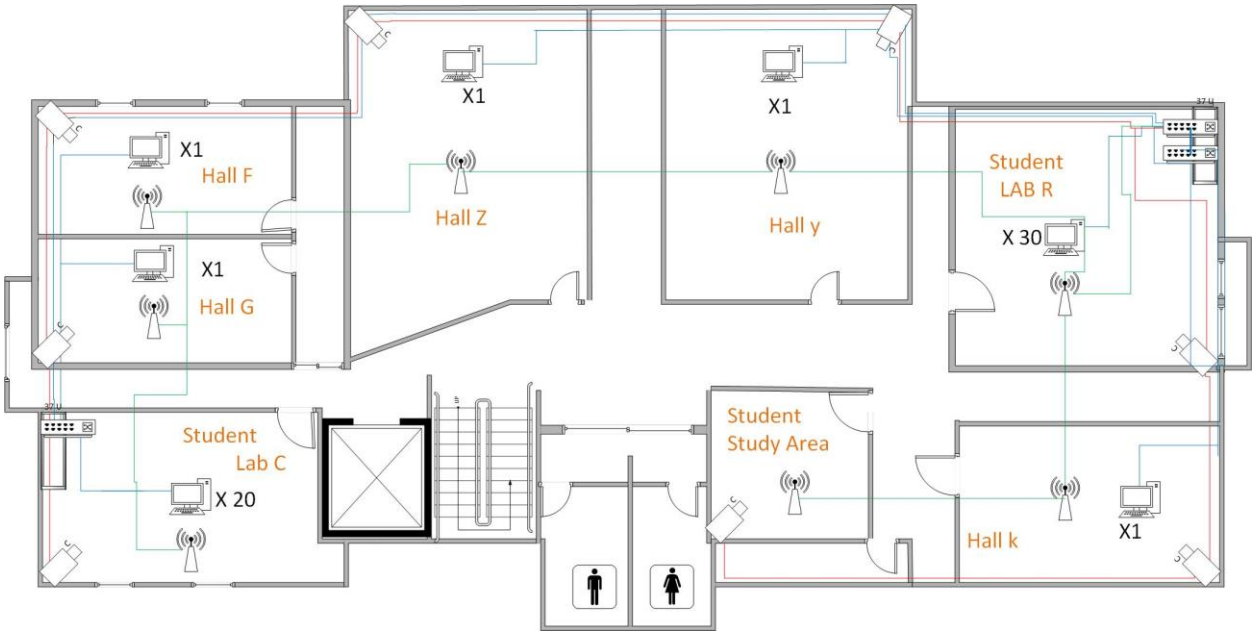


Figure 2.4

Switches	1 x 48 1 x 24
PCs	55
Ap	8
Ip cameras (Sony sncem602rc-indoor/outdoor)	8
Wall cabinet	6U (closed)

Table 1.5

RF Plan for Wireless Devices

The RF Design for position of the access point with detail from Cisco AIRONET 1815 designed by the RF Tool are as such:

AP Name	Aironet 1815
Vendor	Cisco
Quantity	45
Antenna	Internal
Power level	5 GHz
Protocol	**802.11ac**
Security	• Advanced Encryption Standard (AES) for Wi-Fi Protected Access 3 (WPA3), WPA2, WPA • 802.1X, RADIUS Authentication, Authorization, and Accounting (AAA)
Streams	2
Software	Cisco Mobility Express
Max. Clients	400 clients per AP

Table 2.0

RF Plan for the Ground Floor

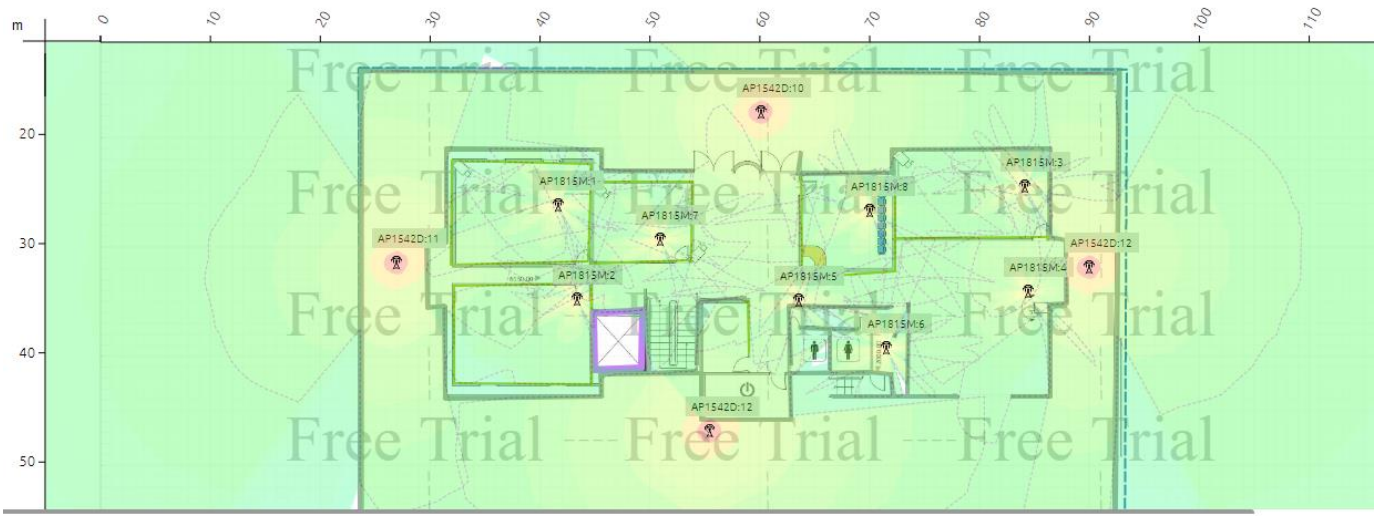


Figure 3.0

RF Plan for the 1st Floor

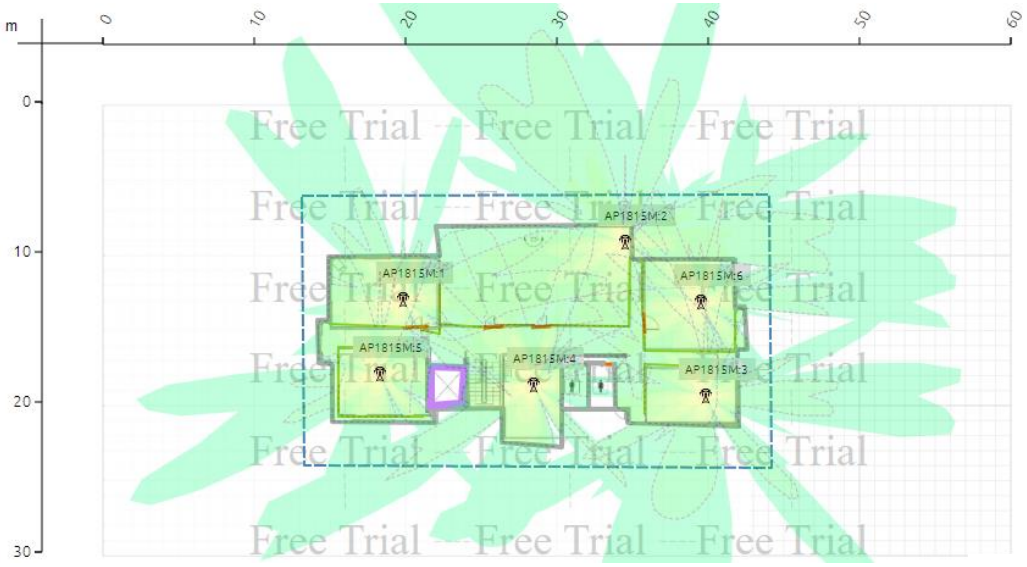


Figure 3.1

- The green zones indicate the uninterrupted signal flow without the creation of Dead zones due to AP device signal collisions in the same floor

RF Plan for the 2nd Floor

Figure 3.2

RF Plan for the 3rd Floor

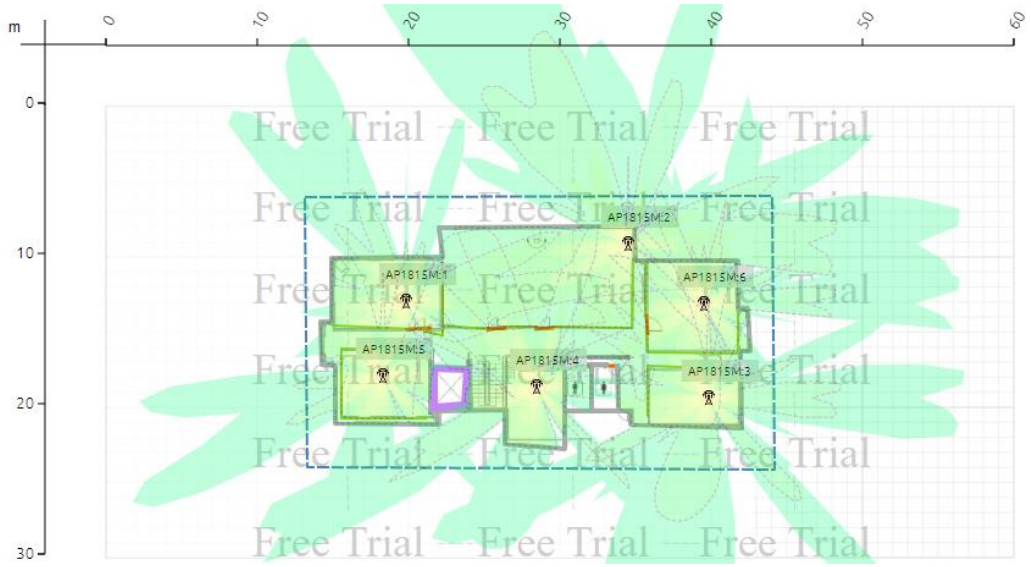


Figure 3.3

Description on the Upgraded network topology

- The university network topology is based on the collapsed core hierarchical model
- There are 2 ISP leased lines acting as a primary (SLT) and backup (DIALOG) respectively towards the network
- Directly connecting to the firewall (XG-7100 HA) securing the external connection through the ISP and internally consisting of tone DMZ switch and two core layers 3 handling the internal network traffic
- Supporting 5 floor switches for each floor of the building, each switch is backed up by a secondary redundant switch is which is directly connected to layers 3 core switches
- The entire network transmission is distributed throughout the network by the access layer or in other words, the floor switches to all users within the network.
- Aps are assigned to each floor with RADIUS authentication allowing each user to connect wirelessly to the internet.
- Furthermore the use of cloud technologies (Azure) are used as redundancy , disaster recovery , and business continuation solutions .
- Also the Azure is further used as a authentications platform where the user credentials are synced with the cloud AD
- Here the use of cloud is directed towards the high availability factor where the student and faculty alike can access the course material in the event of the an infrastructure failure without any hindrance to their daily workloads

Port assignment

Device	***Port***	***Connected device***	***Connected port***
Core layer 3 switch-Firewall	Gig 1/0/2 Gig 1/0/3 (trunking and EtherChannel)	Core layer 3 switch-02 (redundant)	Gig 1/0/2 Gig 1/0/3 (trunking and EtherChannel)
Core layer 3 switch-NAS	Gig 1/0/1	Core layer 3 switch-02 (redundant)	Gig 1/0/1
Core layer 3 switch-Server 1(	Gig 1/0/5	Firewall	Wan port B
Core layer 3 switch-Server 2	Gig 1/0/6-24 (uplink)	Floor switches	Gig 1/0/6-24 (uplink)
Core layer 3 switch-02 (redundant)	Gig 1/0/6-24 (uplink)	Floor switches	Gig 1/0/6-24 (uplink)

Table 3.0

Device Analysis

Core switch

Cisco Catalyst 3650-48PD-S

- Converged wired plus wireless access
- Cisco StackWise-160 technology
- Efficient switch operation
- Environmentally responsible
- Network management tools
- Superior QoS

Device Type	Switch - 48 ports - L3 - Managed – stackable
Enclosure Type	Desktop, rack-mountable 1U
Subtype	Gigabit Ethernet
Ports	48 x 10/100/1000 (PoE+) + 2 x 10 Gigabit SFP+
Power Over Ethernet (PoE)	PoE+
PoE Budget	390 W
Performance	Switching capacity: 176 Gbps Forwarding performance: 130.95 Mbps
Capacity	IPv4 routes: 24000 NetFlow entries: 48000 Virtual interfaces (VLANs): 4094 Switched virtual interfaces (SVIs): 1000
MAC Address Table Size	32000 entries
Jumbo Frame Support	9198 bytes
Routing Protocol	RIP-1, RIP-2, EIGRP, static IP routing, RIPng
Remote Management Protocol	SNMP 1, RMON 1, RMON 2, RMON 3, RMON 9, Telnet, SNMP 3, SNMP 2c, SSH, CLI
Authentication Method	Secure Shell (SSH), RADIUS, TACACS+
Software Included	Cisco IOS IP Base
RAM	4 GB
Flash Memory	2 GB
Status Indicators	Port transmission speed, port duplex mode, system, active, status, PoE
Power Device	Internal power supply - hot-plug
Installed Qty	1 (installed) / 2 (max)
Power Redundancy	Optional

Power Redundancy Scheme	1+1 (with optional power supply)
Power Provided	640 Watt
Voltage Required	AC 120/230 V (50/60 Hz)
MTBF	383,760 hours

Figure 4.0

Cisco Catalyst 2960L-16PS-LL

Figure 4.1

- Fanless operation and operational temperature up to 55°C for deployment outside the wiring closet
- Higher mean time between failure, because they have no moving mechanical parts
- Intuitive web UI for easy deployment and management
- Reduced power consumption and advanced energy management features

Device Type	Switch - 16 ports - Managed
Enclosure Type	Desktop, rack-mountable
Subtype	Gigabit Ethernet
Ports	16 x 10/100/1000 + 2 x Gigabit SFP (uplink)
Power Over Ethernet (PoE)	PoE+
PoE Budget	120 W
Performance	Switching bandwidth: 36 Gbps Forwarding bandwidth: 18 Gbps
Capacity	MAC addresses: 8000 VLANs supported: 4094 Active VLANs: 64 SPAN bidirectional sessions: 1
Jumbo Frame Support	10240 bytes
Routing Protocol	IGMP
Remote Management Protocol	SNMP 3, CLI
Authentication Method	Kerberos, Secure Shell (SSH), RADIUS, TACACS+
Software Included	Cisco IOS LAN Lite
Processor	ARM7: 800 MHz
RAM	512 MB
Flash Memory	256 MB
Power Device	Internal power supply
Voltage Required	AC 120/230 V (50 - 60 Hz)
MTBF	2,416,689 hours
Service & Support	Limited warranty - replacement - lifetime - response time: next business day Technical support - consulting - 90 days

Cisco Catalyst 2960X-24PS-L

Figure 4.2

Manufacturer Cisco
Ports Qty 24
Product Line Cisco Catalyst
Stackable Stackable

- 24 Gigabit Ethernet ports with line-rate forwarding performance
- Gigabit Small Form-Factor Pluggable (SFP) uplinks
- Reduced power consumption and advanced energy management features
- USB and Ethernet management interfaces for simplified operations
- Application visibility and capacity planning with integrated NetFlow-Lite

Device Type	Switch - 24 ports - Managed - stackable
Enclosure Type	Desktop, rack-mountable 1U
Subtype	Gigabit Ethernet
Ports	24 x 10/100/1000 (PoE+) + 4 x Gigabit SFP
Power Over Ethernet (PoE)	PoE+
PoE Budget	370 W
Performance	Switching capacity: 216 Gbps Forwarding performance (64-byte packet size): 71.4 Mbps
Capacity	Virtual interfaces (VLANs): 1023
Jumbo Frame Support	9216 bytes
Max Units in A Stack	8
Remote Management Protocol	SNMP 1, RMON 1, RMON 2, RMON 3, RMON 9, Telnet, SNMP 3, SNMP 2c, HTTP, TFTP, SSH, CLI
Authentication Method	Kerberos, Secure Shell (SSH), RADIUS, TACACS+
Software Included	Cisco IOS LAN Base
Processor	: 600 MHz
RAM	512 MB
Flash Memory	128 MB
MTBF	324,280 hours
Service & Support	Limited warranty - advance parts replacement - lifetime - response time: next business day Technical support - consulting - 90 days

Cisco Catalyst 2960X-48TS-L

Figure 4.3

Manufacturer	Cisco
Ports Qty	48
Product Line	Cisco Catalyst
Stackable	Stackable

- 48 Gigabit Ethernet ports with line-rate forwarding performance
- Gigabit Small Form-Factor Pluggable (SFP) uplinks
- Reduced power consumption and advanced energy management features
- USB and Ethernet management interfaces for simplified operations
- Application visibility and capacity planning with integrated NetFlow-Lite

Device Type	Switch - 48 ports - Managed - stackable
Enclosure Type	Desktop, rack-mountable 1U
Subtype	Gigabit Ethernet
Ports	48 x 10/100/1000 + 4 x Gigabit SFP
Performance	Switching capacity: 216 Gbps Forwarding performance (64-byte packet size): 107.1 Mbps
Capacity	Virtual interfaces (VLANs): 1023
Jumbo Frame Support	9216 bytes
Max Units In A Stack	8
Remote Management Protocol	SNMP 1, RMON 1, RMON 2, Telnet, SNMP 3, SNMP 2c, HTTP, TFTP, SSH, CLI
Authentication Method	Kerberos, Secure Shell (SSH), RADIUS, TACACS+
Software Included	Cisco IOS LAN Base
Processor	: 600 MHz
RAM	512 MB
Flash Memory	128 MB
MTBF	442,690 hours

Severs

Dell PowerEdge R440 Rack server

Physical Server Device Specification

Figure 4.4

The institute uses a Dell PowerEdge R440 Rack server which has been configured as follows,

- Base- Dell PowerEdge R440 Rack Server
- Trusted Platform Module (TPM)- Trusted Platform Module 1.2
- Chassis- 2.5" Chassis with up to 8 Hot Plug Hard Drives
- Processor- Intel® Xeon® Silver 4116 2.1G, 12C/24T, 9.6GT/s, 16M Cache, Turbo, HT (85W) DDR4-2400 (x2)
- Processor Thermal Configuration- 2 CPU standard
- Memory DIMM Type and Speed- 2666MT/s RDIMMs
- Memory Configuration Type- Performance Optimized
- Memory- 8GB RDIMM, 2666MT/s, Single Rank
- RAID- C4, RAID 5 for 3 or more HDDs or SSDs
- Hard Drive- 900GB 15K RPM SAS 12Gbps 512n 2.5in Hot-plug Hard Drive
- Network Card- On-Board Broadcom 5720 Dual Port 1Gb LOM
- Additional Network Cards- Broadcom 5720 Dual Port 1 Gb Network LOM Mezz Card(x1)
- IDSDM and VFlash Card Reader- ISDM and Combo Card Reader
- Internal SD Module- 2x 64GB microSDHC/SDXC Card
- Internal Optical Drive- DVD +/-RW, SATA, Internal
- Server Accessories- 8X DVD-ROM, USB, External
- Power Supply- Dual, Hot Plug, Redundant Power Supply (1+1), 550W
- Power Cord- C13 to C14, PDU Style, 12 AMP, 6.5 Feet (2m) Power Cord, North America (x2)
- Bezel- LCD Bezel for x4 and x8 chassis

- BIOS and Advanced System Configuration Settings- Performance BIOS Settings
- Advanced System Configuration Settings- UEFI BIOS Boot Mode with GPT Partition, Energy Star, Fresh Air Cooling
- Rack Rails- ReadyRails Sliding Rails With Cable Management Arm
- System Documentation- Electronic System Documentation and OpenManage DVD Kit, PowerEdge R440/XR2
- Virtualization Software- VMware ESXi 6.5 U2 Embedded Image on Flash Media (License Not Included)
- Operating System- Windows Server®2016, Standard Ed, Secondary OS, No MEDIA,16 CORE
- Microsoft SQL Server- Microsoft SQL Server 2017 Standard, OEM, Includes 5 Device CALs, NFI, ENGLISH
- Licenses- Windows Server® 2016, Standard Ed, Add License,2CORE, NO MEDIA/KEY (x12)
- OS Media Kits- Windows Server®2016, Standard Edition, Secondary OS, Media Kit
- Client Access Licenses- 50-pack of Windows Server 2019/2016 Device CALs (Standard or Datacenter) (x6)
- Additional Software- Open Manage Integration for VMware vCenter - 1 host increment, 5-year license digitally fulfilled
- Embedded Systems Management- iDRAC9, Enterprise
- Group Manager- iDRAC Group Manager, Enabled
- iDRAC Systems Management Options- Static IP, MAC Address Reporting Service
- Keep Your Hard Drive Service- Keep Your Hard Drive, 7 Years
- Warranty- 7 Years ProSupport Plus Next Business Day Onsite Service
- Deployment Services- Basic Deployment Dell Server R-series 1U/2U

Additional upgraded Device Analysis

DMZ Switch

Figure 4.5

Cisco Catalyst 2960L-16PS-LL

- Fan less operation and operational temperature up to 55°C for deployment outside the wiring closet
- Higher mean time between failure, because they have no moving mechanical parts
- Intuitive web UI for easy deployment and management
- Reduced power consumption and advanced energy management features

Device Type	Switch - 16 ports - Managed
Enclosure Type	Desktop, rack-mountable
Subtype	Gigabit Ethernet
Ports	16 x 10/100/1000 + 2 x Gigabit SFP (uplink)
Power Over Ethernet (PoE)	PoE+
PoE Budget	120 W
Performance	Switching bandwidth: 36 Gbps Forwarding bandwidth: 18 Gbps
Capacity	MAC addresses: 8000 VLANs supported: 4094 Active VLANs: 64 SPAN bidirectional sessions: 1
Jumbo Frame Support	10240 bytes
Routing Protocol	IGMP
Remote Management Protocol	SNMP 3, CLI

Authentication Method	Kerberos, Secure Shell (SSH), RADIUS, TACACS+
Software Included	Cisco IOS LAN Lite
Processor	ARM7: 800 MHz
RAM	512 MB
Flash Memory	256 MB
Power Device	Internal power supply
Voltage Required	AC 120/230 V (50 - 60 Hz)
MTBF	2,416,689 hours

DMZ Server

Dell PowerEdge R20 Rack server

Figure 4.6

Physical Server Device Specification

- Base PowerEdge R240 Server
- Trusted Platform Module (TPM) - No Trusted Platform Module
- Cassis 3.5" Chassis with up to 2 Cabled Hard Drives and Software RAID
- Regulatory - PowerEdge R240 CCC and BIS Marking, No CE Marking
- Processor - Intel® Celeron G4930 3.2GHz, 2M cache, 2C/2T, no turbo (54W)
- Memory - 8GB 2666MT/s DDR4 ECC UDIMM
- RAID C22, RAID 1 for S140 Embedded SATA (2 SATA HDDs or SATA SSDs)
- RAID/Internal Storage Controllers -S140 for Software RAID
- Hard Drive - 1TB 7.2K RPM SATA 6Gbps 512n 3.5in Cabled Hard Drive
- Boot Optimized Storage Cards -BOSS controller card + with 1 M.2 Sticks 240G (No RAID),LP
- Operating System - Default OS System
- OS Media Kits -No Media Required
- Embedded Systems Management - iDRAC9 Basic
- Group Manager - iDRAC Group Manager,Disabled
- Password - iDRAC,Factory Generated Password
- iDRAC Systems Management Options - Static IP
- iDRAC Service Module - None

- PCIe Riser - PCIe Riser with Fan with up to 1 LP, x8 PCIe + 1 FH/HL, x16 PCIe Slots

- Additional Network Cards - On-Board Broadcom 5720 Dual Port 1Gb LOM
- IDSDM and VFlash Card Reader - None
- Internal SD Module - None
- Internal Optical Drive - No Internal Optical Drive
- Power Supply Single, Cabled Power Supply, 250W
- Power Cords - NEMA 5-15P to C13 Wall Plug, 125 Volt, 15 AMP, 10 Feet (3m
- Bezel - No Bezel
- BIOS and Advanced System Configuration Settings - Performance BIOS Setting

- Advanced System Configurations - UEFI BIOS Boot Mode with GPT Partition
- Rack Rails 1U/2U 2/4-Post Static Rails - System Documentation No Systems Documentation, No OpenManage DVD Kit
- Secondary OS - None
- Enabled Virtualization - Yes
- Warranty - Basic Next Business Day 12 Months, 12 Month(s)
- Extended Services - Basic Next Business Day,)

Sever- Failover

Dell PowerEdge R440 Rack server

Additional Updated Physical Server Device Specification as the FAIL over Server

Figure 4.7

The institute uses a Dell PowerEdge R440 Rack server which has been configured as follows,

- Base- Dell PowerEdge R440 Rack Server
- Trusted Platform Module (TPM)- Trusted Platform Module 1.2
- Chassis- 2.5" Chassis with up to 8 Hot Plug Hard Drives
- Processor- Intel® Xeon® Silver 4116 2.1G, 12C/24T, 9.6GT/s, 16M Cache, Turbo, HT (85W) DDR4-2400 (x2)
- Processor Thermal Configuration- 2 CPU standard
- Memory DIMM Type and Speed- 2666MT/s RDIMMs
- Memory Configuration Type- Performance Optimized
- Memory- 8GB RDIMM, 2666MT/s, Single Rank
- RAID- C4, RAID 5 for 3 or more HDDs or SSDs
- Hard Drive- 900GB 15K RPM SAS 12Gbps 512n 2.5in Hot-plug Hard Drive
- Network Card- On-Board Broadcom 5720 Dual Port 1Gb LOM
- Additional Network Cards- Broadcom 5720 Dual Port 1 Gb Network LOM Mezz Card(x1)
- IDSDM and VFlash Card Reader- ISDM and Combo Card Reader

- Internal SD Module- 2x 64GB microSDHC/SDXC Card
- Internal Optical Drive- DVD +/-RW, SATA, Internal
- Server Accessories- 8X DVD-ROM, USB, External
- Power Supply- Dual, Hot Plug, Redundant Power Supply (1+1), 550W
- Power Cord- C13 to C14, PDU Style, 12 AMP, 6.5 Feet (2m) Power Cord, North America (x2)
- Bezel- LCD Bezel for x4 and x8 chassis
- BIOS and Advanced System Configuration Settings- Performance BIOS Settings
- Advanced System Configuration Settings- UEFI BIOS Boot Mode with GPT Partition, Energy Star, Fresh Air Cooling
- Rack Rails- ReadyRails Sliding Rails With Cable Management Arm
- System Documentation- Electronic System Documentation and OpenManage DVD Kit, PowerEdge R440/XR2
- Virtualization Software- VMware ESXi 6.5 U2 Embedded Image on Flash Media (License Not Included)
- Operating System- Windows Server®2016, Standard Ed, Secondary OS, No MEDIA,16 CORE
- Microsoft SQL Server- Microsoft SQL Server 2017 Standard, OEM, Includes 5 Device CALs, NFI, ENGLISH
- Licenses- Windows Server® 2016, Standard Ed, Add License,2CORE, NO MEDIA/KEY (x12)
- OS Media Kits- Windows Server®2016, Standard Edition, Secondary OS, Media Kit
- Client Access Licenses- 50-pack of Windows Server 2019/2016 Device CALs (Standard or Datacenter) (x6)

- Additional Software- Open Manage Integration for VMware vCenter - 1 host increment, 5-year license digitally fulfilled
- Embedded Systems Management- iDRAC9, Enterprise
- Group Manager- iDRAC Group Manager, Enabled
- iDRAC Systems Management Options- Static IP, MAC Address Reporting Service
- Keep Your Hard Drive Service- Keep Your Hard Drive, 7 Years
- Warranty- 7 Years ProSupport Plus Next Business Day Onsite Service
- Deployment Services- Basic Deployment Dell Server R-series 1U/2U

Firewall

XG-7100 1U HA

Figure 4.8

TECHNICAL SPECIFICATIONS

CPU	Intel "Denverton" Atom C3558 2.2 GHz with QuickAssist
CPU Cores	4
Memory Options	8GB DDR4 Non ECC 16GB DDR4 SODIMM Additional Memory (24GB Total)
Storage Options	32GB eMMC Flash 256GB M.2 SATA SSD
Network Interfaces	2x 10GbE Intel x553 SFP+ Ports 8-port 1Gbps Marvell 88E6190 switch, uplinked at 5 Gbps to Intel SoC (2x 2.5 Gbps) for LAN.
Network Expansion Options	4-Port Intel GbE
USB Ports	1X 3.0 2X 2.0
Console Port	Mini USB
Max Active Connections	8.0 Million
Power	Internal AC/DC 100-240V, 50-60 Hz
Case	19" 1U rack mount
Dimensions	19" (483mm) x 1.75" (44mm) x 8.5" (216mm)

Cooling	Active
Operating Temperature	0°C (32°F) to 60°C (140°F)
Hardware Warranty	12 Months
Certifications	CE, FCC, RoHS, UL
Power Consumption	20W (idle)
Support Options	Netgate Global Support

Additional information

- Connect via encrypted Virtual Private Networks (VPN) between offices, let mobile workers connect securely, or connect to the Cloud!
- Use the built-in Amazon VPC Wizard to easily establish VPN connections with Amazon EC2 cloud instances.
- Flexible configuration and support for VPN, load balancing, reporting and monitoring.

- The XG-7100 1U Netgate® Security Gateway Appliance High Availability 2-unit system is designed for fault tolerance and failover to prevent loss of critical services in case of hardware or software failure[1]. It can be configured as a firewall, LAN or WAN router, VPN appliance, DHCP Server, DNS Server, and IDS/IPS with optional packages to deliver a high performance, high throughput front-line security appliance at an excellent price.

- Built with performance, versatility, and low total cost of ownership in mind, this is a low power system built to provide a high level of I/O throughput and optimal performance.

NAS

RS1619xs+

Figure 4.9

CPU	CPU Model	Intel Xeon D-1527
	CPU Architecture	64-bit
	CPU Frequency	4-core 2.2 (base) / 2.7 (turbo) GHz
	Hardware Encryption Engine (AES-NI)	
Memory	System Memory	8 GB DDR4 ECC UDIMM
	Memory Module Pre-installed	8 GB (8 GB x 1)
	Total Memory Slots	4
	Maximum Memory Capacity	64 GB (16 GB x 4)
Storage	Drive Bays	4
	Maximum Drive Bays with Expansion Unit	16 (RX1217/RX1217RP x 1)

	M.2 Drive Slots	2 (NVMe & SATA)
	Compatible Drive Type	• 3.5" SATA HDD • 2.5" SATA HDD • 2.5" SATA SSD • M.2 2280 NVMe & SATA SSD
•	Maximum Single Volume Size*	• 200 TB (32 GB RAM required, for RAID 5 or RAID 6 groups only) • 108 TB
•	Hot Swappable Drive	
External Ports	RJ-45 1GbE LAN Port	4 (with Link Aggregation / Failover support)
	USB 3.0 Port	2
	Expansion Port	1
File System	Internal Drives	• Btrfs • EXT4
•	External Drives	• Btrfs • EXT4 • EXT3 • FAT • NTFS

Others	System Fan	40 mm x 40 mm x 2 pcs
	Fan Speed Mode	• Full-Speed Mode • Cool Mode • Quiet Mode
•	Easy Replacement System Fan	
	Power Recovery	
	Noise Level*	39.3 dB(A)
	Scheduled Power On / Off	
	Wake on LAN / WAN	
	Power Supply Unit / Adapter	150 W
	Redundant Power Supply	
	AC Input Power Voltage	100 V to 240 V AC
	Power Frequency	50/60 Hz, Single Phase
	Power Consumption*	68.68 W (Access) 34.78 W (HDD Hibernation)

IP Cameras and NVR

Indoor Cameras

- **DS-2CD51C5G0-IZS - 12 MP Indoor Varifocal Network Dome Camera**

Figure 5.0

The Hikvision DS-2CD51C5G0-IZS Indoor Varifocal Network Dome Camera provides high definition output. It supports H.265+ video compression, which assures savings in bandwidth and storage.

- Minimum Illumination: Color: 0.008 lux @ (f/1.2, AGC on), 0 lux with IR
- 4000 x 3000 Resolution @ 20 fps
- 2.8 to 12 mm Motorized Lens
- H.265+, H.265, H.264+, H.264
- Main Stream: 20 fps (4000 × 3000), 30 fps (4096 × 2160, 3840 × 2160, 3072 × 1728, 2560 × 1440, 1920 × 1080, 1280 x 720)
- EXIR 2.0 with up to 100 ft (30 m) IR Range
- 12 VDC and PoE (802.3af)

- **DS-2CD5185G0-IZS - 8 MP Indoor VF Network Dome Camera**

Figure 5.1

The Hikvision DS-2CD5185G0-IZS Indoor VF Network Dome Camera provides high definition output. It supports H.265+ video compression, which assures savings in bandwidth and storage.

- Minimum Illumination: Color: 0.009 Lux @ (f/1.2, AGC ON), 0 Lux with IR
- 3840 x 2160 Resolution @ 30 fps
- 2.8 to 12 mm Motorized Lens
- H.265+, H.265, H.264+, H.264
- Main Stream: 30 fps (3840 x 2160, 3200 x 1800, 2560 x 1440, 1920 x 1080, 1080 x 720)
- Up to 100 ft (30 m) IR Range
- 12 VDC and PoE (802.3af)

➢ **DS-2CD5165G0-IZS - 6 MP Indoor VF Network Dome Camera**

Figure 5.2

The Hikvision DS-2CD5165G0-IZS Indoor VF Network Dome Camera provides high definition output. It supports H.265+ video compression, which assures savings in bandwidth and storage.

- Minimum Illumination: Color: 0.009 Lux @ (f/1.2, AGC ON), 0 Lux with IR
- 3200 x 1800 Resolution @ 30 fps
- 2.8 to 12 mm Motorized Lens
- H.265+, H.265, H.264+, H.264
- Main Stream: 30 fps (3200 x 1800, 3072 x 2048, 2560 x 1440, 1920 x 1080, 1280 x 720)
- Up to 100 ft (30 m) IR Range
- 12 VDC and PoE (802.3af)

Outdoor Cameras

➢ **DS-2CD5AC5G0-IZHS - 12 MP Outdoor Varifocal Network Bullet Camera**

Figure 5.3

The Hikvision DS-2CD5AC5G0-IZHS Outdoor Varifocal Network Bullet Camera provides high definition output. It supports H.265+ video compression, which assures savings in bandwidth and storage.

- Minimum Illumination: Color: 0.008 lux @ (f/1.2, AGC on), 0 lux with IR
- 4000 x 3000 Resolution @ 20 fps
- 2.8 to 12 mm Motorized Lens
- H.265+, H.265, H.264+, H.264
- Main Stream 20 fps (4000 × 3000), 30 fps (4096 × 2160, 3840 × 2160, 3072 × 1728, 2560 × 1440, 1920 × 1080, 1280 x 720)
- EXIR 2.0 with up to 165 ft (50 m) IR Range
- 12 VDC and PoE (802.3af)

➢ **DS-2CD6924G0-IHS(/NFC)**

Figure 5.4

- 180° splicing image to present all scenes that covered by the camera
- Built-in heater, ultra-low temperature start up to -40℃
- Water and dust resistant (IP67) and vandal proof(IK10)
- Smart human density counting and public behaviors analysis to increase public area safety

- **DS-2CD2347G1-L(U)**

Figure 5.5

- Max. 2688 × 1520 @30fps
- 2.8 mm/4 mm/6 mm fixed lens
- H.265+, H.264+
- 120dB WDR
- Built-in mic (Optional)
- Built-in micro SD/SDHC/SDXC slot, up to 128G
- 0.0014 @ (F1.0, AGC ON), 0 Lux with Light
- 24/7 full time color

Network Video Recorders

- **DS-7732NI-Q4**

Figure 5.6

- Up to 32 channel IP cameras can be connected
- Supports decoding H.265+/H.265/H.264+/H.264 video formats
- Up to 4K high-definition live view, storage and playback
- Up to 4-ch @ 1080p decoding capacity
- Up to 256 Mbps high incoming bandwidth ensures IP cameras can be connected
- 1 HDMI and 1 VGA interfaces: both interfaces support independent video output
- 4 HDDs for continuous video recording
- Compatible with third-party network cameras

Ip Addressing Table

Major Network:172.16.0.0/19 Available IP addresses in major network:8190 Number of IP address needed :2647 Available Ip Addresses in allocated subnets 4094 About 36% of available major network address apace is used About 86% of subnetted network address space is used							
Subnet Name	**Needed Size**	**Allocated Size**	**Addess**	**Mask**	**Dec Mask**	**Assignable Range**	**Broadcast**
Student Wifi	2000	2046	172.16.0.0	/21	255.255.248.0	172.16.0.1 - 172.16.7.254	172.16.7.255
Labs	275	510	172.16.8.0	/23	255.255.254.0	172.16.8.1 - 172.16.9.254	172.16.9.255
Guest WIFI	75	126	172.16.10.0	/25	255.255.255.128	172.16.10.1 - 172.16.10.126	172.16.10.127
Staff WFI	70	126	172.16.10.128	/26	255.255.255.128	172.16.10.129 - 172.16.10.254	172.16.10.255
Academic staff WIFI	50	62	172.16.11.0	/26	255.255.255.192	172.16.11.1 - 172.16.11.62	172.16.11.63
CCTV Surveillance	45	62	172.16.11.64	/26	255.255.255.192	172.16.11.65 - 172.16.11.126	172.16.11.127
AP Management	40	62	172.16.11.128	/26	255.255.255.192	172.16.11.129 - 172.16.11.190	172.16.11.191
Exams	20	30	172.16.11.192	/27	255.255.255.224	172.16.11.193 - 172.16.11.222	172.16.11.223
Finanace	13	14	172.16.11.224	/28	255.255.255.240	172.16.11.225 - 172.16.11.238	172.16.11.239
Device Management	13	14	172.16.11.240	/28	255.255.255.240	172.16.11.241 - 172.16.11.254	172.16.11.255
HR	10	14	172.16.12.0	/28	255.255.255.240	172.16.12.1 - 172.16.12.14	172.16.12.15
Internal server	10	14	172.16.12.16	/28	255.255.255.240	172.16.12.17 - 172.16.12.30	172.16.12.31
IT Dept	10	14	172.16.12.32	/28	255.255.255.240	172.16.12.33 - 172.16.12.46	172.16.12.47
Marketing	8	14	172.16.12.48	/28	255.255.255.240	172.16.12.49 - 172.16.12.62	172.16.12.63
Firewall 1	2	2	172.16.12.64	/30	255.255.255.252	172.16.12.65 - 172.16.12.66	172.16.12.67
Firewal 2	2	2	172.16.12.68	/30	255.255.255.252	172.16.12.65 - 172.16.12.66	172.16.12.71
Conncetion 1	2	2	172.16.12.72	/30	255.255.255.252	172.16.12.73 - 172.16.12.74	172.16.12.75
Connection 2	2	2	172.16.12.76	/30	255.255.255.252	172.16.12.77 - 172.16.12.78	172.16.12.79
Storage	4	4	192.167.5.0	/28	255.255.255.240	192.167.5.1 - 192.167.5.4	192.167.5.5
DMZ	8	14	10.10.10.0	/28	255.255.255.240	10.10.10.1- 10.10.10.14	10.10.10.15

Table 4.0

Bandwidth Requirements

The ITIT campus network is provided with an Internet connection via the installation of two leased lines for redundancy purposes. The bandwidth calculations are done assuming peak hours- the time of the day when the most Internet-connected devices are being used concurrently, so it will provide us with an image of the campus network during its most bandwidth-stressful time.
The users' bandwidth requirements were calculated by classifying the, under the following criteria.

- Light users that only need to carry out simple web-surfing and emails.
- Medium users that require a bandwidth to carry out more intensive Internet services such as file downloading, streaming and emailing.
- Intensive users that require a high bandwidth connection for more intensive tasks such as live video-conferencing.

The bandwidth separated for all three types of users are shown below.

1. Light users- 125kbps
2. Medium users- 500kbps
3. Heavy users- 1Mbps

Department	Highest no. concurrent users	Usage per user	Bandwidth Required
PC Labs	200	500kbps	100Mbps
Academic Staff	50	500kbps	25Mbps
Staff Department	30	1Mbps	30Mbps
Examination	15	500kbps	7.5Mbps
CCTV Surveillance	10	500kbps	5Mbps
Internal servers	14	1Mbps	5Mbps
Guest WIFI	70	125kbps	0.25Mbps
Device Management	45	125kbps	2Mbps
Finance	8	125kbps	0.25Mbps
HR	8	125kbps	0.25Mbps
Marketing	7	125kbps	1Mbps
IT Dept.	10	125kbps	3Mbps
Student WIFI	2000	125kbps	0.25Mbps
AP Management	2	125kbps	1Mbps

Table 5.0

Updated VLAN Description

VLANs are created across each floor as sub-networks that carry out each user group traffic within the network efficiently. This is also done to remove the uncontrolled broadcast traffic reaching another network. VLAN also provides a layer of network security and cost reduction option by logically separating hosts which is connected to the same switch (no need for additional switches)

VLAN 3 - Student WIFI
VLAN 3 is used by the entire student population to gain access to the internet and roam social media. This VLAN is placed with speed restrictions, as this is not a critical service
Range: 172.16.0.1 - 172.16.7.254

VLAN 4 - Lab
The Lab uses to VLAN 4 where file sharing, lecture material and hands on contented is available with much restrictions.
Range: 172.16.8.1 - 172.16.9.254

VLAN 5 -Guest WIFI
VLAN 5 is used to by the institute Guests usability for 70 users for internet access with different speed due to the Guest user account
Range: 172.16.10.1 - 172.16.10.126

VLAN 6 - Academic Staff
VLANs 6 is used to send/receive secure, confidential examination papers and share lecture material in collaboration with the faculty VLAN
Range: 172.16.11.1 - 172.16.11.62

VLAN 7 – CCTV Surveillance
The security VLAN is provided with resistive access to the internet and limited user access privileges with a wider bandwidth for video transmission.
In addition, This is a separate VLAN that handles the Ip camera traffic received from all five floors in the institute
Range: 172.16.11.65 - 172.16.11.126

VLAN 8- Exams
Due to the sensitivity of the exams and data leakage, this particular VLAN has restrictive access to internet and other VLAN traffic
Range: 172.16.11.193 - 172.16.11.222

VLAN 9 – Finance
VLAN 9 used to support the Finance department by providing secure financial transactions, create databases and reports of daily transactions
Range: 172.16.11.225 - 172.16.11.238

VLAN 10- AP management

Also, a separate VLAN that mangoes the AP Ip distribution through the network. Each AP is assigned an IP for identification when connection to the server

Range: 172.16.11.129 - 172.16.11.190

VLAN 11 - HR

The administrative and human resource management is given a less restrictive usage as daily updates are done with routinely tasks in the administration

Range: 172.16.12.1 - 172.16.12.14

VLAN 12- Internal Servers

The internal network related traffic is sent through this VLAN, system related information and other sys log information including critical messages are tagged by this VLAN

Range: 172.16.12.17 - 172.16.12.30

VLAN 13- Device Management

Used to manage and remotely handle the network devices in the network. Currently an internal LAN with 14 switches

Range: 172.16.11.241 - 172.16.11.254

VLAN 14 - IT Department

The system administrators and system support engineers use this VLAN for internal system communication and error analysis

Range: 172.16.12.33 - 172.16.12.46

VLAN 15 - Marketing

Used for higher bandwidth as this marketing handles the publicity and social presence of the university, with real time interactive communication

Range: 172.16.12.49 - 172.16.12.62

VLAN 16 – Staff WIFI

Limited WIFI speed for the internet accessibility of the university non faculty staff members

Range: 172.16.10.129 - 172.16.10.254

VLAN floor layout

VLAN Number	Description
VLAN 3	ITIT Wi-Fi for 2000 users through the 5 floors
VLAN 4	Student LABs with a total of 275 PCs in the ground , 2nd , 3rd, and 4th floors
VLAN 5	Guest network limited with 70 logins through the 5 floors
VLAN 6	Faculty with 30 PC nodes attached in the 1st and 3rd floor
VLAN 7	Security with 10 PC nodes in the 1st floor
VLAN 8	Exam center with 15 PC nodes attached in the 2nd floor
VLAN 9	Management with 30 PC nodes in the 1st floor
VLAN 10	AP management with 40 AP devices through the 5 floors
VLAN 11	The HR is with 12 PC nodes in the 2nd floor
VLAN 12	Internal servers reside in the3rd Floor with restricted access to the server room
VLAN 13	Device Management with 13 switches
VLAN 14	IT Department, where the system supports and system management resides in the network. Positioned in the 3rd floor
VLAN 15	Marketing where the publicity of the ITIT institute is done. Unrestrained internet access and social media is accessible. Positioned in the 2nd floor
VLAN 16	Staff Wi-Fi is provided to all 4 floors within the building
VLAN 17	Storage is reserved for traffic with internal storage related traffic in the 3rd floor
VLAN 99	Native VLAN where all untagged traffic is sent throughout the internal network

Table 6.0

Protocols and services

Routing protocols

Default routing

This is configured on core routers to route the traffic from inside network to ISP router for unknown traffic (towards internet)

Inter VLAN routing

Core routers are configured to route the traffic between different VLAN in the network. The traffic will reach the core routers from core switch which are connected by trunk link. All VLAN networks will be shown as directly connected routes in routing table (sub interfaces are used)

HSRP Protocol

HSRP Background and Operations

“”One way to achieve near-100 percent network uptime is to use HSRP, which provides network redundancy for IP networks, ensuring that user traffic immediately and transparently recovers from first hop failures in network edge devices or access circuits.

- This is routing protocol is used for the ITIT University core layer switches due to the availability techniques it utilizes providing the network infra-structure with dependency and reliable uptime

By sharing an IP address and a MAC (Layer 2) address, two or more Multi-layer switches can act as a single "virtual" multi-layer switch.

The members of the virtual Multi-layer switch group continually exchange status messages. This way, one Multi-layer switch can assume the either routing responsibility of another, should it go out of commission for planned or unplanned reasons. Hosts continue to forward IP packets to a consistent IP and MAC address, and the changeover of devices doing the routing is transparent.

HSRP Operation

A large class of legacy host implementations that don't support dynamic discovery are capable of configuring a default Core Switch Running a dynamic Multi-layer switch discovery mechanism on every host may not be feasible for a number of reasons, including administrative overhead, processing overhead, security issues, or lack of a protocol implementation for some platforms. HSRP provides failover services to these hosts.

Using HSRP, a set of Multi-layer switches that works in concert to present the illusion of a single virtual Multi-layer switch to the hosts on the LAN. This set is known as an HSRP group or a standby group. A single Multi-layer switch elected from the group is responsible for forwarding the packets that hosts send to the virtual Multi-layer switch. This Multi-layer switch

is known as the Active Multi-layer switch. Another Multi-layer switch is elected as the Standby Multi-layer switch. In the event that the Active Core switch fails, the Standby assumes the packet-forwarding duties of the Active Core switch. Although an arbitrary number of Multi-layer switch may run HSRP, only the Active Core switch forwards the packets sent to the virtual Multi-layer switch.

To minimize network traffic, only the Active and Standby Core switches send periodic HSRP messages once the protocol has completed the election process. If the Active Core switch fails, the Standby Core switch takes over as the Active Core switch. If the Standby Multi-layer switch fails or becomes the Active Multi-layer switch, then another Multi-layer switch is elected as the Standby Multi-layer switch.

On a particular LAN, multiple hot standby groups may coexist and overlap. Each standby group emulates a single virtual Multi-layer switch. The individual Multi-layer switch may participate in multiple groups. In this case, the Multi-layer switch maintains separate state and timers for each group.

Each standby group has a single, well-known MAC address, as well as an IP address.

HSRP Features

1. Preemption

The HSRP preemption feature enables the Core switch with highest priority to immediately become the Active Core switch. Priority is determined first by the priority value that you configure, and then by the IP address. In each case a higher value is of greater priority.

When a higher priority Core switch preempts a lower priority Core switch, it sends a coup message. When a lower priority active Core switch receives a coup message or hello message from a higher priority active Core switch, it changes to the speak state and sends a resign message.

2. *Preempt Delay*

The preempt delay feature allows preemption to be delayed for a configurable time period, allowing the Core switch to populate its routing table before becoming the active Core switch.

To configure HSRP priority and preemption, use the standby **[***group***] [priority** *number***] [preempt [delay [minimum]***seconds***] [sync** *seconds***]]** command.

3. Interface Tracking

Interface tracking allows you to specify another interface on the Core switch for the HSRP process to monitor in order to alter the HSRP priority for a given group.

If the specified interface's line protocol goes down, the HSRP priority of this Core switch is reduced, allowing another HSRP Core switch with higher priority can become active (if it has preemption enabled).

To configure HSRP interface tracking, use the standby [group] track interface [priority] command.

When multiple tracked interfaces are down, the priority is reduced by a cumulative amount. If you explicitly set the decrement value, then the value is decreased by that amount if that interface is down, and decrements are cumulative. If you do not set an explicit decrement value, then the value is decreased by 10 for each interface that goes down, and decrements are cumulative.

The HSRP behavior with this configuration is:

- 0 interfaces down = no decrease (priority is 110)
- 1 interface down = decrease by 10 (priority becomes100)
- 2 interfaces down = decrease by 10 (priority becomes 90)

4. Multiple HSRP Groups

The multiple HSRP (MHSRP) groups feature was added in Cisco IOS latest releases. This feature further enables redundancy and load-sharing within networks, and allows redundant Core switch to be more fully utilized. While a Core switch is actively forwarding traffic for one HSRP group, it can be in standby or in the listen state for another group.

5. Configurable MAC Address

Normally when HSRP is used to help end stations locate the first hop gateway for IP routing. The end stations are configured with a default gateway. However, HSRP can provide first hop redundancy for other protocols. Some protocols, such as Advanced Peer-to-Peer Networking (APPN), use the MAC address to identify the first hop for routing purposes.

In this case, it is often necessary to be able to specify the virtual MAC address using the standby mac-address command. The virtual IP address is unimportant for these protocols. The actual syntax of the command is **standby [group] mac-address**

6. Syslog Support

Support for syslog messaging for HSRP information was added in Cisco IOS releases. This feature allows for more efficient logging and tracking of the current active and standby Core switches on syslog servers.

7. HSRP Debugging

The HSRP debugging command is relatively simple. To enable HSRP debugging, would simply be the debug standby command, which enabled output of HSRP state and packet information for all standby groups on all interfaces.

A debug condition that allows the output from the **standby debug** command to be filtered based upon interface and group number. The command utilizes the **debug condition** as follows: **debug condition standby** *interface group***.** The interface you specify must be a valid interface capable of supporting HSRP. The group can be any group (0 - 255).

8. Authentication

The HSRP authentication feature consists of a shared clear-text key contained within the HSRP packets. This feature prevents the lower priority Core switch from learning the standby IP address and standby timer values from the higher priority Core switch.

To configure the HSRP authentication string, use the standby authentication *string* command.

9. IP Redundancy

HSRP provides stateless redundancy for IP routing. HSRP by itself is limited to maintaining its own state. It assumes that each Core switch builds and maintains its own routing tables independently of other Multi-layered switches. The IP redundancy feature provides a mechanism that allows HSRP to provide a service to client applications so that they can implement stateful failover.

EtherChannel Protocol

Link Aggregation Control Protocol (LACP) and Port Aggregation Protocol (PAgP) EtherChannels have automatic configuration with either Port Aggregation Protocol (PAgP) or Link Aggregation Control Protocol (LACP). Configuring EtherChannels manually can be done as well. PAgP is a Cisco-proprietary protocol that can only run on Cisco switches and on those switches that licensed vendors license to support PAgP.

IEEE 802.3ad defines LACP. LACP allows Cisco switches to manage Ethernet channels between switches that conform to the 802.3ad protocol. You can configure up to 16 ports to form a channel. Eight of the ports are in active mode and the other eight are in standby mode. When any of the active ports fail, a standby port becomes active. Standby mode works only for LACP, not for PAgP.

By using any one of these protocols, a switch learns the identity of partners able to support either PAgP or LACP and learns the capabilities of each interface. The switch then dynamically groups interfaces with similar configurations into a single logical link (channel or aggregate port); the switch bases these interface groups on hardware, administrative, and port parameter constraints.

For example, PAgP groups the interfaces with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After PAgP groups the links into an EtherChannel, PAgP adds the group to the spanning tree as a single switch port.

However, the ITIT university network will be implementing the PAgP mode in its configuration of Ether Channeling throughout the network switches to ensure high end reliability and efficiency in data communication.

PAgP Modes

This are the most listed user-configurable EtherChannel modes for the **channel-group** interface configuration command. Switch interfaces exchange PAgP packets only with partner interfaces with the auto or desirable mode configuration.

- auto—Places an interface into a passive negotiation state, in which the interface responds to PAgP packets that the interface receives but does not start PAgP packet negotiation. This setting minimizes the transmission of PAgP packets.
- desirable—Places an interface into an active negotiation state, in which the interface starts negotiations with other interfaces through the send of PAgP packets.

- on—Forces the interface into an EtherChannel without PAgP or LACP. With the on mode, a usable EtherChannel exists only when an interface group in the on mode has a connection to another interface group in the on mode.

PAgP Restrictions

PAgP aids in the automatic creation of FEC links. PAgP packets transmit between FEC-capable ports to negotiate the formation of a channel. Some restrictions have been deliberately introduced into PAgP. The restrictions are:

- PAgP does not form a bundle on ports with configuration for dynamic VLANs. PAgP requires that all ports in the channel belong to the same VLAN or that the ports have trunk port configurations. When a bundle already exists and you modify the VLAN of a port, all ports in the bundle change to match that VLAN.

- PAgP does not group ports that operate at different speeds or port duplex. If you change speed and duplex when a bundle exists, PAgP changes the port speed and duplex for all ports in the bundle.

- PAgP modes are `off`, `auto`, `desirable`, and `on`. Only the combinations `auto-desirable`, `desirable-desirable`, and `on-on` allow the formation of a channel. If a device on one side of the channel does not support PAgP, such as a router, the device on the other side must have PAgP set to `on`. All Catalyst switches that run CatOS support PAgP protocol channel negotiation.

Dynamic Trunking Protocol (DTP)

There are different types of trunking protocols. If a port can become a trunk, it may also have the ability to trunk automatically. In some cases, the port may even be able to negotiate what type of trunking to use on the port. This ability to negotiate the trunking method with the other device has the name Dynamic Trunking Protocol (DTP).

Spanning-Tress Protocol - RPVST+

Background Information

802.1D Spanning Tree Protocol (STP) has a drawback of slow convergence. Cisco Catalyst switches support three types of STPs, which are PVST+, rapid-PVST+ and MST. Rapid-PVST+ is based on IEEE 802.1w standard and has a faster convergence than 802.1D. RSTP (IEEE 802.1w) natively includes most of the Cisco proprietary enhancements to the 802.1D Spanning Tree, such as BackboneFast and UplinkFast. Rapid-PVST+ has these unique features:

- Uses Bridge Protocol Data Unit (BPDU) version 2 which is backward compatible with the 802.1D STP, which uses BPDU version 0.

- All the switches generate BPDUs and send out on all the ports every 2 seconds, whereas in 802.1D STP only the root bridge sends the configuration BPDUs.
- Port Roles—Root port, designated port, alternate port and backup port.
- Port States—Discarding, Learning, and Forwarding.
- Port Types—Edge Port (PortFast), Point-to-Point and Shared port.

Rapid-PVST uses RSTP to provide faster convergence. When any RSTP port receives legacy 802.1D BPDU, it falls back to legacy STP and the inherent fast convergence benefits of 802.1w are lost when it interacts with legacy bridges.

Rapid-PVST+ Migration

Rapid-PVST+ uses the same BPDU format as the 802.1D and it is backward compatible. It is difficult to convert all the switches in the enterprise network at the same time to rapid-PVST+. Because of the backward compatibility, conversion done phase by phase is possible. It is recommended to implement the changes in the scheduled maintenance window because the spanning tree reconfiguration disrupts the traffic flow.

Spanning Tree UplinkFast and BackboneFast features are PVST+ features. These are disabled when enabling rapid-PVST+ because those features are built within rapid-PVST+. Therefore, during the migration these commands must be removed. The configuration of the features such as PortFast, BPDUguard, BPDUfilter, root guard, and loopguard are applicable in rapid-PVST+ mode as well

RADIUS (Remote Authentication Dial-In User Service)

RADIUS server is implemented in the server room in 2nd floor. It provides centralized authentication, authorization and accounting (AAA) services for users who connect and use the network service
To authenticate users for wireless access we need to implement RADIUS technology. In order to do this we need to implement two server roles such as-

- Active Directory Certificate Services (ADCS)
- Network Policy and Access Services (NPAS)

DNS (Domain Name System)

DNS is configured in DNS server, which is in the server room in 3nd floor. All the hosts in this network are assumed to be connected to domain. So, each hosts (workstations) have their unique domain name. So, inside users can use the specific domain name to connect to each host remotely. But computers cannot understand the name. It should be converted to numbers called

IP address. So, DNS server maintain the map of domain name of each host to its corresponding IP address. Thus, management and complexity of network can be reduced.

The Domain Name System is a hierarchical and decentralized naming system for computers, services, or other resources connected to the Internet or a private network. It associates various information with domain names assigned to each of the participating entities.

DHCP (Dynamic Host Configuration Protocol)

DHCP service is installed in the DHCP server which resides in server room. IP address pool for different VLAN will be created in DHCP server. So DHCP server dynamically assign the IP address to the hosts in the network. Static IP address that will be used with in the VLAN can be removed from the IP address pool (excluded address) in DHCP server. Main advantage of using this protocol is reliable IP address configuration to hosts (reduce configuration errors caused by manual IP assignment), and reduced network administration (centralized management

Implementation

Automating Network Tasks using Network Programming Scripts

In the original network within the college campus, complete with all the core & access layer switches, firewalls, servers, and end hosts, no programming was implemented. This was a core objective of the network system upgrade project. Programming in the field of networking allows administrators to write simple scripts that can be used to automate particular tasks, this can vastly improve the current capabilities of the task done manually, reduce the error margin while widening the scale of the task, and also most importantly save labor time.
The below sections report the network automation scripts that were implemented as part of the network upgrade project and provides proof of concepts to show the working of the scripts. The scripting language used was Python.

Automating VLAN creation

The below script was made to automate the creation of VLANs on the core and access layer switches. The script uses SSH to invoke a shell in the device to pass commands.
The default VLAN database before running the script is show below.

```
ACC-SW1>
ACC-SW1>en
ACC-SW1#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gi0/0, Gi0/1, Gi0/2, Gi0/3
                                                Gi1/0, Gi1/1
1002 fddi-default                     act/unsup
1003 trcrf-default                    act/unsup
1004 fddinet-default                  act/unsup
1005 trbrf-default                    act/unsup
ACC-SW1#
```

Figure 6.0

Note that the switch only contains the default VLANs 1, 1002-1005 (used for Fibre Distributed Data Interfaces and Token Rings). In this proof of concept, the script will be used to create the VLANs ranging from 2 to 10.

The script and its execution is shown below,

```
import paramiko
import time
from sys import argv

script, ip = argv

ip_address = ip
username = raw_input("Enter username: ")
password = raw_input("Enter password: ")

ssh_client = paramiko.SSHClient()
ssh_client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
ssh_client.connect(hostname=ip_address,username=username,password=password)

print ("SSH connection successful to ", ip_address)

remote_connection = ssh_client.invoke_shell()
remote_connection.send("configure terminal\n")

for x in range (2,11):
        print "Creating VLAN " + str(x)
        remote_connection.send("vlan " + str(x) + "\n")
        v_name = raw_input("Enter VLAN name: ")
        remote_connection.send("name " + v_name + "\n")
        time.sleep(0.5)

remote_connection.send("end\n")
```

Figure 6.1

```
root@NetworkAutomation-1:~# python vlanscript.py 172.16.11.8
Enter username: itit-admin
Enter password:
('SSH connection successful to ', '172.16.11.8')
Creating VLAN 2
Enter VLAN name: TEST-VLAN2
Creating VLAN 3
Enter VLAN name: TEST-VLAN3
Creating VLAN 4
Enter VLAN name: TEST-VLAN4
Creating VLAN 5
Enter VLAN name: TEST-VLAN5
Creating VLAN 6
Enter VLAN name: TEST-VLAN6
Creating VLAN 7
Enter VLAN name: TEST-VLAN7
Creating VLAN 8
Enter VLAN name: TEST-VLAN8
Creating VLAN 9
Enter VLAN name: TEST-VLAN9
Creating VLAN 10
Enter VLAN name: TEST-VLAN10
```

Figure 6.2

```
ACC-SW1#show vlan br

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gi0/0, Gi0/1, Gi0/2, Gi0/3
                                                Gi1/0, Gi1/1
2    TEST-VLAN2                       active
3    TEST-VLAN3                       active
4    TEST-VLAN4                       active
5    TEST-VLAN5                       active
6    TEST-VLAN6                       active
7    TEST-VLAN7                       active
8    TEST-VLAN8                       active
9    TEST-VLAN9                       active
10   TEST-VLAN10                      active
1002 fddi-default                     act/unsup
1003 trcrf-default                    act/unsup
1004 fddinet-default                  act/unsup
1005 trbrf-default                    act/unsup
```

Figure 6.2

The switch terminal output is also displayed on the screen for confirmation.

```
ACC-SW1(config)#vlan 2
ACC-SW1(config-vlan)#name TEST-VLAN2
ACC-SW1(config-vlan)#vlan 3
ACC-SW1(config-vlan)#name TEST-VLAN3
ACC-SW1(config-vlan)#vlan 4
ACC-SW1(config-vlan)#name TEST-VLAN4
ACC-SW1(config-vlan)#vlan 5
ACC-SW1(config-vlan)#name TEST-VLAN5
ACC-SW1(config-vlan)#vlan 6
ACC-SW1(config-vlan)#name TEST-VLAN6
ACC-SW1(config-vlan)#vlan 7
ACC-SW1(config-vlan)#name TEST-VLAN7
ACC-SW1(config-vlan)#vlan 8
ACC-SW1(config-vlan)#name TEST-VLAN8
ACC-SW1(config-vlan)#vlan 9
ACC-SW1(config-vlan)#name TEST-VLAN9
ACC-SW1(config-vlan)#vlan 10
ACC-SW1(config-vlan)#name TEST-VLAN10
ACC-SW1(config-vlan)#end
ACC-SW1#
root@NetworkAutomation-1:~#
```

Figure 6.3

This script can drastically reduce the time spent on making VLANs. Instead of manually adding multiple VLANs to the tens of switches one by one this single script can be run against all the available and needed switches to create as many VLANs as the administrator requires concurrently.

Automating Port Configurations

The same script can be modified to automate the configuration of the switch port-security. All unused switch ports are moved to black hole VLAN and shutdown. The script is shown below.

```
#!/usr/bin/env python

import paramiko
import time
from sys import argv

script, ip = argv

ip_address = ip
username = raw_input("Enter username: ")
password = raw_input("Enter password: ")

ssh_client = paramiko.SSHClient()
ssh_client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
ssh_client.connect(hostname=ip_address,username=username,password=password)

print ("SSH connection successful to ", ip_address)

remote_connection = ssh_client.invoke_shell()
remote_connection.send("configure terminal\n")

print "Implementing Port Security " + str(x)
remote_connection.send("interface range GigabitEthernet  2/0-1\n")
remote_connection.send("shutdown\n")
```

Figure 6.4

```
print ("Shutting down unused port\n")

for x in range (0,4):
        remote_connection.send("interface range GigabitEthernet 0/" + str(x) + "\n")
        remote_connection.send("switchport mode access\n")
        remote_connection.send("switchport port-security\n")
        remote_connection.send("switchport port-security mac-address sticky\n")
        remote_connection.send("switchport port_security maximum 2\n")
        remote_connection.send("switchport port-security violation shutdown\n")
        time.sleep(0.5)

remote_connection.send("end\n")

time.sleep(1)
output = remote_connection.recv(65535)
```

Figure 6.5

Automating Switch IOS backups

The configuration file of the network devices must be backed up in a timely manner in order to ensure security, availability, and reliability. The script writes the current configuration file to a new file and can be set to run periodically using cron jobs.

```
import datetime
import paramiko
import time
from sys import argv

script, ip = argv

ip_address = ip
username = raw_input("Enter username: ")
password = raw_input("Enter password: ")

ssh_client = paramiko.SSHClient()
ssh_client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
ssh_client.connect(hostname=ip_address,username=username,password=password)

print ("SSH connection successful to ", ip_address)

remote_connection = ssh_client.invoke_shell()
remote_connection.send("terminal length 0\n")

print ("Getting running configuration...\n")
remote_connection.send("show running-config\n")
time.sleep(10)
remote_connection.send("exit\n")
time.sleep(0.5)
output = remote_connection.recv(65535)

file = 'SW '+str(ip)+" " +  str(datetime.date.today().isoformat())

op_file = open(file,'w')
op_file.write(output.decode("utf-8"))
print ("Writing configuration to file...\n")
op_file.close()

ssh_client.close
```

Figure 6.6

```
root@NetworkAutomation-1:~# ls
backupconf.py  portsec.py
root@NetworkAutomation-1:~# python backupconf.py 172.16.11.8
Enter username: itit-admin
Enter password:
('SSH connection successful to ', '172.16.11.8')
Getting running configuration...

Writing configuration to file...

root@NetworkAutomation-1:~# ls
'SW 172.16.11.8 2020-08-30'  backupconf.py  portsec.py
```

Figure 6.7

```
!
boot-start-marker
boot-end-marker
!
!
!
username itit-admin privilege 15 password 0 cisco
no aaa new-model
!
!
!
!
!
vtp domain CISCO-vIOS
vtp mode transparent
!
!
!
ip domain-name itit.com
ip cef
no ipv6 cef
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
vlan 10
 name TEST-VLAN10
!
vlan 20
```

Figure 6.8

Network Implementation

Given the Proposed updated of the network infrastructure and the additional departmental updates, the network configuration was amended according to the upgrade requirements starting with the creation of new VLANs and assigned Ip addresses.
Moreover, the re-configuration of the Routing protocols and their respective Ip addressing are further addressed in the Network implementation segment.

Configuration of CORE 1 Switch

Step 1: change the default hostname of the primary core switch as **CORE 1**

```
Switch(config)#
Switch(config)#hostname CORE1
CORE1(config)#enable secret cisco
CORE1(config)#
```

Figure 7.0

Step 2: Create the VLANs in CORE 1 L3 Switch specific to each department within the university

```
Switch(config)#
Switch(config)#
Switch(config)#vlan 99
Switch(config-vlan)#name NATIVE
Switch(config-vlan)#
Switch(config-vlan)#vlan 3
Switch(config-vlan)#name Student_Wifi
Switch(config-vlan)#vlan 4
Switch(config-vlan)#name Labs
Switch(config-vlan)#vlan 5
Switch(config-vlan)#name Guest_Wifi
Switch(config-vlan)#vlan 6
Switch(config-vlan)#name Academic_Staff
Switch(config-vlan)#vlan 7
Switch(config-vlan)#name CCTV_surveillance
Switch(config-vlan)#vlan 8
Switch(config-vlan)#name Exams
Switch(config-vlan)#vlan 9
Switch(config-vlan)#name Finace
Switch(config-vlan)#vlan 10
Switch(config-vlan)#name AP_management
Switch(config-vlan)#vlan 11
Switch(config-vlan)#name HR
Switch(config-vlan)#vlan 12
Switch(config-vlan)#name Internal_servers
Switch(config-vlan)#vlan 13
Switch(config-vlan)#name Device_management
Switch(config-vlan)#vlan 14
Switch(config-vlan)#name IT_Dept
Switch(config-vlan)#vlan 15
Switch(config-vlan)#name Marketing
Switch(config-vlan)#vlan 16
Switch(config-vlan)#name Staff
Switch(config-vlan)#
Switch(config-vlan)#
```

Figure 7.1

Step 3: Assign the Ip address in respective of each VLAN according to the host Ip address range provided to each VLAN

```
Switch(config)#int vlan 3
Switch(config-if)#ip add 172.16.0.2 255.255.248.0
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#
Switch(config)#int vlan 4
Switch(config-if)#ip add 172.16.8.2 255.255.254.0
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 5
Switch(config-if)#ip add 172.16.10.2 255.255.255.128
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 6
Switch(config-if)#ip add 172.16.11.2 255.255.255.192
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 7
Switch(config-if)#ip add 172.16.11.66 255.255.255.192
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 8
Switch(config-if)#ip add 172.16.11.194 255.255.255.224
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 9
Switch(config-if)#ip add 172.16.11.226 255.255.255.240
Switch(config-if)#no shut
Switch(config-if)#exi
```

Figure 7.2

```
Switch(config)#int vlan 10
Switch(config-if)#ip add 172.16.11.130 255.255.255.192
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 11
Switch(config-if)#ip add 172.16.12.2  255.255.255.240
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 12
Switch(config-if)#ip add 172.16.12.18  255.255.255.240
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
```

Figure 7.3

```
Switch(config)#int vlan 13
Switch(config-if)#ip add 172.16.11.242  255.255.255.240
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 14
Switch(config-if)#ip add 172.16.12.34  255.255.255.240
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
```

Figure 7.4

```
Switch(config)#
Switch(config)#int vlan 15
Switch(config-if)#ip add 172.16.12.49 255.255.255.240
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
Switch(config)#int vlan 16
Switch(config-if)#ip add 172.16.10.130  255.255.255.128
Switch(config-if)#no shut
Switch(config-if)#exi
Switch(config)#
```

Figure 7.5

Step 4: After the configurations of the required VLANs, Verify the VLAN configuration done in CORE 1 (L3) switch using the command " **show VLAN**"

```
Switch(config)#
Switch(config)#do sh vlan

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active    Gi1/0/1, Gi1/0/2, Gi1/0/3
                                                 Gi1/0/4, Gi1/0/5, Gi1/0/6
                                                 Gi1/0/7, Gi1/0/8, Gi1/0/9
                                                 Gi1/0/10, Gi1/0/11, Gi1/0/12
                                                 Gi1/0/13, Gi1/0/14, Gi1/0/15
                                                 Gi1/0/16, Gi1/0/17, Gi1/0/18
                                                 Gi1/0/19, Gi1/0/20, Gi1/0/21
                                                 Gi1/0/22, Gi1/0/23, Gi1/0/24
                                                 Gi1/1/1, Gi1/1/2, Gi1/1/3
                                                 Gi1/1/4
3    Student_Wifi                      active
4    Labs                              active
5    Guest_Wifi                        active
6    Academic_Staff                    active
7    CCTV_surveillance                 active
8    Exams                             active
9    Finace                            active
10   AP_management                     active
11   HR                                active
12   Internal_servers                  active
13   Device_management                 active
14   IT_Dept                           active
15   Marketing                         active
16   Staff                             active
99   NATIVE                            active
1002 fddi-default                      act/unsup
1003 token-ring-default                act/unsup
1004 fddinet-default                   act/unsup
1005 trnet-default                     act/unsup
```

Figure 7.6

Step 5: Creating EtherChannels between the **CORE 1** switch and **CORE 2** switch
As EtherChannel is a is a port link aggregation technology, it allows grouping of several physical Ethernet links to create one logical Ethernet link for the purpose of providing fault-tolerance and high-speed links between routers.

```
CORE1(config)#
CORE1(config)#int port
CORE1(config)#int port-channel 40
CORE1(config-if)#sw mode tr
CORE1(config-if)#sw mode trunk
CORE1(config-if)#exit
CORE1(config)#
```

Figure 7.7

```
CORE1(config)#
CORE1(config)# int range g1/0/9-10
CORE1(config-if-range)#channel-group 40 mode active
CORE1(config-if-range)#no shut
CORE1(config-if-range)#exit
CORE1(config)#
CORE1(config)#
```

Figure 7.8

Step 6: Verification of the EtherChannel configuration is confirmed with **"show EtherChannel summery"** command
As seen, the EtherChannels are successfully formed, with indication to the protocols (LACP) and the specific ports used

```
CORE1(config)#
CORE1(config)#do sh etherchannel sum
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port

        A - formed by Auto LAG


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
40     Po40(SU)        LACP      Gi1/0/9(P)  Gi1/0/10(P)
```

Figure 7.9

Step 7: Enable the IP routing for the connectivity between different subnetworks on the CORE 1 switch

```
CORE1(config)#
CORE1(config)#ip routing
CORE1(config)#
```

Figure 7.10

Step8 : Verify the Ip's Assigned to each individual VLAN specifically and to verify the connectivity of the ports and interfaces using **"show Ip interface brief "**

```
CORE1(config)#do sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
Vlan1                  unassigned      YES unset  administratively down down
Vlan3                  172.16.0.2      YES manual up                    up
Vlan4                  172.16.8.2      YES manual up                    up
Vlan5                  172.16.10.2     YES manual up                    up
Vlan6                  172.16.11.2     YES manual up                    up
Vlan7                  172.16.11.66    YES manual up                    up
Vlan8                  172.16.11.194   YES manual up                    up
Vlan9                  172.16.11.226   YES manual up                    up
Vlan10                 172.16.11.130   YES manual up                    up
Vlan11                 172.16.12.2     YES manual up                    up
Vlan12                 172.16.12.18    YES manual up                    up
Vlan13                 172.16.11.242   YES manual up                    up
Vlan14                 172.16.12.34    YES manual up                    up
Vlan15                 172.16.12.49    YES manual up                    up
Vlan16                 172.16.10.130   YES manual up                    up
GigabitEthernet0/0     unassigned      YES unset  down                  down
GigabitEthernet1/0/1   unassigned      YES unset  down                  down
GigabitEthernet1/0/2   unassigned      YES unset  down                  down
GigabitEthernet1/0/3   unassigned      YES unset  down                  down
GigabitEthernet1/0/4   unassigned      YES unset  down                  down
GigabitEthernet1/0/5   unassigned      YES unset  down                  down
GigabitEthernet1/0/6   unassigned      YES unset  down                  down
GigabitEthernet1/0/7   unassigned      YES unset  down                  down
GigabitEthernet1/0/8   unassigned      YES unset  down                  down
GigabitEthernet1/0/9   unassigned      YES unset  up                    up
GigabitEthernet1/0/10  unassigned      YES unset  up                    up
GigabitEthernet1/0/11  unassigned      YES unset  down                  down
GigabitEthernet1/0/12  unassigned      YES unset  down                  down
GigabitEthernet1/0/13  unassigned      YES unset  down                  down
GigabitEthernet1/0/14  unassigned      YES unset  down                  down
GigabitEthernet1/0/15  unassigned      YES unset  down                  down
GigabitEthernet1/0/16  unassigned      YES unset  down                  down
GigabitEthernet1/0/17  unassigned      YES unset  down                  down
GigabitEthernet1/0/18  unassigned      YES unset  down                  down
GigabitEthernet1/0/19  unassigned      YES unset  down                  down
GigabitEthernet1/0/20  unassigned      YES unset  down                  down
```

Figure 7.11

Step 8: Configure the HSRP with the VLAN interfaces and HSRP helper addresses. Here the CORE 1 router is used as the Active router, Configured with a higher pre-emptive priority (110)

```
CORE1(config)#int vlan 3
CORE1(config-if)#standby 3 ip 172.16.0.1
CORE1(config-if)#standby 3 priority 110
CORE1(config-if)#standby 3 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
CORE1(config)#int vlan 4
CORE1(config-if)#standby 4 ip 172.16.8.1
CORE1(config-if)#standby 4 priority 110
CORE1(config-if)#standby 4 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
CORE1(config)#int vlan 5
CORE1(config-if)#standby 5 ip 172.16.10.1
CORE1(config-if)#standby 5 priority 110
CORE1(config-if)#standby 5 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 6
CORE1(config-if)#standby 6 ip  172.16.11.1
CORE1(config-if)#standby 6 priority 110
CORE1(config-if)#standby 6 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 7
CORE1(config-if)#standby 7 ip 172.16.11.65
CORE1(config-if)#standby 7 priority 110
CORE1(config-if)#standby 7 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
CORE1(config)#int vlan 8
CORE1(config-if)#standby 8 ip 172.16.11.193
% Address 172.16.11.193 in group 0
CORE1(config-if)#standby 8 priority 110
CORE1(config-if)#standby 8 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
```

Figure 7.12

```
CORE1(config)#
CORE1(config)#int vlan 9
CORE1(config-if)#standby 9 ip  172.16.11.225
% Address 172.16.11.225 in group 0
CORE1(config-if)#standby 9 priority 110
CORE1(config-if)#standby 9 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 10
CORE1(config-if)#standby  10 ip 172.16.11.129
% Address 172.16.11.129 in group 0
CORE1(config-if)#standby 10 priority 110
CORE1(config-if)#standby 10 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 11
CORE1(config-if)#standby 11 ip 172.16.12.1
CORE1(config-if)#standby 11 priority 110
CORE1(config-if)#standby 11 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 12
CORE1(config-if)#standby 12 ip 172.16.12.17
CORE1(config-if)#standby 12 priority 110
CORE1(config-if)#standby 12 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
CORE1(config)#int vlan 13
CORE1(config-if)#standby 13 ip 172.16.11.241
CORE1(config-if)#standby 13 priority 110
CORE1(config-if)#standby 13 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 14
CORE1(config-if)#standby 14 ip 172.16.12.33
CORE1(config-if)#standby 14 priority 110
CORE1(config-if)#standby 14 preempt
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 15
CORE1(config-if)#standby 15 ip 172.16.12.49
CORE1(config-if)#standby 15 priority 110
CORE1(config-if)#standby 15 preempt
CORE1(config-if)#exi
```

Figure 7.13

```
CORE1(config)#int vlan 16
CORE1(config-if)#standby 16 ip 172.16.10.129
CORE1(config-if)#standby 16 priority 110
CORE1(config-if)#standby 16 preempt
CORE1(config-if)#exi
CORE1(config)#
```

Figure 7.14

Step 9: Configure the HSRP routing protocol Helper address to the VLANs configured as to route the traffic to the secondary router (standby) should the primary Core Router fail

```
CORE1(config)#
CORE1(config)#int vlan 3
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
CORE1(config)#
CORE1(config)#int vlan 4
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
CORE1(config)#int vlan 5
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
CORE1(config)#int vlan 6
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 7
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#int vlan 8
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#int vlan 9
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#int vlan 10
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 11
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 12
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#
```

Figure 7.15

```
CORE1(config)#int vlan 13
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 14
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 15
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
CORE1(config)#
CORE1(config)#int vlan 16
CORE1(config-if)#ip helper-address 172.16.12.21
CORE1(config-if)#exi
```

Figure 7.16

Step 10 : Verify the HSRP configuration in CORE 1 L3 Switch using the **" show Standby"** Command

```
CORE1>en
CORE1#sh standby
Vlan3 - Group 3
  State is Active
    2 state changes, last state change 00:39:39
  Virtual IP address is 172.16.0.1
  Active virtual MAC address is 0000.0c07.ac03 (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac03 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 0.816 secs
  Preemption enabled
  Active router is local
  Standby router is 172.16.0.3, priority 110 (expires in 10.784 sec)
  Priority 110 (configured 110)
  Group name is "hsrp-Vl3-3" (default)
Vlan4 - Group 4
  State is Active
    2 state changes, last state change 00:39:50
  Virtual IP address is 172.16.8.1
  Active virtual MAC address is 0000.0c07.ac04 (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac04 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 0.608 secs
  Preemption enabled
  Active router is local
  Standby router is 172.16.8.3, priority 110 (expires in 9.360 sec)
  Priority 110 (configured 110)
  Group name is "hsrp-Vl4-4" (default)
Vlan5 - Group 5
  State is Active
    2 state changes, last state change 00:39:49
  Virtual IP address is 172.16.10.1
  Active virtual MAC address is 0000.0c07.ac05 (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac05 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 0.496 secs
  Preemption enabled
  Active router is local
  Standby router is 172.16.10.3, priority 110 (expires in 8.384 sec)
  Priority 110 (configured 110)
  Group name is "hsrp-Vl5-5" (default)
```

Figure 7.17

```
Vlan6 - Group 6
  State is Active
    2 state changes, last state change 00:39:51
  Virtual IP address is 172.16.11.1
  Active virtual MAC address is 0000.0c07.ac06 (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac06 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.920 secs
  Preemption enabled
  Active router is local
  Standby router is 172.16.11.3, priority 110 (expires in 10.832 sec)
  Priority 110 (configured 110)
  Group name is "hsrp-Vl6-6" (default)
Vlan7 - Group 7
  State is Active
    2 state changes, last state change 00:39:51
  Virtual IP address is 172.16.11.65
  Active virtual MAC address is 0000.0c07.ac07 (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac07 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.696 secs
  Preemption enabled
  Active router is local
  Standby router is 172.16.11.67, priority 110 (expires in 11.616 sec)
  Priority 110 (configured 110)
  Group name is "hsrp-Vl7-7" (default)
Vlan8 - Group 0
  State is Active
    2 state changes, last state change 00:39:49
  Virtual IP address is 172.16.11.193
  Active virtual MAC address is 0000.0c07.ac00 (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.928 secs
  Preemption disabled
  Active router is local
  Standby router is 172.16.11.195, priority 100 (expires in 10.080 sec)
  Priority 100 (default 100)
  Group name is "hsrp-Vl8-0" (default)
```

Figure 7.18

```
Vlan9 - Group 0
  State is Active
    2 state changes, last state change 00:39:48
  Virtual IP address is 172.16.11.225
  Active virtual MAC address is 0000.0c07.ac00 (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.496 secs
  Preemption disabled
  Active router is local
  Standby router is 172.16.11.227, priority 100 (expires in 10.480 sec)
  Priority 100 (default 100)
  Group name is "hsrp-Vl9-0" (default)
Vlan10 - Group 0
  State is Active
    2 state changes, last state change 00:39:49
  Virtual IP address is 172.16.11.129
  Active virtual MAC address is 0000.0c07.ac00 (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.016 secs
  Preemption disabled
  Active router is local
  Standby router is 172.16.11.131, priority 100 (expires in 10.688 sec)
  Priority 100 (default 100)
  Group name is "hsrp-Vl10-0" (default)
Vlan11 - Group 11
  State is Active
    2 state changes, last state change 00:39:39
  Virtual IP address is 172.16.12.1
  Active virtual MAC address is 0000.0c07.ac0b (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac0b (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.264 secs
  Preemption enabled
  Active router is local
  Standby router is 172.16.12.3, priority 110 (expires in 9.536 sec)
  Priority 110 (configured 110)
  Group name is "hsrp-Vl11-11" (default)
```

Figure 7.19

Note : As seen in the above the HSRP protocol is successfully enabled with preemptive and hello timers synchronized as well

Configuration of CORE 2 Switch

Step 1 : change the default hostname of the primary core switch as **CORE 2**

```
Switch(config)#hostname CORE2
CORE2(config)#
CORE2(config)#enable secret cisco
CORE2(config)#
```

Figure 7.20

Step 2: Create the VLAN Table in CORE 2 L3 Switch

```
CORE2(config)#
CORE2(config)#
CORE2(config)#vlan 99
CORE2(config-vlan)#name NATIVE
CORE2(config-vlan)#
CORE2(config-vlan)#vlan 3
CORE2(config-vlan)#name Student_Wifi
CORE2(config-vlan)#vlan 4
CORE2(config-vlan)#name Labs
CORE2(config-vlan)#vlan 5
CORE2(config-vlan)#name Guest_Wifi
CORE2(config-vlan)#vlan 6
CORE2(config-vlan)#name Academic_Staff
CORE2(config-vlan)#vlan 7
CORE2(config-vlan)#name CCTV_surveillance
CORE2(config-vlan)#vlan 8
CORE2(config-vlan)#name Exams
CORE2(config-vlan)#vlan 9
CORE2(config-vlan)#name Finace
CORE2(config-vlan)#vlan 10
CORE2(config-vlan)#name AP_management
CORE2(config-vlan)#vlan 11
CORE2(config-vlan)#name HR
CORE2(config-vlan)#vlan 12
CORE2(config-vlan)#name Internal_servers
CORE2(config-vlan)#vlan 13
CORE2(config-vlan)#name Device_management
CORE2(config-vlan)#vlan 14
CORE2(config-vlan)#name IT_Dept
CORE2(config-vlan)#vlan 15
CORE2(config-vlan)#name Marketing
CORE2(config-vlan)#vlan 16
CORE2(config-vlan)#name Staff
CORE2(config-vlan)#
CORE2(config-vlan)#
```

Figure 7.21

Step 3: Assign the Ip address for respective each VLAN

```
CORE2(config)#int vlan 3
CORE2(config-if)#ip add 172.16.0.3 255.255.248.0
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#
CORE2(config)#int vlan 4
CORE2(config-if)#ip add 172.16.8.3 255.255.254.0
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 5
CORE2(config-if)#ip add 172.16.10.3 255.255.255.128
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 6
CORE2(config-if)#ip add 172.16.11.3 255.255.255.192
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 7
CORE2(config-if)#ip add 172.16.11.67 255.255.255.192
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 8
CORE2(config-if)#ip add 172.16.11.195 255.255.255.224
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 9
CORE2(config-if)#ip add 172.16.11.227 255.255.255.240
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#
```

Figure 7.22

```
CORE2(config)#
CORE2(config)#int vlan 10
CORE2(config-if)#ip add 172.16.11.131 255.255.255.192
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 11
CORE2(config-if)#ip add 172.16.12.3  255.255.255.240
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
```

Figure 7.23

```
CORE2(config)#
CORE2(config)#int vlan 12
CORE2(config-if)#ip add 172.16.12.19  255.255.255.240
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#
CORE2(config)#int vlan 13
CORE2(config-if)#ip add 172.16.11.243  255.255.255.240
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 14
CORE2(config-if)#ip add 172.16.12.35  255.255.255.240
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#
CORE2(config)#int vlan 15
CORE2(config-if)#ip add 172.16.12.50 255.255.255.240
CORE2(config-if)#no shut
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 16
CORE2(config-if)#ip add 172.16.10.131  255.255.255.128
CORE2(config-if)#no shut
CORE2(config-if)#exi
```

Figure 7.24

Step 4 : Verify the VLAN configuration done in **CORE 2** L3 switch using the command " show VLAN"

```
CORE2(config)#
CORE2(config)#do sh vlan

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active    Gi1/0/1, Gi1/0/2, Gi1/0/3
                                                 Gi1/0/4, Gi1/0/5, Gi1/0/6
                                                 Gi1/0/7, Gi1/0/8, Gi1/0/9
                                                 Gi1/0/10, Gi1/0/11, Gi1/0/12
                                                 Gi1/0/13, Gi1/0/14, Gi1/0/15
                                                 Gi1/0/16, Gi1/0/17, Gi1/0/18
                                                 Gi1/0/19, Gi1/0/20, Gi1/0/21
                                                 Gi1/0/22, Gi1/0/23, Gi1/0/24
                                                 Gi1/1/1, Gi1/1/2, Gi1/1/3
                                                 Gi1/1/4
3    Student_Wifi                      active
4    Labs                              active
5    Guest_Wifi                        active
6    Academic_Staff                    active
7    CCTV_surveillance                 active
8    Exams                             active
9    Finace                            active
10   AP_management                     active
11   HR                                active
12   Internal_servers                  active
13   Device_management                 active
14   IT_Dept                           active
15   Marketing                         active
16   Staff                             active
99   NATIVE                            active
1002 fddi-default                      act/unsup
1003 token-ring-default                act/unsup
1004 fddinet-default                   act/unsup
1005 trnet-default                     act/unsup
```

Figure 7.25

- Verify the Ip's Assigned to eac .N specifically ,using **" show Ip interface brief "**

```
CORE2(config)#
CORE2(config)#
CORE2(config)#do sh ip int br
Interface              IP-Address      OK? Method Status                Protocol
Vlan1                  unassigned      YES unset  administratively down down
Vlan3                  172.16.0.3      YES manual up                    up
Vlan4                  172.16.8.3      YES manual up                    up
Vlan5                  172.16.10.3     YES manual up                    up
Vlan6                  172.16.11.3     YES manual up                    up
Vlan7                  172.16.11.67    YES manual up                    up
Vlan8                  172.16.11.195   YES manual up                    up
Vlan9                  172.16.11.227   YES manual up                    up
Vlan10                 172.16.11.131   YES manual up                    up
Vlan11                 172.16.12.3     YES manual up                    up
Vlan12                 172.16.12.19    YES manual up                    up
Vlan13                 172.16.11.243   YES manual up                    up
Vlan14                 172.16.12.35    YES manual up                    up
Vlan15                 172.16.12.50    YES manual up                    up
Vlan16                 172.16.10.131   YES manual up                    up
GigabitEthernet0/0     unassigned      YES unset  down                  down
GigabitEthernet1/0/1   unassigned      YES unset  down                  down
GigabitEthernet1/0/2   unassigned      YES unset  down                  down
GigabitEthernet1/0/3   unassigned      YES unset  down                  down
GigabitEthernet1/0/4   unassigned      YES unset  down                  down
GigabitEthernet1/0/5   unassigned      YES unset  down                  down
GigabitEthernet1/0/6   unassigned      YES unset  down                  down
GigabitEthernet1/0/7   unassigned      YES unset  down                  down
GigabitEthernet1/0/8   unassigned      YES unset  down                  down
GigabitEthernet1/0/9   unassigned      YES unset  up                    up
GigabitEthernet1/0/10  unassigned      YES unset  up                    up
GigabitEthernet1/0/11  unassigned      YES unset  down                  down
GigabitEthernet1/0/12  unassigned      YES unset  down                  down
GigabitEthernet1/0/13  unassigned      YES unset  down                  down
GigabitEthernet1/0/14  unassigned      YES unset  down                  down
GigabitEthernet1/0/15  unassigned      YES unset  down                  down
GigabitEthernet1/0/16  unassigned      YES unset  down                  down
GigabitEthernet1/0/17  unassigned      YES unset  down                  down
GigabitEthernet1/0/18  unassigned      YES unset  down                  down
```

Step 5: Configure EtherChannel between the **CORE 1** switch and **CORE 2** switch

```
CORE2(config)#
CORE2(config)#int range g1/0/9-10
CORE2(config-if-range)#channel-group 40 mode active
Creating a port-channel interface Port-channel 40
```

Figure 7.27

```
CORE2(config)#
CORE2(config)#int port-channel 40
CORE2(config-if)#sw mo trunk
CORE2(config-if)#exit
CORE2(config)#
```

Figure 7.28

Step 6 : Verification of the EtherChannel configuration

```
CORE2#sh etherchannel sum
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port

        A - formed by Auto LAG


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
40     Po40(SU)        LACP      Gi1/0/9(P)  Gi1/0/10(P)
```

Figure 7.29

Step 7 : Enable the IP routing on the CORE 2 (L3) switch

```
CORE2(config)#
CORE2(config)#ip routing
CORE2(config)#
```

Figure 7.30

Step 8 : Configure the HSRP with the VLAN interfaces and HSRP helper addresses . Here the CORE 2 router is used as the **Standby** router.

```
CORE2(config)#
CORE2(config)#int vlan 3
CORE2(config-if)#standby 3 ip 172.16.0.1
CORE2(config-if)#standby 3 priority 110
CORE2(config-if)#standby 3 preempt
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#
CORE2(config)#int vlan 4
CORE2(config-if)#standby 4 ip 172.16.8.1
CORE2(config-if)#standby 4 priority 110
CORE2(config-if)#standby 4 preempt
CORE2(config-if)#exi
CORE2(config)#
```

Figure 7.31

```
CORE2(config)#
CORE2(config)#int vlan 5
CORE2(config-if)#standby 5 ip 172.16.10.1
CORE2(config-if)#standby 5 priority 110
CORE2(config-if)#standby 5 preempt
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 6
CORE2(config-if)#standby 6 ip  172.16.11.1
CORE2(config-if)#standby 6 priority 110
CORE2(config-if)#standby 6 preempt
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 7
CORE2(config-if)#standby 7 ip 172.16.11.65
CORE2(config-if)#standby 7 priority 110
CORE2(config-if)#standby 7 preempt
CORE2(config-if)#exi
CORE2(config)#
```

Figure 7.32

```
CORE2(config)#int vlan 11
CORE2(config-if)#standby 11 ip 172.16.12.1
CORE2(config-if)#standby 11 priority 110
CORE2(config-if)#standby 11 preempt
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 12
CORE2(config-if)#standby 12 ip 172.16.12.17
CORE2(config-if)#standby 12 priority 110
CORE2(config-if)#standby 12 preempt
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#
CORE2(config)#int vlan 13
CORE2(config-if)#standby 13 ip 172.16.11.241
CORE2(config-if)#standby 13 priority 110
CORE2(config-if)#standby 13 preempt
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 14
CORE2(config-if)#standby 14 ip 172.16.12.33
CORE2(config-if)#standby 14 priority 110
CORE2(config-if)#standby 14 preempt
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 15
CORE2(config-if)#standby 15 ip 172.16.12.49
CORE2(config-if)#standby 15 priority 110
CORE2(config-if)#standby 15 preempt
CORE2(config-if)#exi
CORE2(config)#
CORE2(config)#int vlan 16
CORE2(config-if)#standby 16 ip 172.16.10.129
CORE2(config-if)#standby 16 priority 110
CORE2(config-if)#standby 16 preempt
CORE2(config-if)#exi
CORE2(config)#
```

Figure 7.33

Step 9 : Verify the HSRP Configuration done in CORE 2 router
Using the command **"show Standby"**

```
CORE2(config)#do sh standby
Vlan3 - Group 3
  State is Standby
    1 state change, last state change 00:06:34
  Virtual IP address is 172.16.0.1
  Active virtual MAC address is 0000.0c07.ac03 (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac03 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.144 secs
  Preemption enabled
  Active router is 172.16.0.2, priority 110 (expires in 10.320 sec)
  Standby router is local
  Priority 110 (configured 110)
  Group name is "hsrp-Vl3-3" (default)
Vlan4 - Group 4
  State is Standby
    1 state change, last state change 00:06:33
  Virtual IP address is 172.16.8.1
  Active virtual MAC address is 0000.0c07.ac04 (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac04 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.488 secs
  Preemption enabled
  Active router is 172.16.8.2, priority 110 (expires in 8.368 sec)
  Standby router is local
  Priority 110 (configured 110)
  Group name is "hsrp-Vl4-4" (default)
```

Figure 7.34

```
Vlan5 - Group 5
  State is Standby
    1 state change, last state change 00:06:33
  Virtual IP address is 172.16.10.1
  Active virtual MAC address is 0000.0c07.ac05 (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac05 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.528 secs
  Preemption enabled
  Active router is 172.16.10.2, priority 110 (expires in 10.912 sec)
  Standby router is local
  Priority 110 (configured 110)
  Group name is "hsrp-Vl5-5" (default)
Vlan6 - Group 6
  State is Standby
    1 state change, last state change 00:06:34
  Virtual IP address is 172.16.11.1
  Active virtual MAC address is 0000.0c07.ac06 (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac06 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 0.992 secs
  Preemption enabled
  Active router is 172.16.11.2, priority 110 (expires in 9.552 sec)
  Standby router is local
  Priority 110 (configured 110)
  Group name is "hsrp-Vl6-6" (default)
Vlan7 - Group 7
  State is Standby
    1 state change, last state change 00:06:31
  Virtual IP address is 172.16.11.65
  Active virtual MAC address is 0000.0c07.ac07 (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac07 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.856 secs
  Preemption enabled
  Active router is 172.16.11.66, priority 110 (expires in 10.288 sec)
  Standby router is local
  Priority 110 (configured 110)
  Group name is "hsrp-Vl7-7" (default)
```

Figure 7.35

```
Vlan8 - Group 0
  State is Standby
    1 state change, last state change 00:06:33
  Virtual IP address is 172.16.11.193
  Active virtual MAC address is 0000.0c07.ac00 (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.128 secs
  Preemption disabled
  Active router is 172.16.11.194, priority 100 (expires in 10.896 sec)
  Standby router is local
  Priority 100 (default 100)
  Group name is "hsrp-Vl8-0" (default)
Vlan9 - Group 0
  State is Standby
    1 state change, last state change 00:06:33
  Virtual IP address is 172.16.11.225
  Active virtual MAC address is 0000.0c07.ac00 (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 0.128 secs
  Preemption disabled
  Active router is 172.16.11.226, priority 100 (expires in 9.856 sec)
  Standby router is local
  Priority 100 (default 100)
  Group name is "hsrp-Vl9-0" (default)
```

Figure 7.36

```
Vlan10 - Group 0
  State is Standby
    1 state change, last state change 00:06:32
  Virtual IP address is 172.16.11.129
  Active virtual MAC address is 0000.0c07.ac00 (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac00 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 0.080 secs
  Preemption disabled
  Active router is 172.16.11.130, priority 100 (expires in 8.736 sec)
  Standby router is local
  Priority 100 (default 100)
  Group name is "hsrp-Vl10-0" (default)
Vlan11 - Group 11
  State is Standby
    1 state change, last state change 00:04:25
  Virtual IP address is 172.16.12.1
  Active virtual MAC address is 0000.0c07.ac0b (MAC Not In Use)
    Local virtual MAC address is 0000.0c07.ac0b (v1 default)
  Hello time 3 sec, hold time 10 sec
```

Figure 7.37

Note: As shown From the Above configurations, the HSRP protocol is seen to be successfully configured.
With the state as “Standby” and the pre-emptive value disabled .

Step 10 : The overall successful HSRP the status of the networking infrastructure of ITIT University

```
CORE2(config)#
*Sep  5 05:22:43.975: %HSRP-5-STATECHANGE: Vlan6 Grp 6 state Speak -> Standby
*Sep  5 05:22:44.006: %HSRP-5-STATECHANGE: Vlan3 Grp 3 state Speak -> Standby
*Sep  5 05:22:44.428: %HSRP-5-STATECHANGE: Vlan5 Grp 5 state Speak -> Standby
*Sep  5 05:22:44.459: %HSRP-5-STATECHANGE: Vlan4 Grp 4 state Speak -> Standby
*Sep  5 05:22:44.736: %HSRP-5-STATECHANGE: Vlan9 Grp 0 state Speak -> Standby
*Sep  5 05:22:45.012: %HSRP-5-STATECHANGE: Vlan8 Grp 0 state Speak -> Standby
*Sep  5 05:22:45.398: %HSRP-5-STATECHANGE: Vlan10 Grp 0 state Speak -> Standby
*Sep  5 05:22:46.189: %HSRP-5-STATECHANGE: Vlan7 Grp 7 state Speak -> Standby
CORE2(config)#
```

Figure 7.38

```
CORE2(config)#
CORE2(config)#
*Sep  5 05:24:51.604: %HSRP-5-STATECHANGE: Vlan13 Grp 13 state Speak -> Standby
*Sep  5 05:24:52.150: %HSRP-5-STATECHANGE: Vlan16 Grp 16 state Speak -> Standby
*Sep  5 05:24:52.245: %HSRP-5-STATECHANGE: Vlan11 Grp 11 state Speak -> Standby
*Sep  5 05:24:52.711: %HSRP-5-STATECHANGE: Vlan14 Grp 14 state Speak -> Standby
*Sep  5 05:24:52.712: %HSRP-5-STATECHANGE: Vlan12 Grp 12 state Speak -> Standby
CORE2(config)#
CORE2(config)#
```

Figure 7.39

Server System Infrastructure Upgrade

The datacenter of the institute hosts the internal servers of the campus. These include servers, which provide authentication services, DNS, DHCP, deployment services, internal mail services, database, and file and print sharing services. These services run in a virtualized environment provided by the type 02 hypervisor VMWare Workstation 15.6, using multiple virtual machines on top of one physical server as shown in the diagram below.

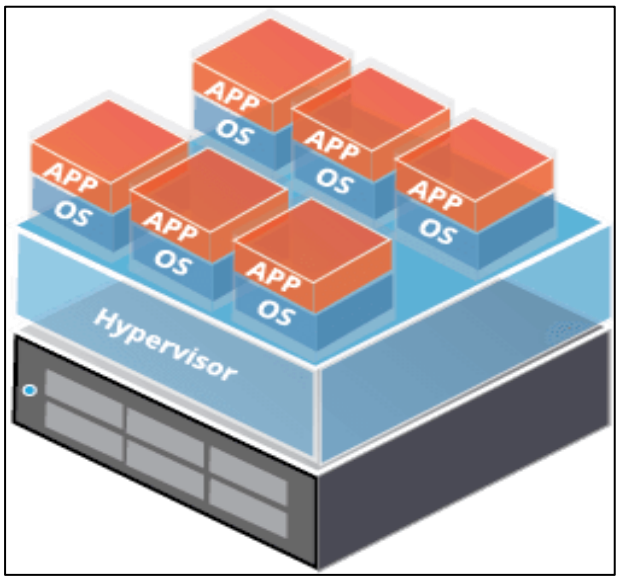


Figure 8.0

The main flaw in this type of environment is the lack of redundancy, fault tolerance, and the improper selection of a hypervisor which has caused various performance issues. This was a problem which needed to be heavily addressed during the upgrade proposal since internal users often had complained about the downtime of critical services such as user authentication, file sharing, and mailing. In order to solve these issues a new server infrastructure complete with heavy virtualization, redundancy, and fault tolerance was implemented. The steps taken in this new implementation was as follows.

- Installation of two new physical servers for failover services and as a backup server.
- Installation of a better performing type 01 hypervisor (VMWare ESXi 6.7.0)
- Installation and setting up of a new Network Attached Storage (NAS) devices complete with redundancy to provide storage services to the internal servers.
- Implementing critical services in a public cloud to provide redundancy in case of total on-site failure.

The institute also required to implement its own web server to host the website (www.itit.com) and a learning management website (LMS) for the students and teachers as part of their "Education Without Boundaries" program (hosted at www.itit-lms.com). To satisfy these requirements a demilitarized zone (DMZ) open to the public Internet had to be created and new hardware and software had to be installed. The implementation plan for the DMZ was as follows,

- Install a network switch to connect the DMZ hosts into the firewalls.
- Connect two new Dell PowerEdge R240 as DMZ servers to serve the roles of web servers and reverse proxies.
- Install and configure web servers to cater the uses of online visitors and academic entities.
- Configure and install reverse proxies to forward requests and responses between the web server and the clients, and to increase security, performance and reliability.
- Provide redundancy and fault tolerance to the web and reverse proxy services.

Technologies used for the Upgrade

- **Type 01 Hypervisor- VMWare Workstation ESXi 6.7.0**

  VMware ESXi (formerly ESX) is an enterprise-class, type-1 hypervisor developed by VMware for deploying and serving virtual computers. As a type-1 hypervisor, ESXi is not a software application that is installed on an operating system (OS); instead, it includes and integrates vital OS components, such as a kernel. This product was used to replace the existing VMWare Workstation virtualization software due to the fact that a type 01 hypervisor runs natively on the hardware, and VMware Workstation runs on top of an operating system. Running natively on the hardware allows native to access the hardware resources, where VMware Workstation is an application which has dependencies to the host operating system.

Figure 8.1

- **Failover and backup servers- Dell PowerEdge R440 Rack Server**

  Two new Dell PowerEdge R440 Rack Servers were installed to match with the existing server of the institute, which is of the same model. The new servers were also configured identically as the preexisting one (see page 35 for detailed hardware configuration).

Figure 8.1

- **DMZ Server Hardware- Dell PowerEdge R240 Rack server (x 2)**

  The Dell PowerEdge R240 server was chosen as the server model to be placed within the demilitarized zone to provide web and reverse proxy services. This hardware was chosen while considering the overall budget and required performance and the R240 provides a well-rounded budget-performance ratio. Dell themselves states that this server is an “entry-level 1U rack server designed for web hosting and multi-purpose applications”. (See page 35 for detailed hardware specifications)

Figure 8.2

- **Network Attached Storage- Synology Rackstation RS1619xs+ (x 2)**

  RS1619xs+ is a high-performance and scalable 1U rackmount NAS designed with upgradable memory and M.2 SSD cache configuration to meet the needs of modern businesses that require a flexible, reliable, and efficient storage solution. This model offers dependable storage and service foundation engineered for performance intensive tasks and optimized for virtualized environments. (see page 40 for detailed hardware configuration).

Figure 8.3

- **DMZ Switch- Cisco Catalyst 2960L-16PS-LL**

  The Cisco Catalyst 2960L switch is an entry-level, fixed-configuration, Gigabit Ethernet switch that provides enterprise-class Layer 2 access. This switch will be placed in an isolated subnet designed especially for the DMZ and access to it will be strictly secured (as per Cisco's guidelines in the "Cisco Guide to Harden CISCO IOS Devices") , with only certain internal clients being able to access its management plane and pointing authentication to the internal RADIUS server. (see page 33 for detailed hardware configuration).

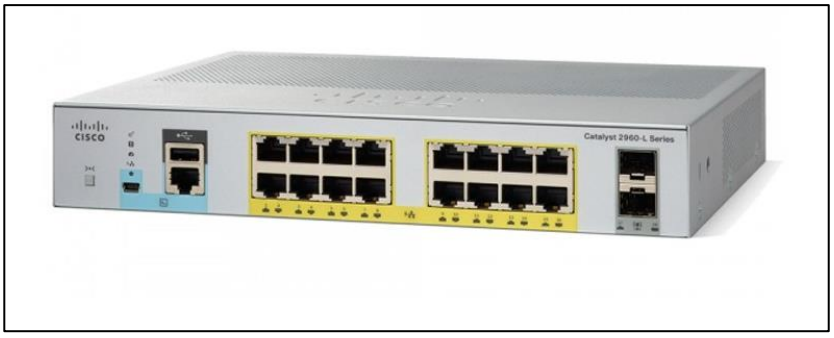


Figure 8.4

- **Migrating critical services to the public cloud- Microsoft Azure**

  Microsoft Azure was chosen to provide the extension of on premise servers to the cloud. The on premise servers were connected to an Azure virtual network using a site-to-site VPN, the traffic flows from the internal on-premises network and the Azure network using an IPsec VPN tunnel implemented at the enterprise edge.

Figure 8.5

Installed services and packages

- **Active Directory Domain Services (Windows Server 2016)**

  Active Directory Domain Services (ADDS) is installed to manage the entire Windows domain of the internal network (itit.com). It is used to store information of the members of the domain, this includes both users and computers, verify their credentials for authentication, and to define their levels of access.
  The active directory domain structure is as follows,

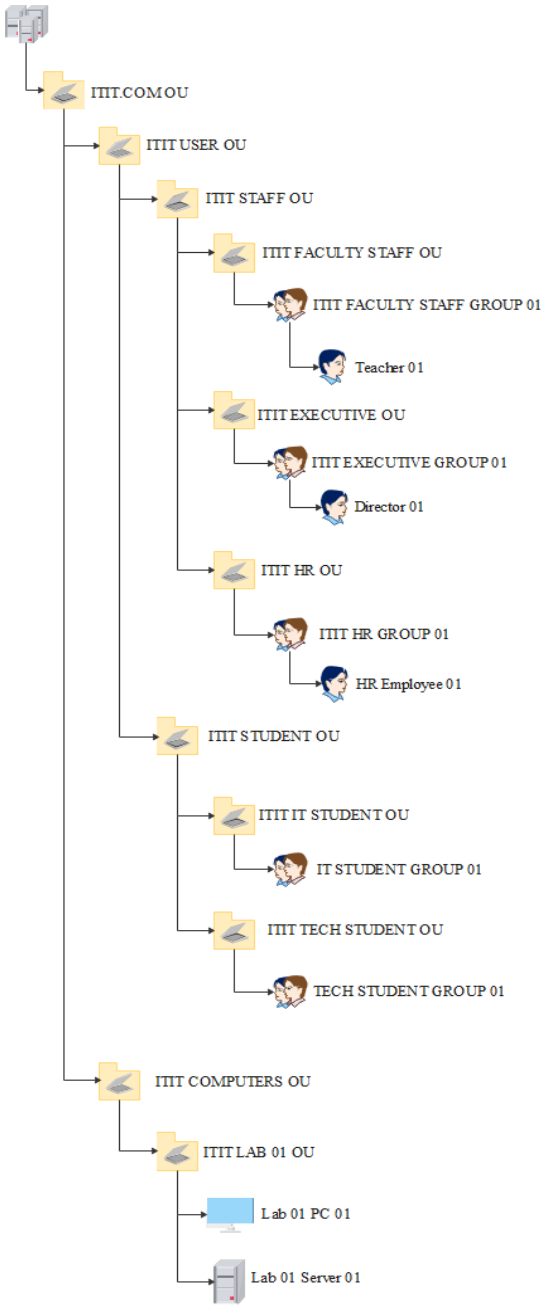


Figure 8.6

- **Domain Name System (Windows Server 2016)**

  Domain Name System will act as the phonebook to the internal domain of itit.com. This service will be hosted within the Active Directory Domain Controller server (since the domain controller will be the one to hold the naming registries for the domain) and will contain all the name records for all the internal servers and other important devices within the network. The DNS naming services will only be available to the internal clients and no outsiders will be able to use it for querying. Hosting an internal DNS server will provide the below benefits

  - DNS queries can be resolved internally. If internal users relied on an external DNS server then the queries will have to be routed out of the internal network into the Internet and back. This can cause performance issues within the network, and many resources will have to be used for a simple task (address translation etc.). Having an internal DNS will keep local DNS querying within the network and prevent them from leaving the trusted internal network.

- Internal DNS can be used to cache information. This reduces the number of request sent externally to authoritative servers. DNS queries are small in size but the sheer volume of them within a network will have impacts.
- Internal DNS servers are very flexible when it comes to managing multiple internal domains.

- **Dynamic Host Control Protocol (Windows Server 2016)**

  Dynamic Host Control Protocol (DHCP) will provide computers with automatic IP addressing and network configuration parameters. The network devices (routers, switches, servers, and firewall etc.) will be configured to use manually configured static IP addresses and so the DHCP service will not provide any configurations to these devices, but only to the end user devices (workstations, laptops, smartphones etc.) within the network.

- **Network Access Control Services (Windows Server 2016)**

  To authenticate users for wireless access we need to implement RADIUS technology. In order to do this we need to implement two server roles

  1. **Active Directory Certificate Services (ADCS)** - To issue authentication and validation certificates for Active Directory users and computers.
  2. **Network Policy and Access Services (NPAS)** - To perform authentication, authorization, and accounting for wireless, authenticating switch, and remote access dial-up and virtual private network connections (VPN). It is the Microsoft implementation of the RADIUS server.

  The RADIUS server provides centralized authentication, authorization and accounting (AAA) services for users who connect and use the wireless network service. Two separate VLANs are also provided for both ITIT domain hosts (domain-joined, trusted devices) and for non-domain joined users (BYOD users) to partition the wireless network access. Guest users have a separate wireless network with no authentication with no means to communicate with the campus internal network (except for what is accessible through the Internet) of the institute and it can only be used to browse the Internet. The ITIT internal users' wireless network allows staff, students and other employees to communicate wirelessly with each other and also to traverse the Internet.

- **Mail Server (CentOS 8)**

  An internal mail server was newly implemented within the local network to send and receive emails between clients, here the mail server acts just like a post office- it receives mail from one client and forwards it to the recipients. The institute wished to have an on premise internal mail server to have more control and flexibility over the emailing system of the campus, to log details about incoming and outgoing messages, view logs for connection and authorization attempts from local mail clients for IMAP, POP3, and SMTP, archiving and handling of mails, and for system administration purposes.
  The mail server used in this implementation was Zimbra Collaboration 9 Network Edition installed on a Red Hat Enterprise Linux Server.

- **File and Printer Sharing (CentOS 8)**

  File servers are a critical part of every organization's IT infrastructure. They provide access to files and databases for desktop/notebook users and for server-based applications. The file server allows the campus to store files on central, shared disks, which users and applications can access as if they were Directly Attached Storage (DAS) on their individual machines. Centralizing files onto centralized servers improves storage security, backup,

and administration. The following benefits were kept in mind when implementing the file server,

- File servers enable files to be shared easily by multiple users and eliminate the need for users to leave their computer on for other users who need access their files.
- File servers enable you to allocate storage quickly and easily. Increase available storage more cost-effectively. Replace or add internal drivers to desktops or application servers without taking them offline.
- File servers improve data security and backups. Recover faster from system problems. Search and locate specific files more quickly and easily. Comply with regulations and improve business continuity.

For this implementation Samba was chosen as the software to provide file sharing services. Samba is a free software released under the terms of GNU General Public License that uses the SMB and CIFS protocols and runs on almost all UNIX or UNIX-like systems. It also supports Microsoft Active Directory and can be set up as a domain controller or a member of the domain.

- **Web Server (CentOS 8)**

A web server is a network service that serves content to a client over the web. This typically means web pages, but any other documents can be served as well. Web servers are also known as HTTP servers, as they use the hypertext transport protocol (HTTP). The institution needed to provide two websites- the www.itit.com which served as the main website of the campus, and an online learning management system at www.itit-lms.com where students and lecturers can exchange education materials and participate in virtual classrooms.

The web server software selected was an Apache HTTP Server instance running on Red Hat Enterprise Linux Server 8. Apache is free, open source, and can be run on multiple platforms and is also one of the most reliable and widely used web server software on the Internet hosting more than 300 million domains. For the campus' requirement, a single Apache web server instance was set up to host both websites using name based virtual hosting which allows multiple web hosts to run on a single instance using the same IP address. With name-based virtual hosting, the server relies on the client to report the hostname as part of the HTTP headers and looks up the hostname in the DNS records (locally or of a preconfigured server) to maps the hostname to the proper IP address. This is relatively easy to configure and reduces the demand for the scarce IP addresses.

- **Reverse Proxy (CentOS 8)**

A reverse proxy server was configured to sit in front of the web server to forward traffic between the client and the web server, to an external client these requests seem to have originated from the reverse proxy itself and thus hides the existence of the backend web server.

The reverse proxy software used in the implementation was Nginx installed once again on a server running Red Hat Enterprise Linux and was configured to provide caching, traffic logging, and SSL offloading.

Figure 8.7

Server

Figure 8.8

Virtualization Architecture

The campus datacenter servers runs in a heavily virtualized environment. Two physical server hardware units running VMWare ESXi 6.7.0 provide a platform for the virtual machines with server software installed on them to run on and provide services to the clients. While these virtual machines do run "on" the VMWare ESXi hypervisor in the physical server, these virtual machines are stored on the network attached storage (NAS) devices. The NAS devices act as **datastores**- which can be defined as an independent storage space for centralizing the virtual machines' physical storage. This is important in providing fault tolerance and redundancy to the system, such that, in the event a physical host unit becomes unavailable the virtual machine can continue to run in the other unit through a process called migration (configurations will be explained in detail in later sections).

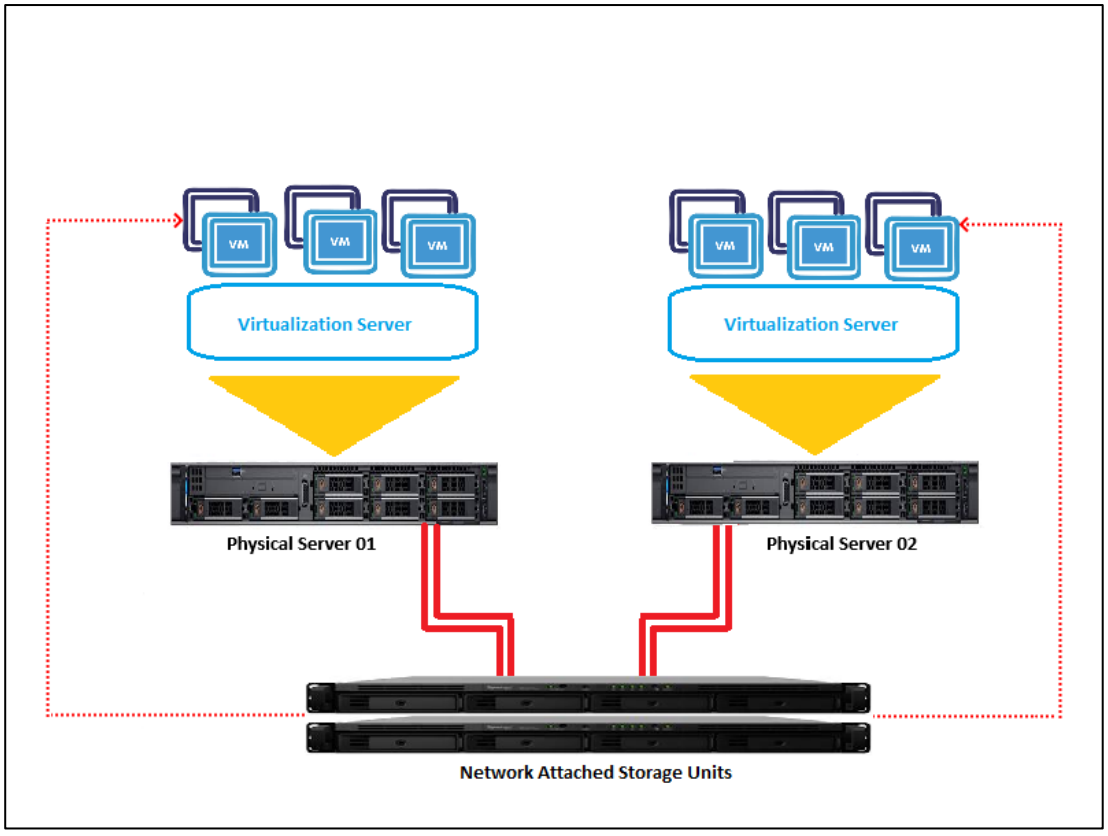


Figure 8.9

Internal Server Topology (Logical)

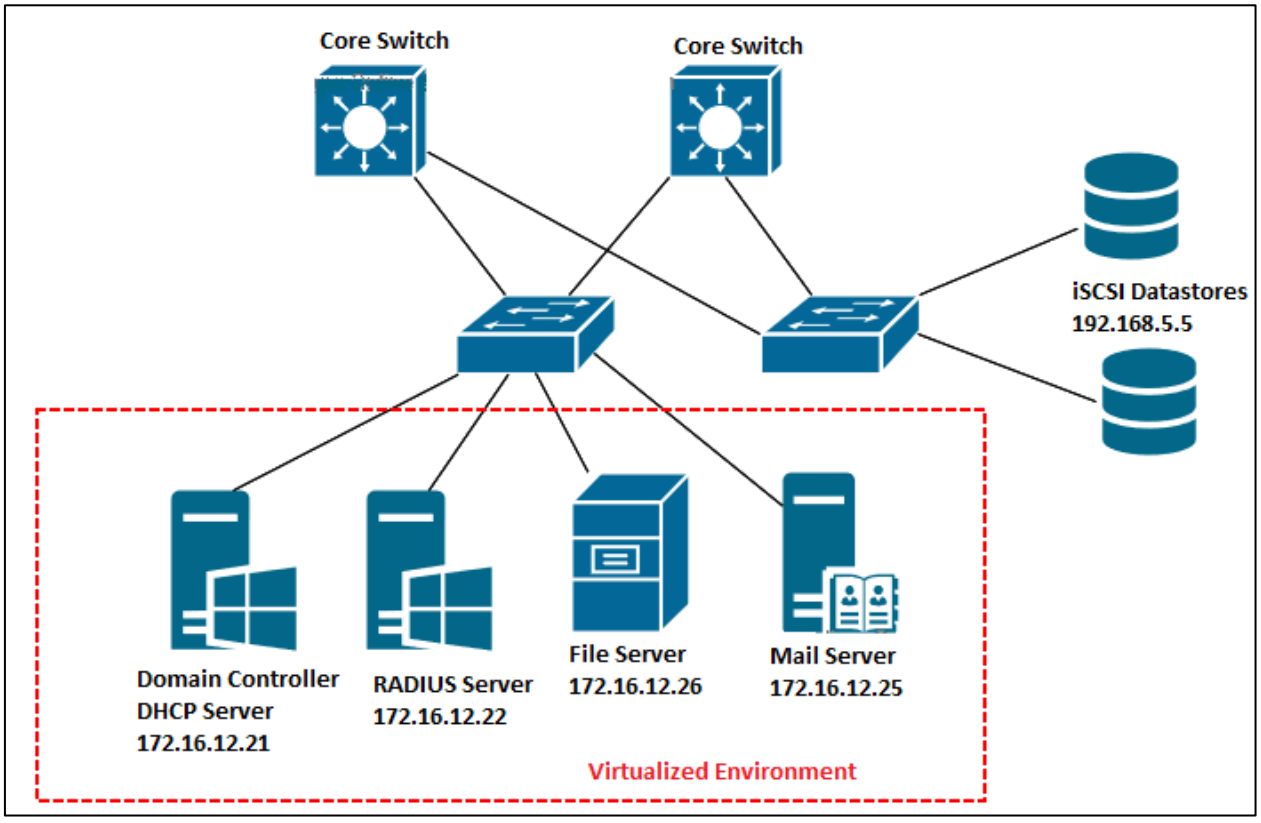


Figure 8.10

Demilitarized Zone Server Topology (Logical)

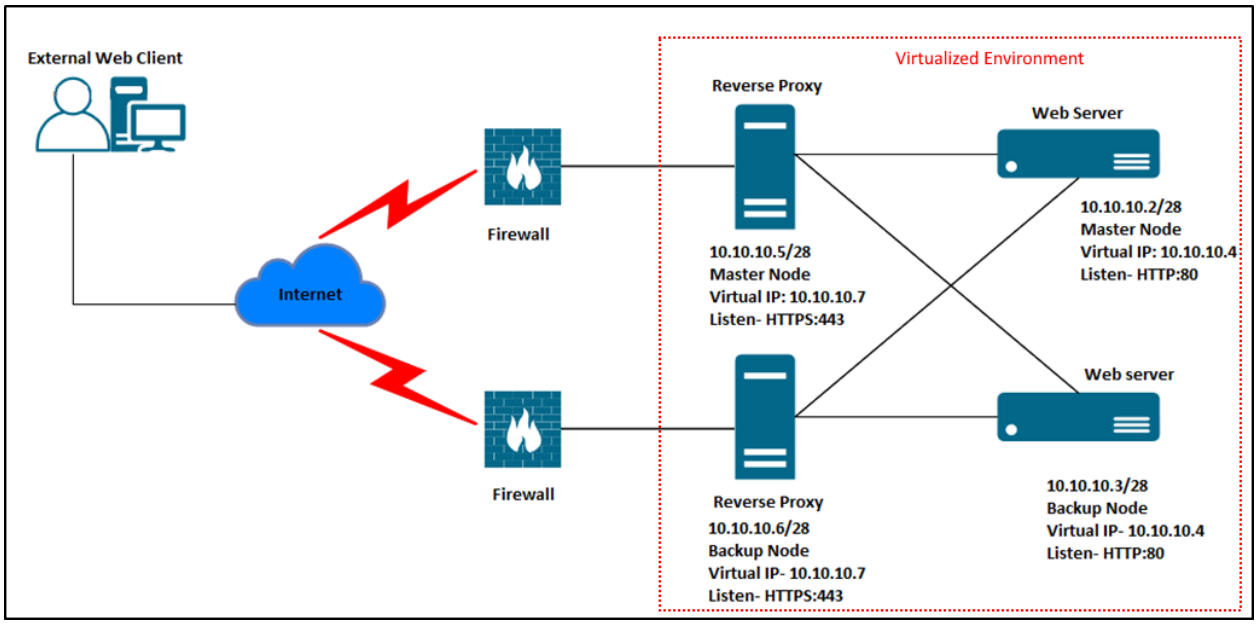


Figure 8.11

Installation and Configuration of Servers
Installing VMWare ESXi 6.7.0

Prerequisites:

- Compatible hardware
- ISO image of VMWare ESXi 6.7.0 (VMKernel Release Build 14320388)
- Bootable device (DVD or USB etc.)

1. Mount the ISO image and power on the host hardware, and allow to boot from the ISO image.
2. The boot process will initiate and will progress to the Loading ESXi installer screen.

Figure 8.12

3. The screen will then start to load the installation files from the installer

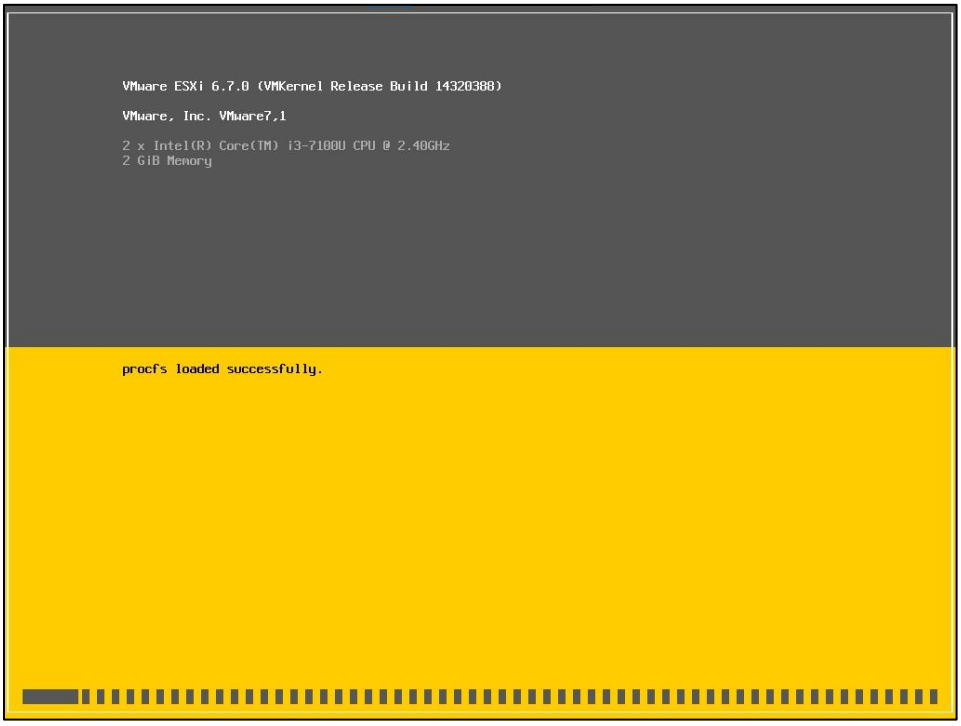


Figure 8.13

4. Once the ESXi installer has loaded up all the installation files, the "Welcome to the VMWare ESXi 6.7.0 Installation" will appear on the screen. Press the Enter to continue with the installation.

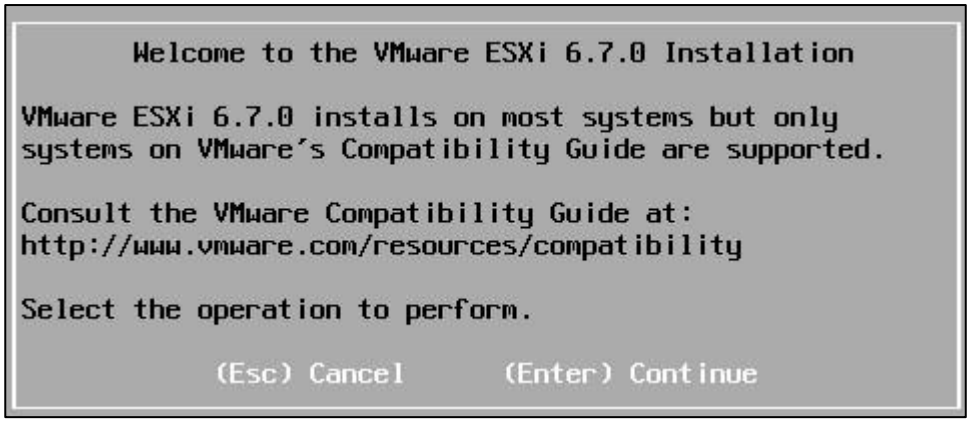


Figure 8.14

5. Select the boot storage device (this is the storage device where VMWare ESXi will be installed. Note that, any existing data will be erased). Select the storage for installation and press Enter to continue.

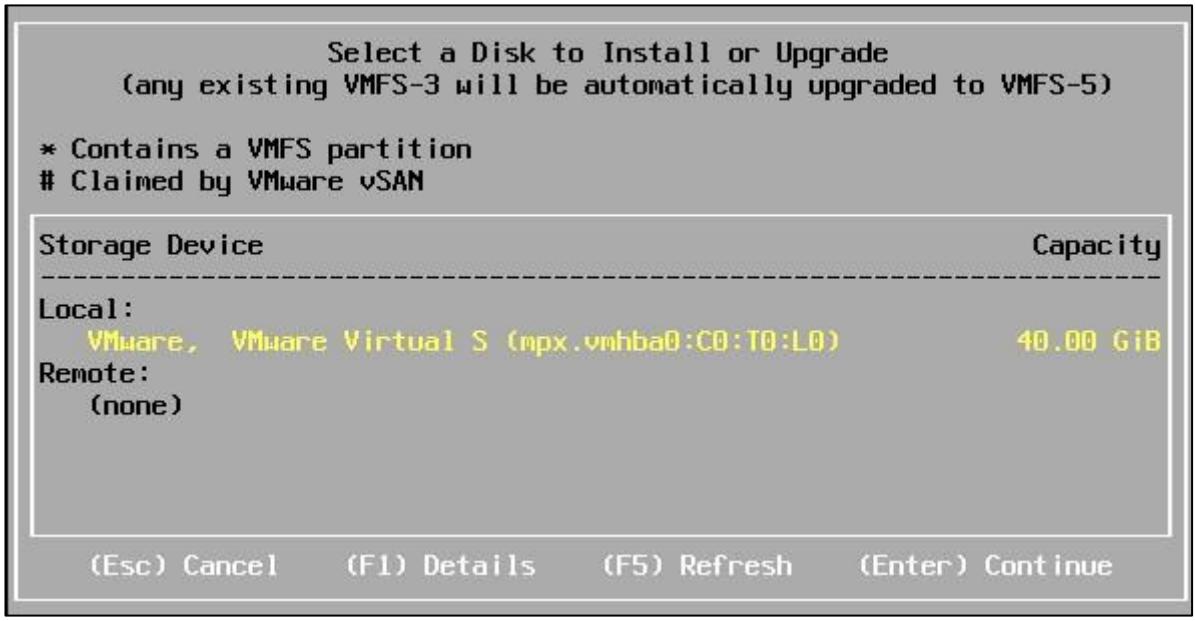


Figure 8.15

6. Select the Keyboard Input layout and press "Enter" to continue.

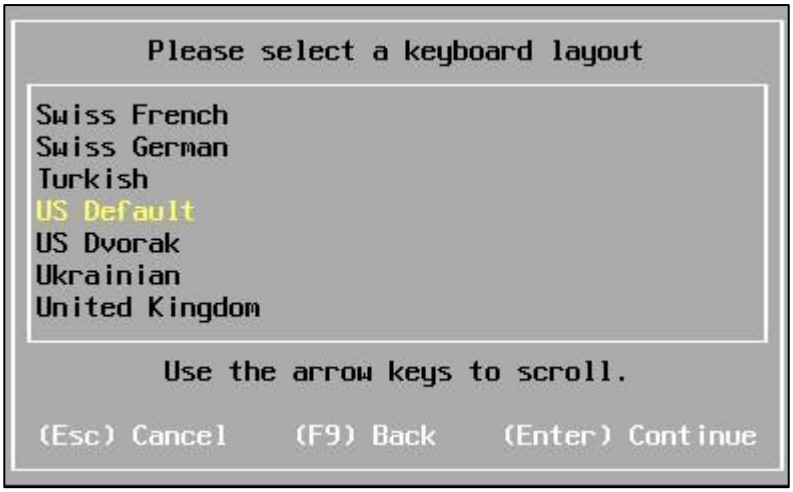


Figure 8.16

7. Enter a password for the root account and confirm the password by retyping. Be sure to select a strong root password as this will allow access to your ESXi host.

Figure 8.17

The installation process will now start. Do not turn off the host until the installation has completed. Once the installation reaches 100%, remove the bootable device containing the ISO and restart the host to continue with the post-installation configurations.

Figure 8.18

Press Enter to trigger a reboot (confirm that the installation media has been removed from the host).

Figure 8.19

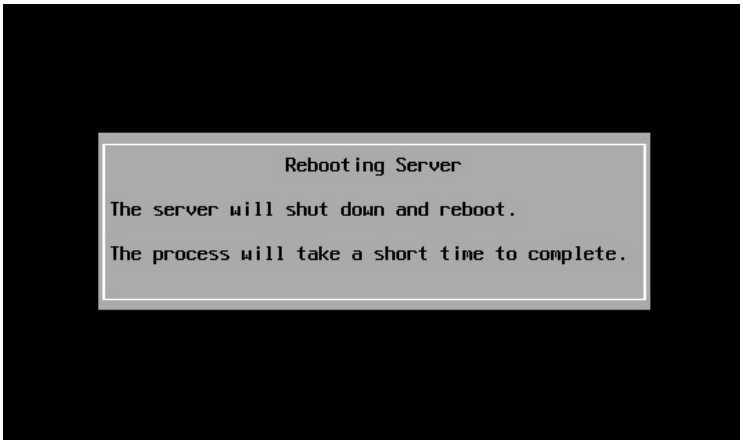


Figure 8.20

8. Once the host has been rebooted the VMWare ESXi 6.7.0 Direct Console User Interface (DCUI) screen will be visible on the display. It is shown that the host has not received a valid IP address and that the DHCP lookup has failed, this is acceptable as the ESXi hosts needs to be configured with static IP addresses. Press F2 and enter the root password to configure the network interfaces and addresses.

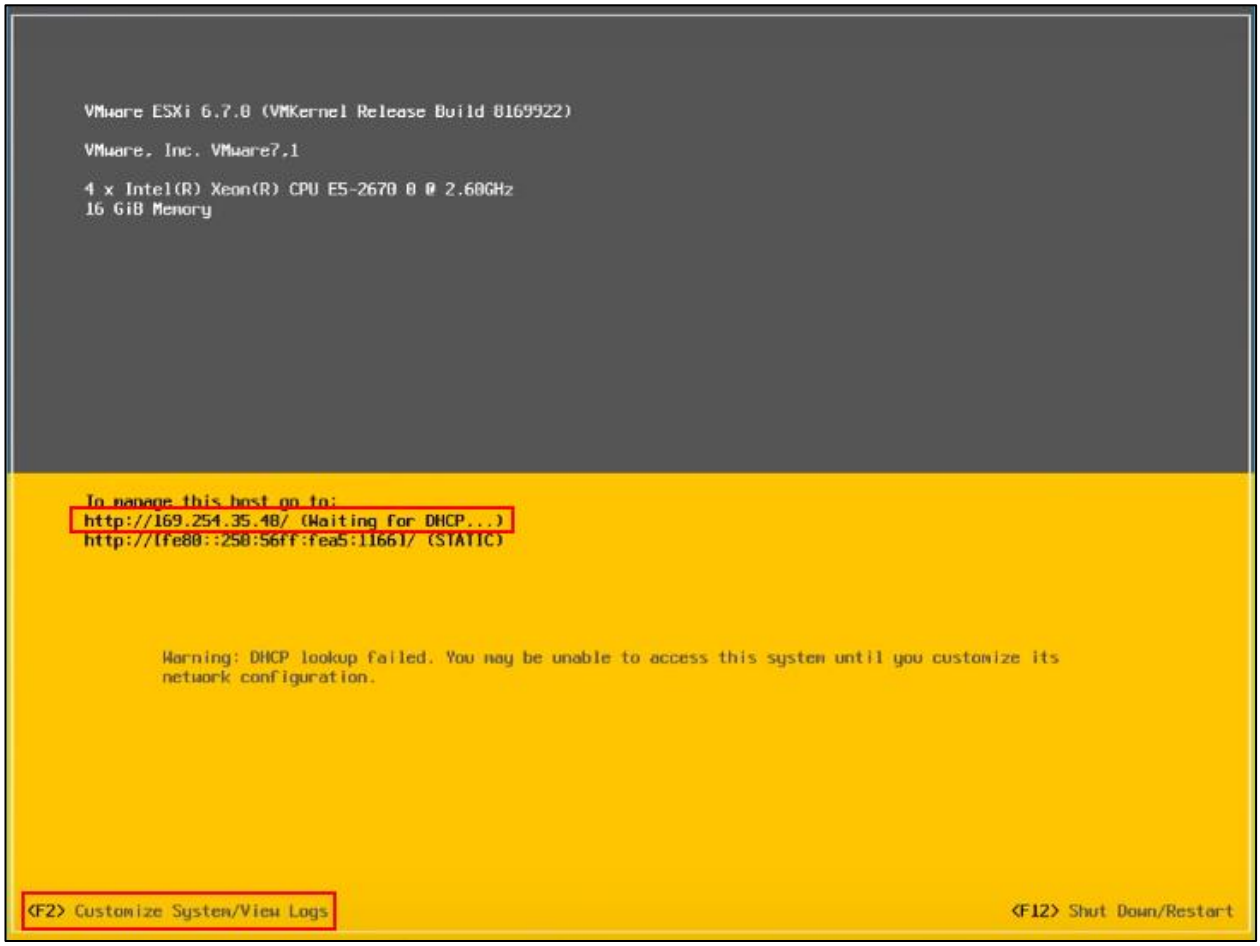


Figure 8.21

9. In the configuration menu select “Configure Management Network” and then select “Network Adapters” to choose the network interface card that needs to be configured with the management network addressing.

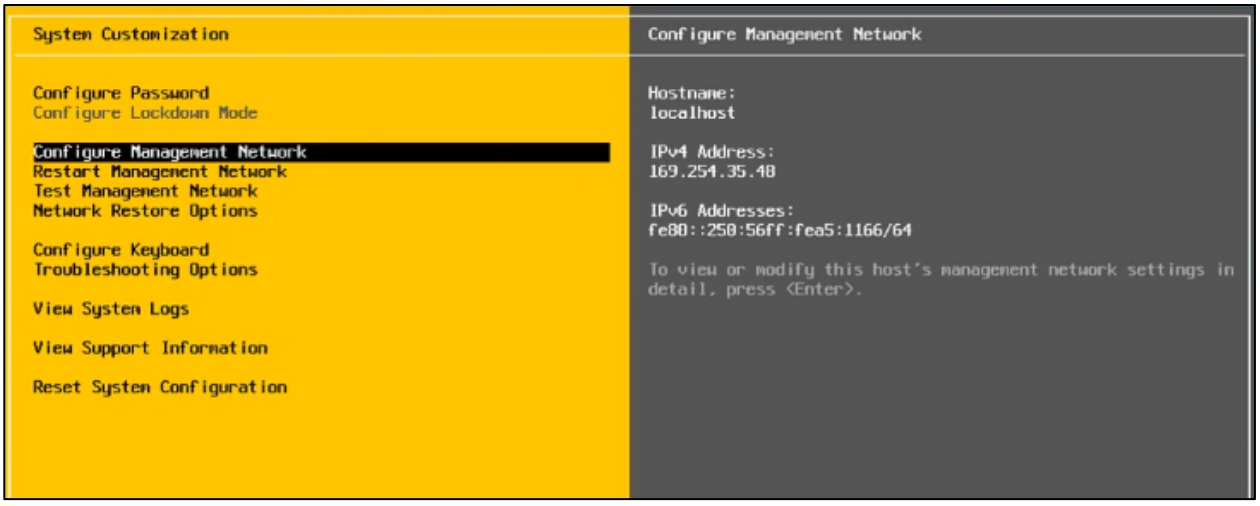


Figure 8.22

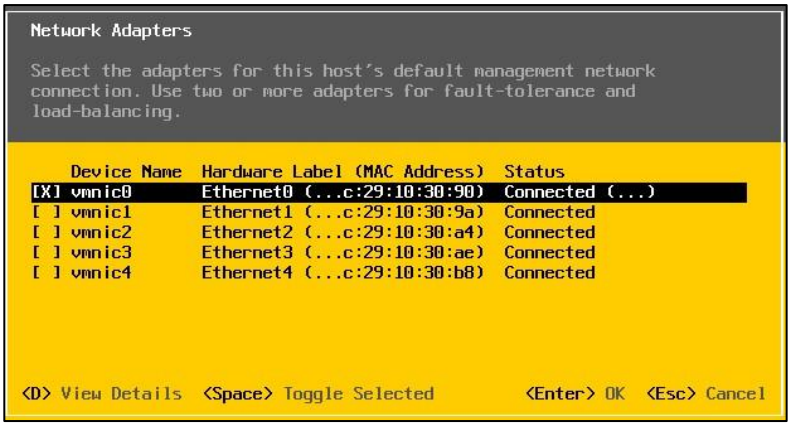


Figure 8.23

10. Next select "IPv4 Configuration". Choose the "Set static IPv4 address and network configuration" by pressing the spacebar and enter the relevant addressing information. Press Enter when done.

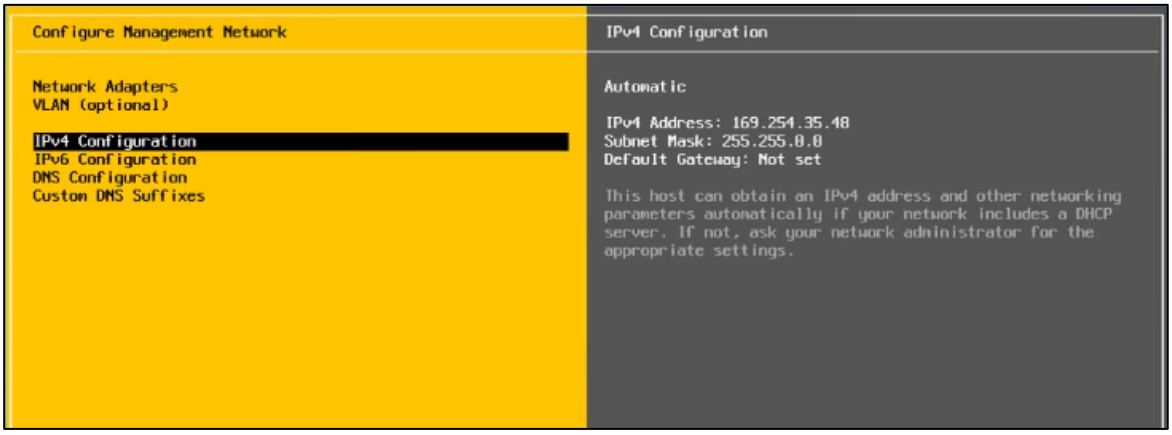


Figure 8.24

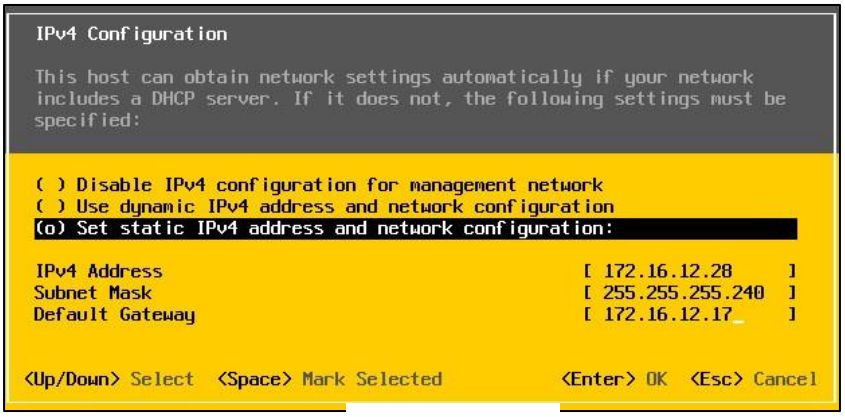


Figure 8.25

11. Next select "DNS Configuration" and set a hostname to the ESXi host and press Enter to confirm.

Figure 8.26

12. Once all the IP addressing information has been entered press Esc to exit the configuration menu.

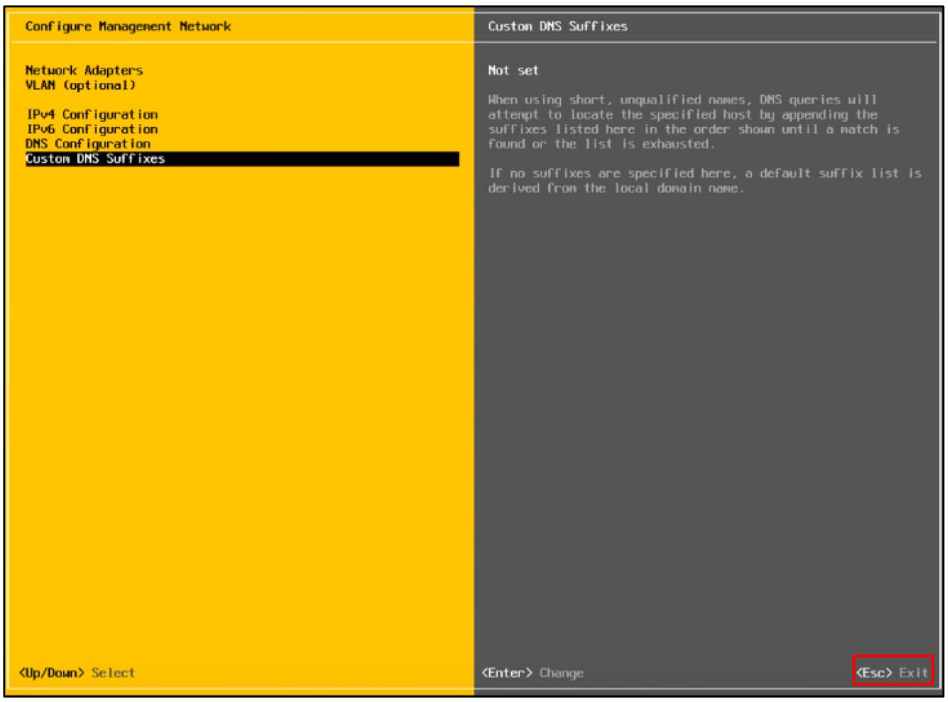


Figure 8.27

When prompted to reboot the host press Y. This is required to apply the changes.

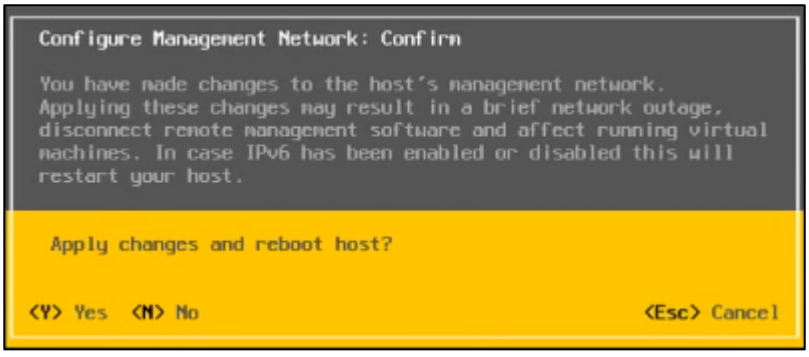


Figure 8.28

Figure 8.29

Once the ESXi host has been rebooted, access the DCUI display and confirm that the changes has been applied. The set IP address should be visible.

Figure 8.29

Configure the other ESXi host using the same procedure. The IP addressing information for the hosts are as follows.

1. **ESXi 01**

   IP address: 172.16.12.28
   Subnet mask: 255.255.240
   Default Gateway: 172.16.12.17
   Hostname: esxi-01.itit.com

2. **ESXi 02**

   IP address: 172.16.12.29
   Subnet mask: 255.255.255.240
   Default Gateway: 172.16.12.17
   Hostname: esxi-02.itit.com

The second ESXi host has been configured with the above IP address.

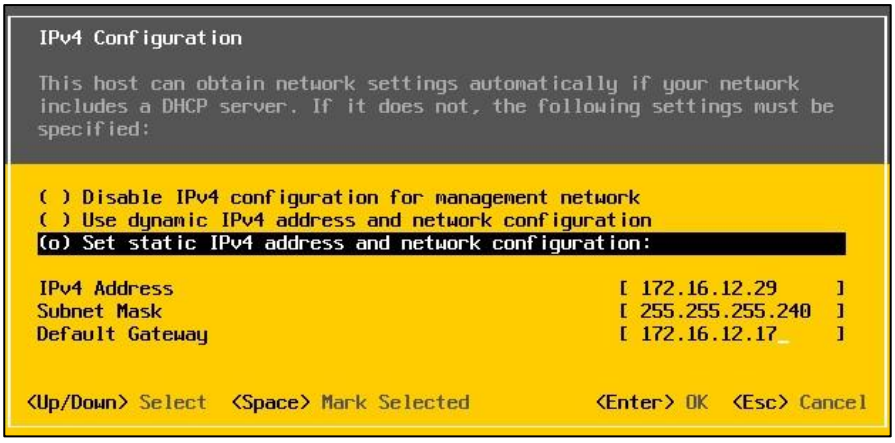


Figure 8.30

The ESXi web interface can now be accessed by keying in the IP address of the ESXi host or using the URL. This provides a GUI that can allow the user setup configurations and monitor activities easily.

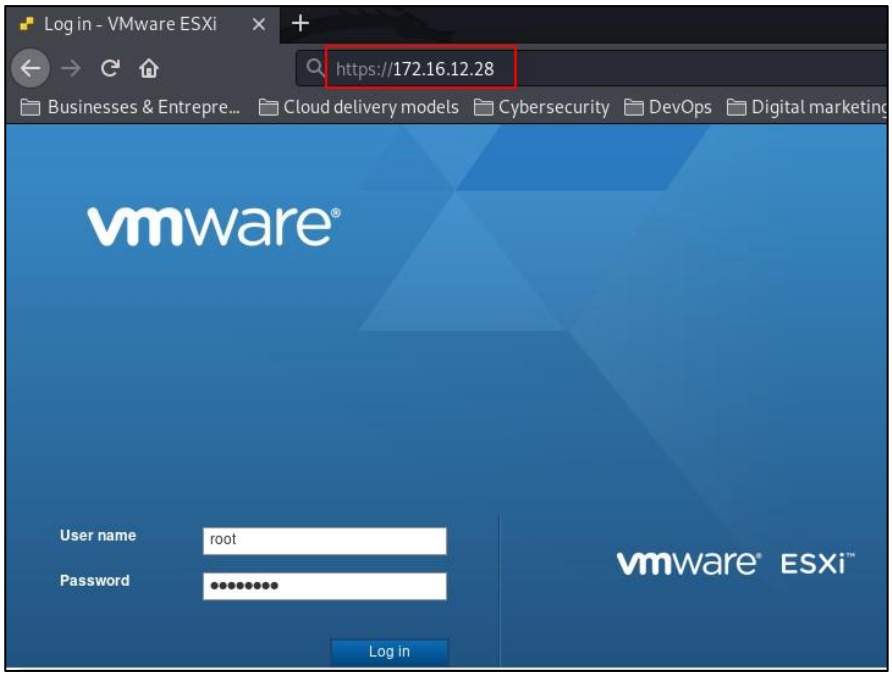


Figure 8.31

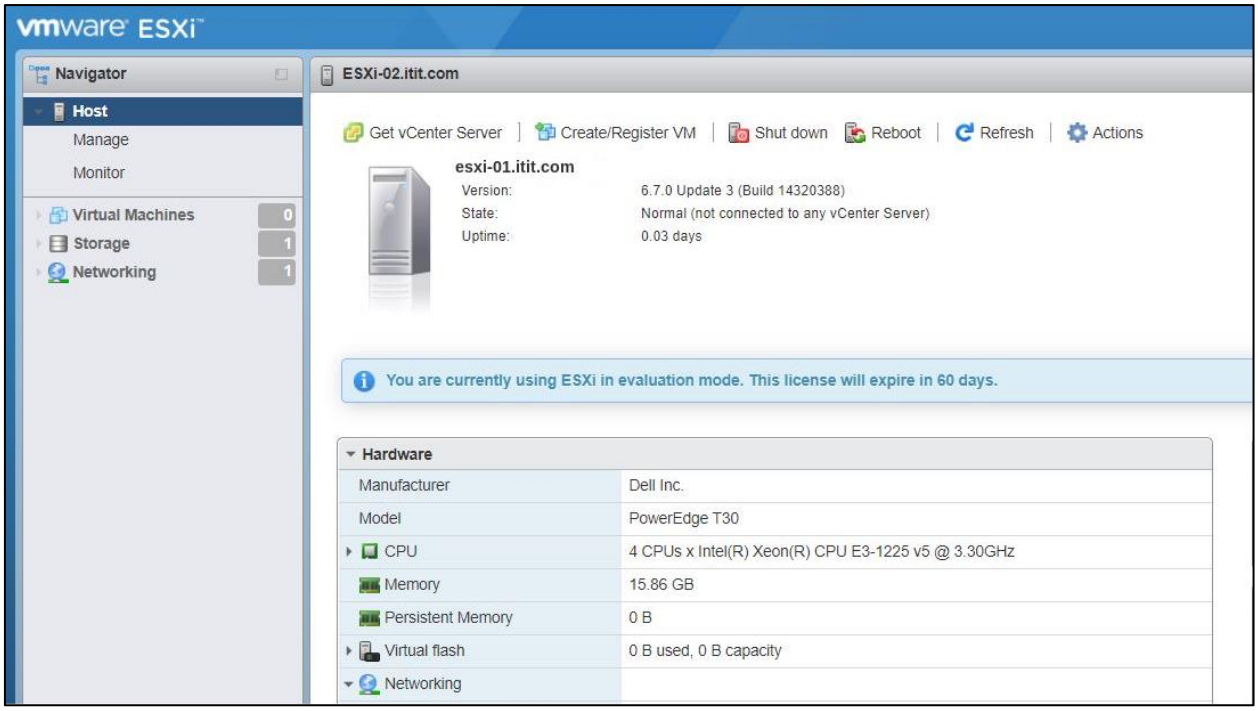


Figure 8.32

Below is the web interface of the second ESXi host as proof of setup.

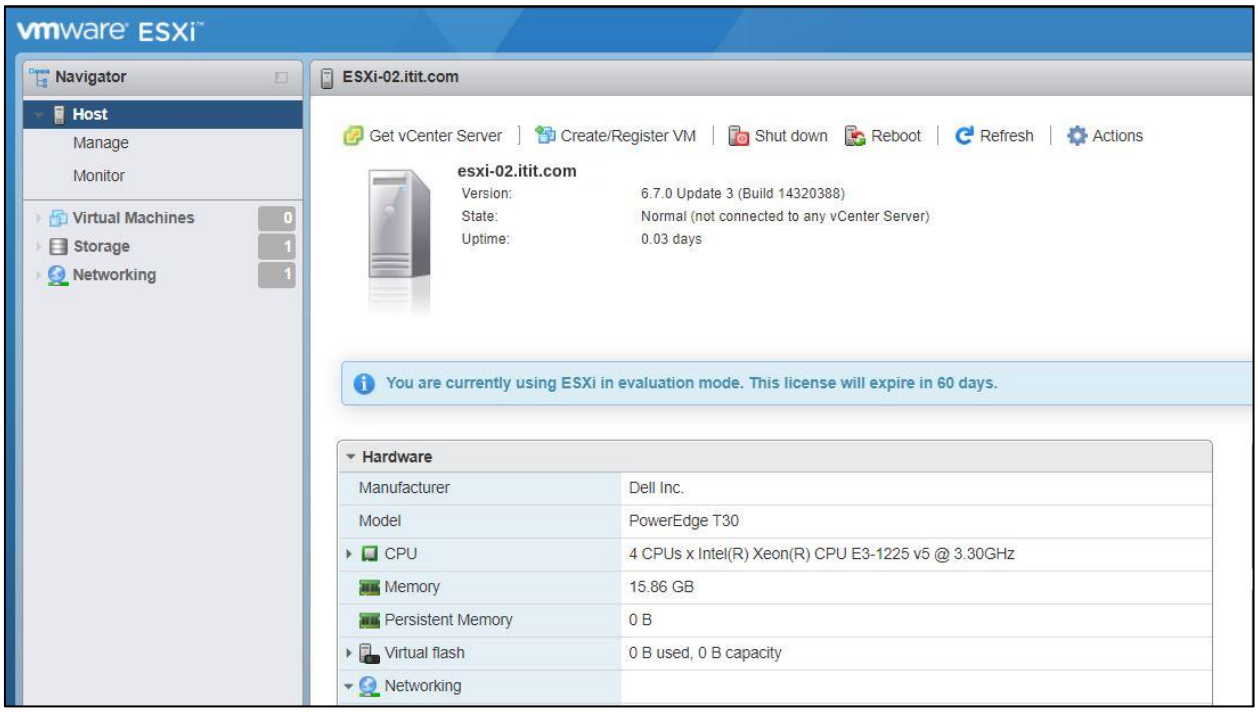


Figure 8.33

Configuring Network Attached Storage Devices

The next step would be to configure the network storage devices which would act as the datastores to store the physical data of the virtual servers that are to be installed. The steps that were needed to be done to set up the network storage devices is listed below,

1. Setup high availability between the network access storage devices.
2. Create a storage pool in the storage cluster.
3. Create volume(s) of storage within the storage pool.
4. Create LUNs (Logical Unit Numbers)
5. Map the LUNs as iSCSI targets.
6.

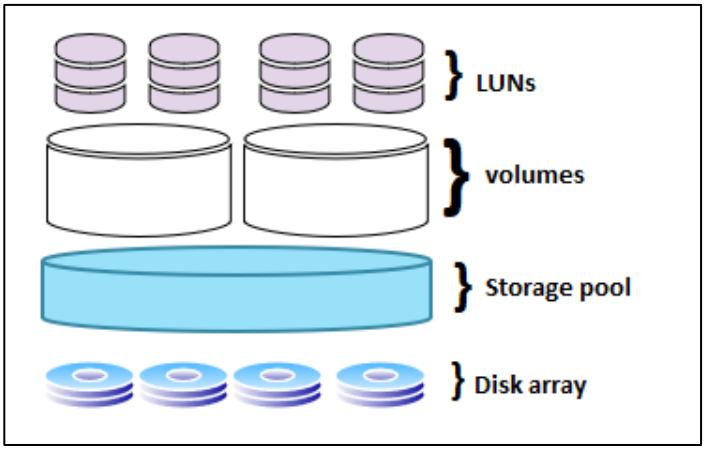


Figure 8.34

The storage area topology is shown below. The two devices are connected to the storage area switch using an aggregated link to provide maximum bandwidth for the transfer of data and also to provide a minor level of link redundancy, since if one line were to occur the link would continue to work before switching. The NAS devices are also interconnected using a heartbeat link, once the high-availability cluster has been created, the Heartbeat connection facilitates data synchronization and replication between the active and passive servers. If contact between the two devices is lost for a given amount of time, the system will determine if a switchover is necessary.

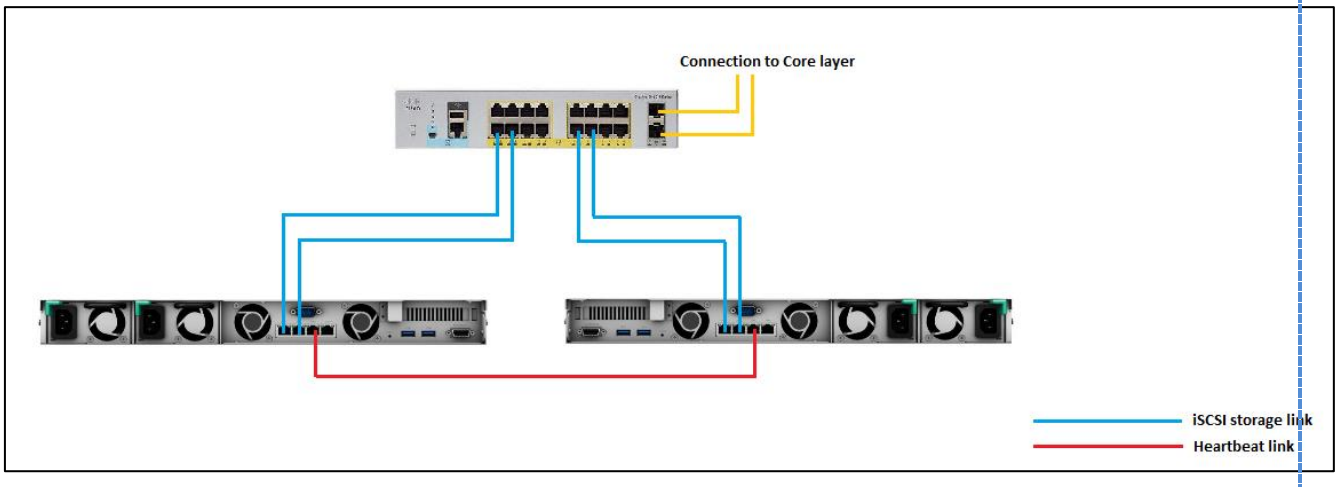


Figure 8.35

The above mentioned setup providing high availability for the datastore is configured using the Synology High Availability Manager package as follows.

01. Access the DSM interface of the NAS that is to be configured as the active unit and select the Synology High Availability package.

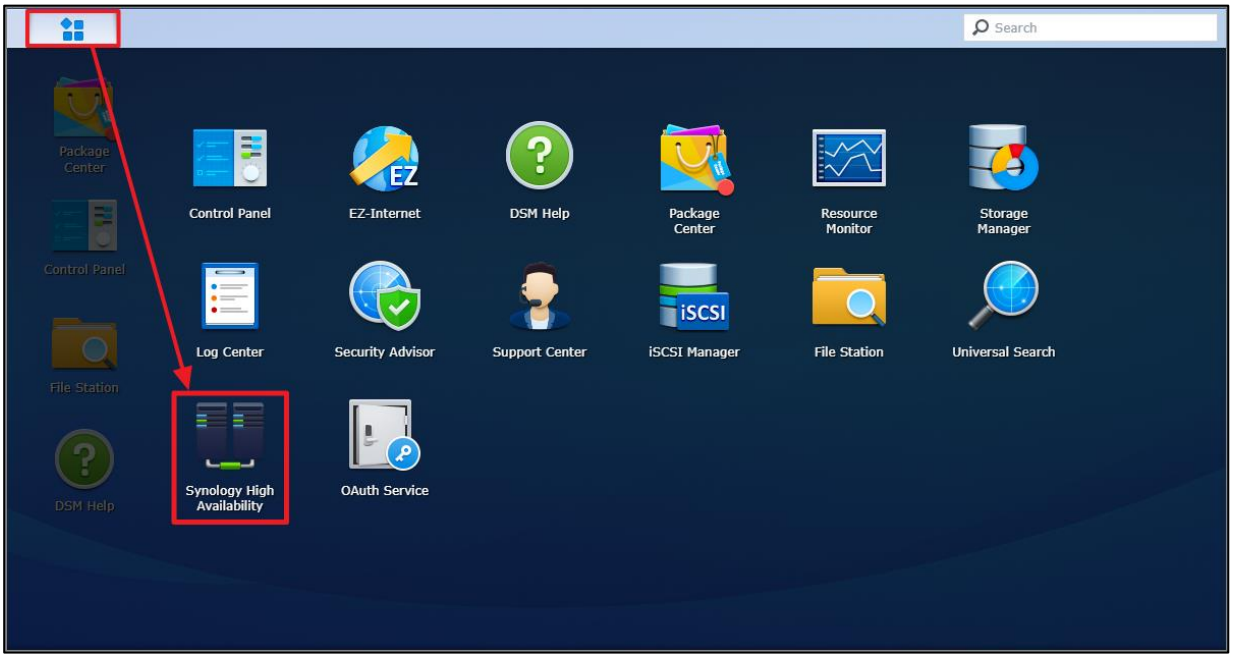


Figure 8.36

02. Assign the proper interfaces as the cluster connection and the heartbeat connection.

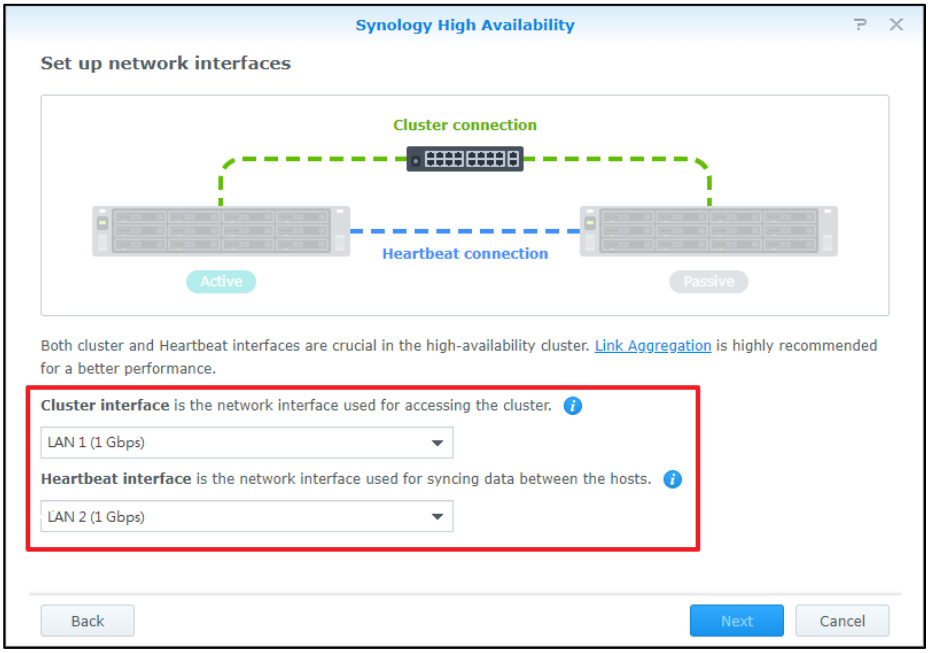


Figure 8.37

03. Enter the passive NAS unit information and user credentials.

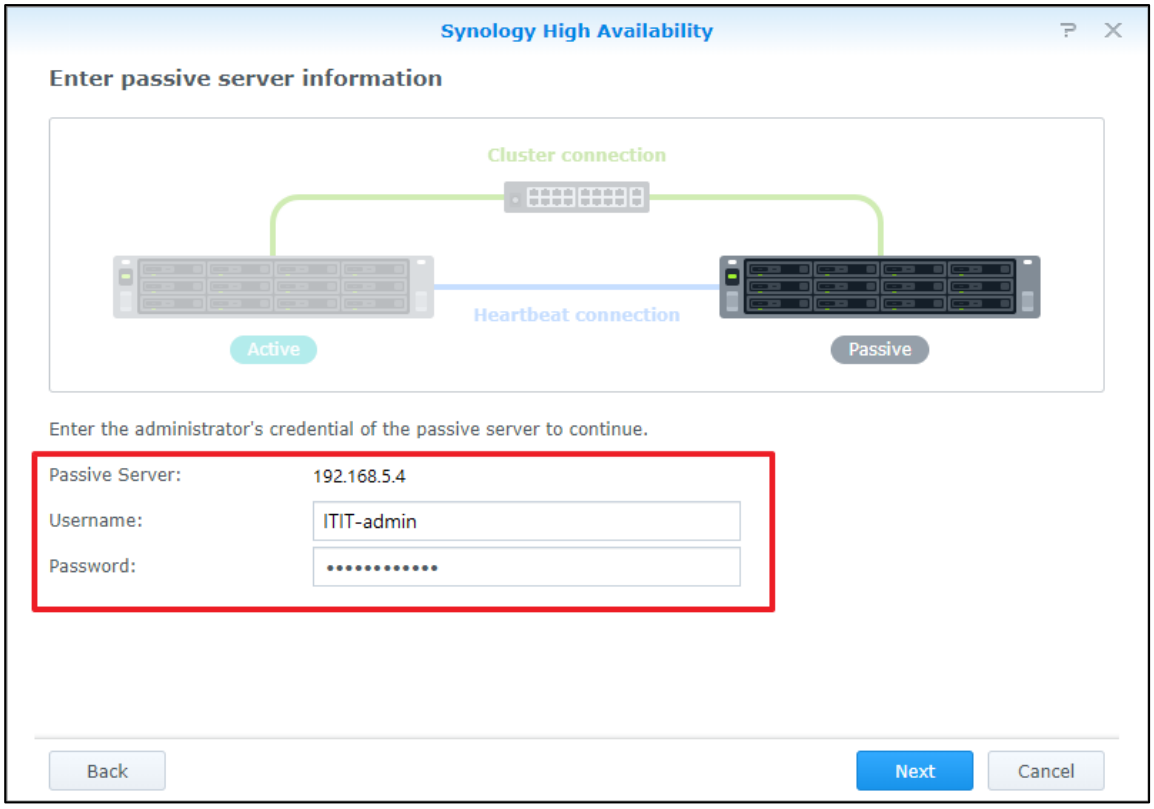


Figure 8.38

04. Enter the information for the high availability cluster. The IP address entered here will be a virtual IP address to represent the cluster.

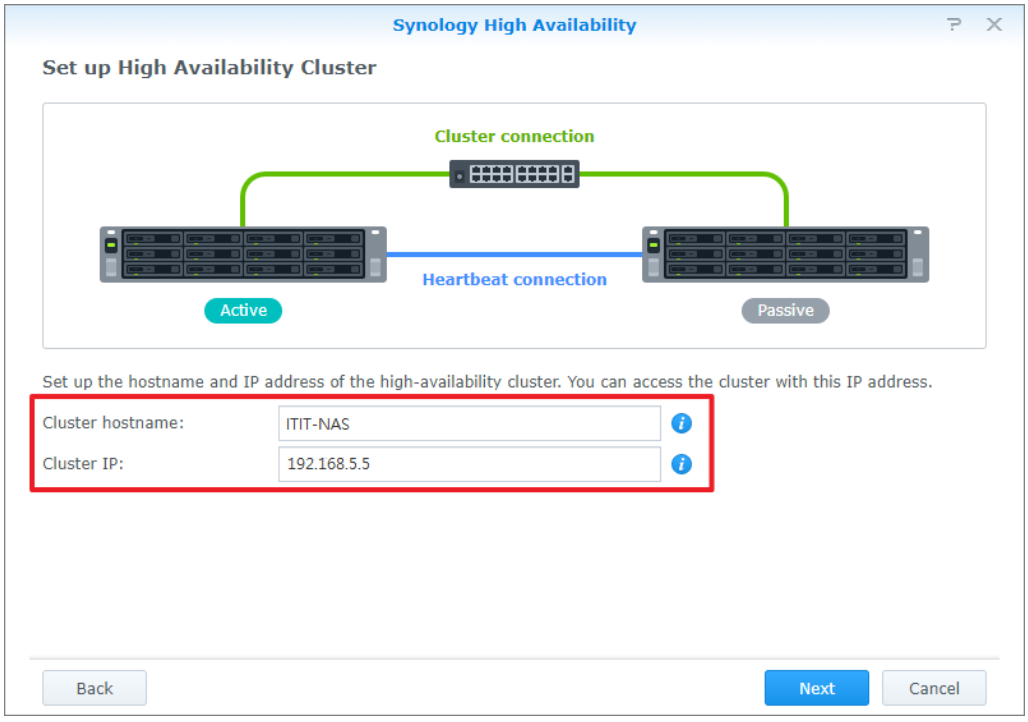


Figure 8.39

05. The system will carry out a few checks to verify if the high availability cluster can be created successfully. Click Next if all the requirements pass.

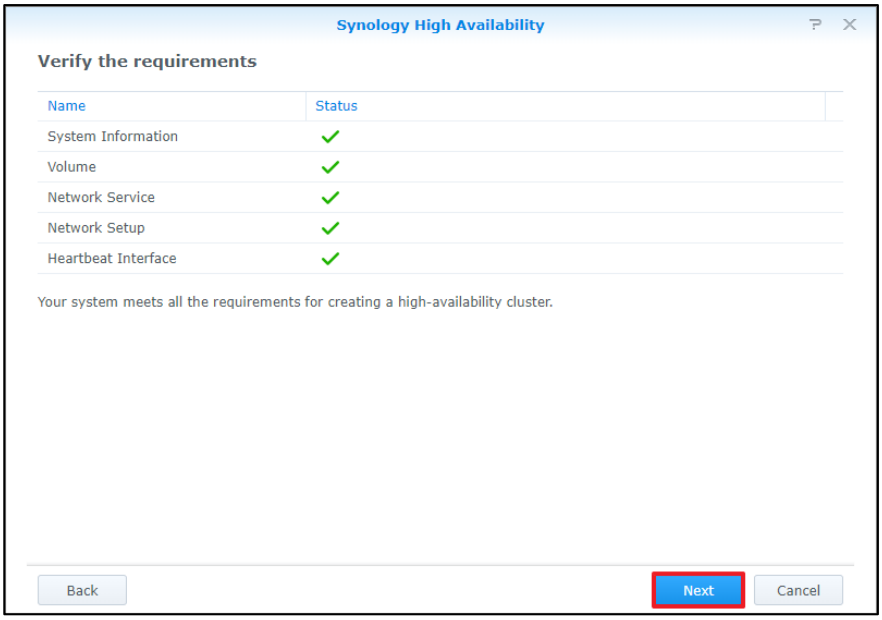


Figure 8.40

06. Since no volumes or LUNs were created in any of the storage devices select the first option and continue.

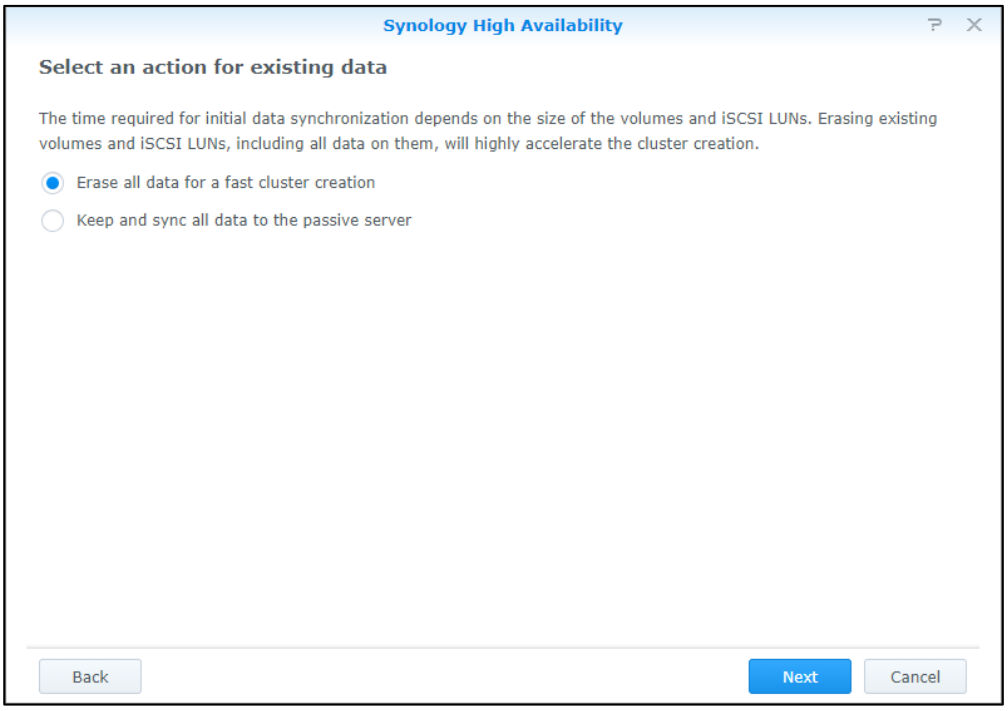


Figure 8.41

07. Confirm the settings and click Apply to create the cluster.

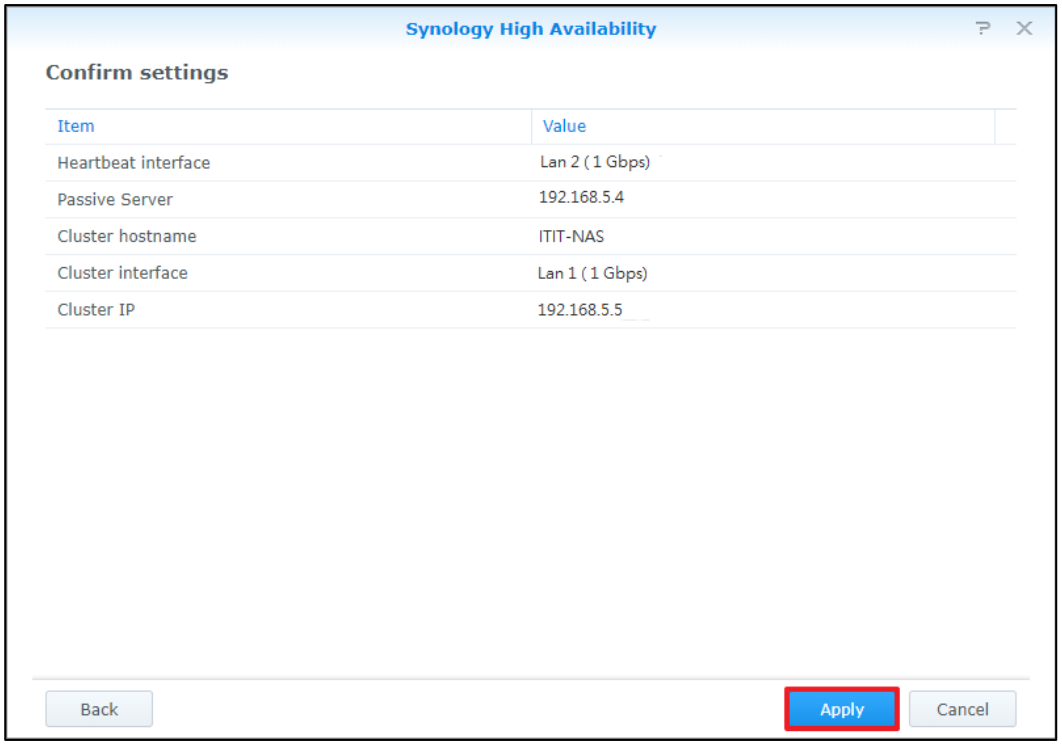


Figure 8.42

08. The system will warn that all the drives will be formatted and all data erased for the creation of the high availability cluster. Tick the box to confirm the selection and click Yes,

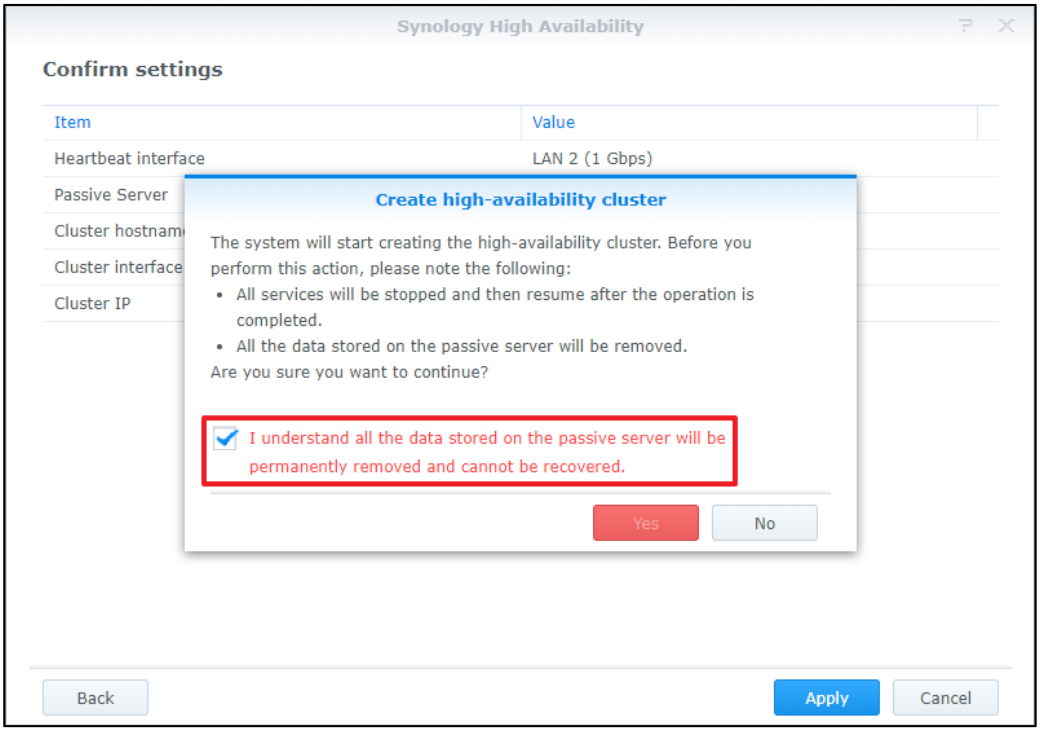


Figure 8.43

09. The cluster will start creating. Ensure that the power is not switched off in o=both storage units until the high availability cluster is created. This process can take a few minutes.

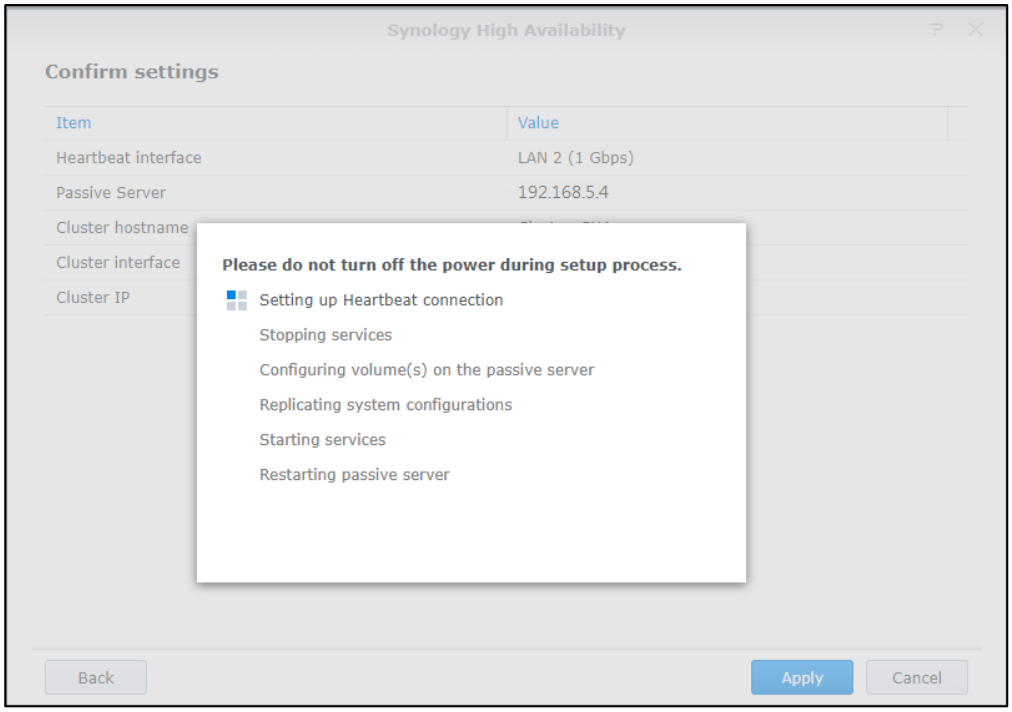


Figure 8.44

10. Once the cluster has been created check the Overview section of the Synology High Availability Manager and ensure that the cluster is up and healthy.

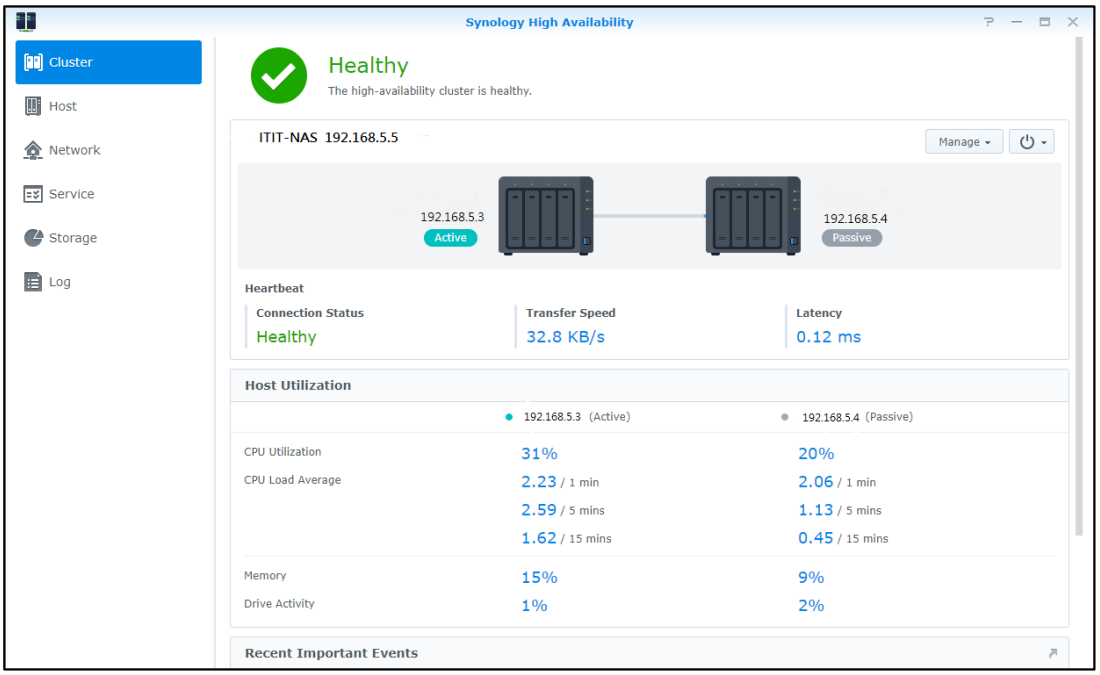


Figure 8.45

Once the NAS cluster has been configured with a virtual IP address the storage can be configured to provide highly available datastores to store the virtual servers. The below configurations explain how to setup the storage level configurations, these changes will be made on the active unit and replicated on the passive unit, which will then be ready to take over the role should the active unit were to ever fail.

01. Enter the virtual IP address of the storage cluster (192.168.5.5:5000) or the hostname (http://itit-nas:5000) and access into the storage cluster.

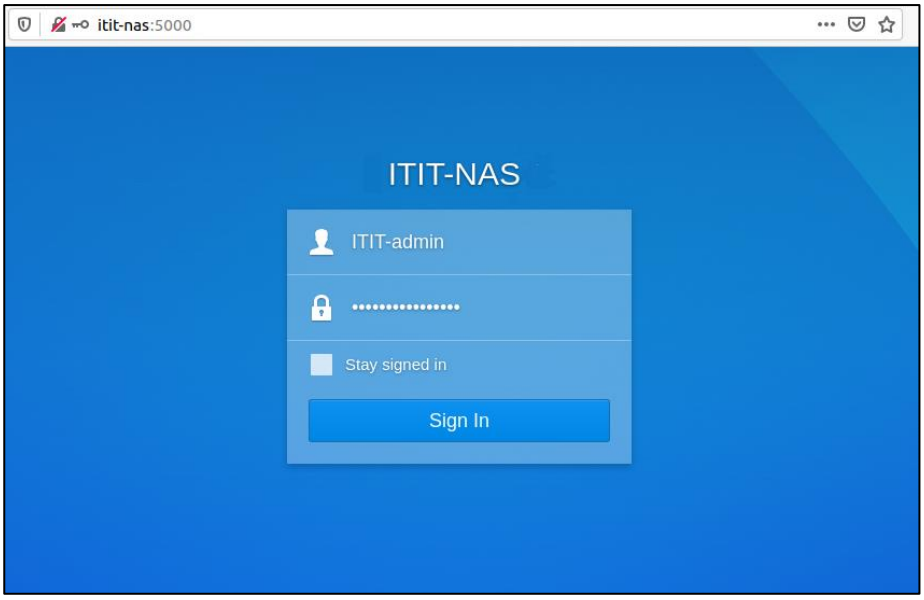


Figure 8.46

02. Once signed in go to the DSM menu and select Storage Manager.

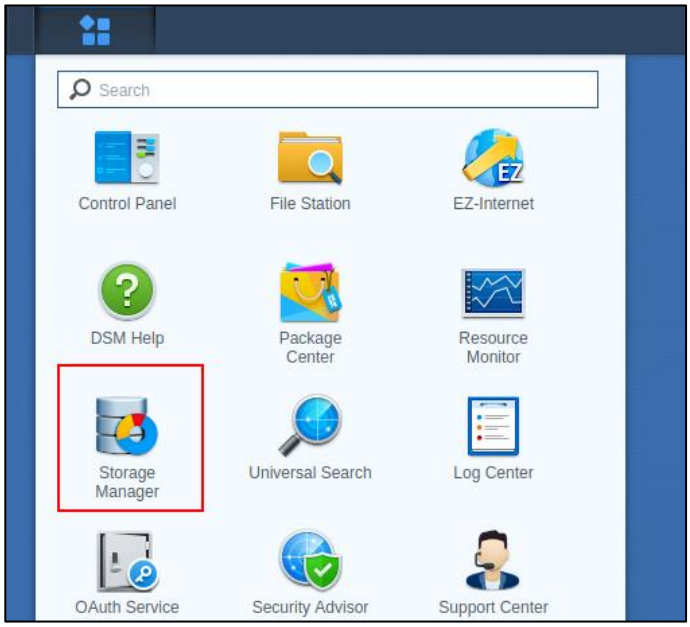


Figure 8.47

Select “Storage Pool” from the window and click on the “Create” button to make a new storage pool. The Volume creation wizard will then start, follow the steps as mentioned.

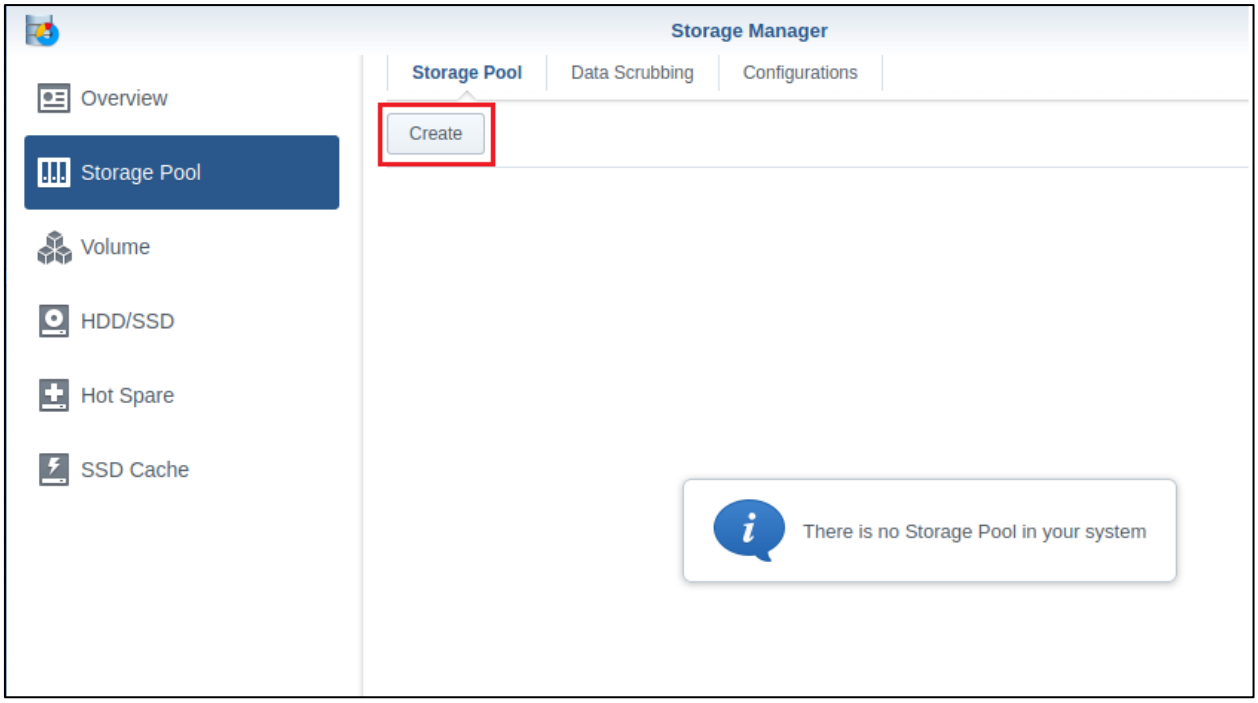


Figure 8.48

03. Select the "Better Performance" option. This will create a single volume for storage with RAID feature enabled instead of Synology Hybrid RAID (SHR).

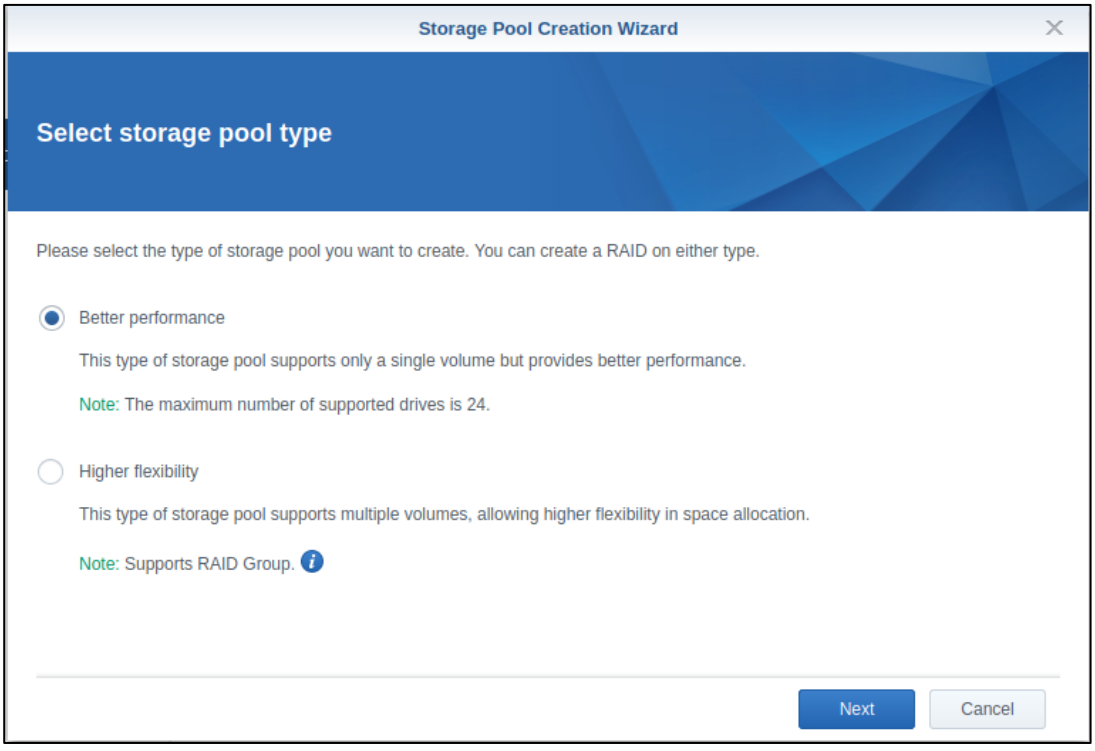


Figure 8.49

04. The storage pool was named as 'ITIT-NAS Storage Pool' and selected RAID type was RAID 1 (mirroring). Click Next to continue with the Storage Pool Creation Wizard.

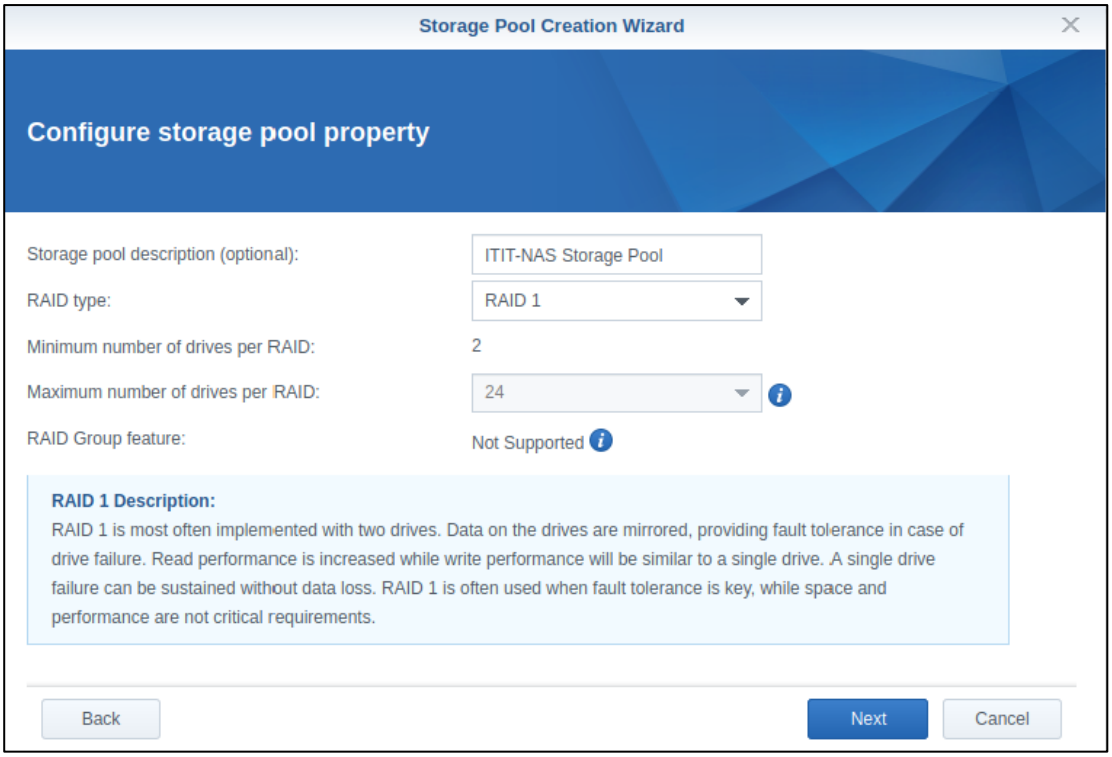


Figure 8.50

05. Choose the physical drives that are to be included within the storage pool. These disks will be formatted for the creation of the storage people, ensure that necessary backups have been taken.

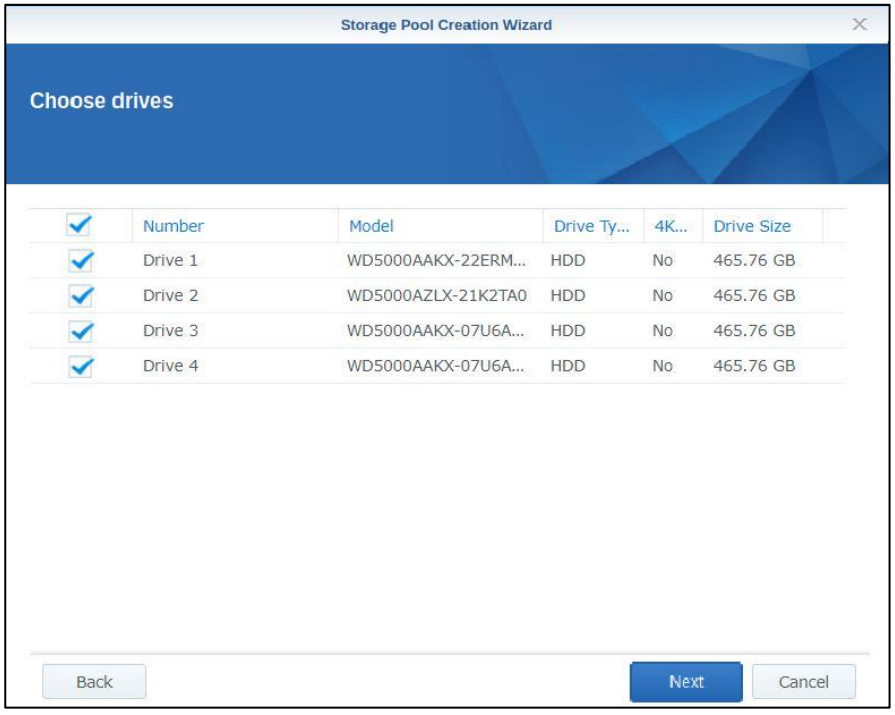


Figure 8.51

Click OK to continue formatting the drives to procees with the storage pool creation.

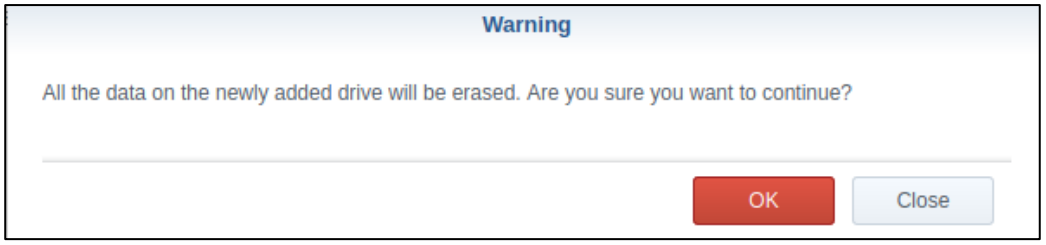


Figure 8.52

06. Check the drives for any bad sectors by selecting “Yes”. Any bad sectors found will be automatically remapped if found. Click Next to confirm the settings and begin the creation of the new storage pool.

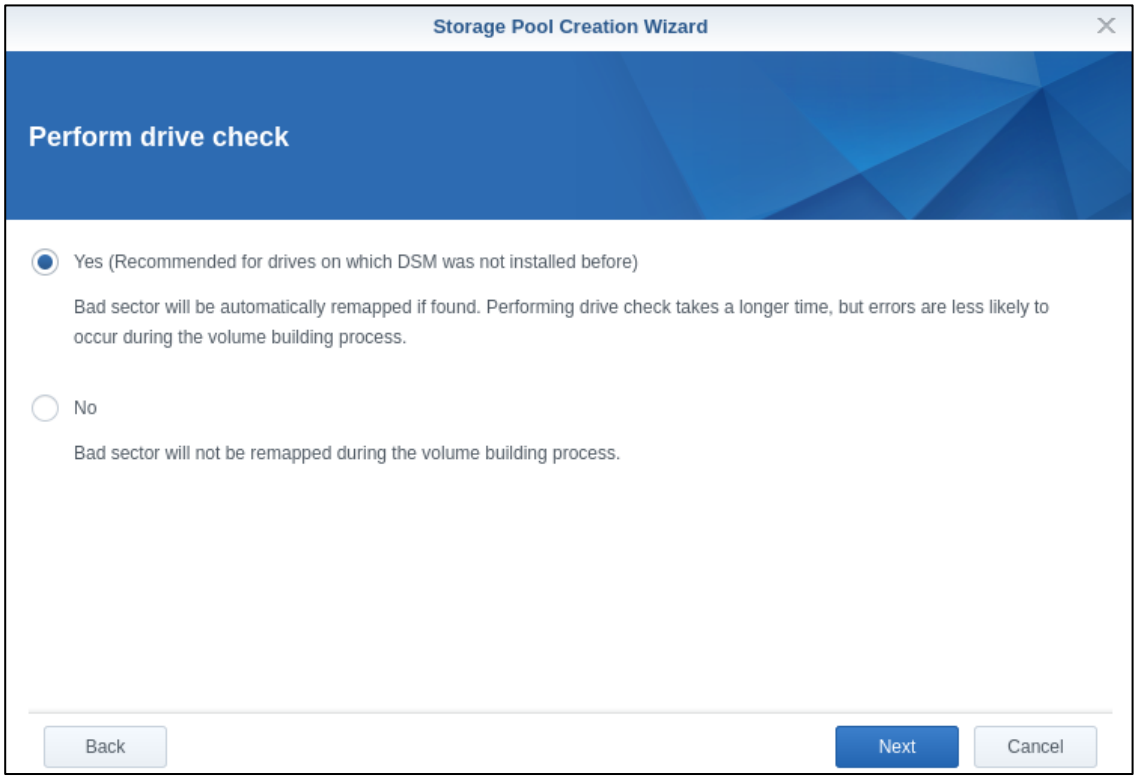


Figure 8.53

07. Confirm the settings that will be applied into the storage cluster. If no changes are to be made, click “Apply”.

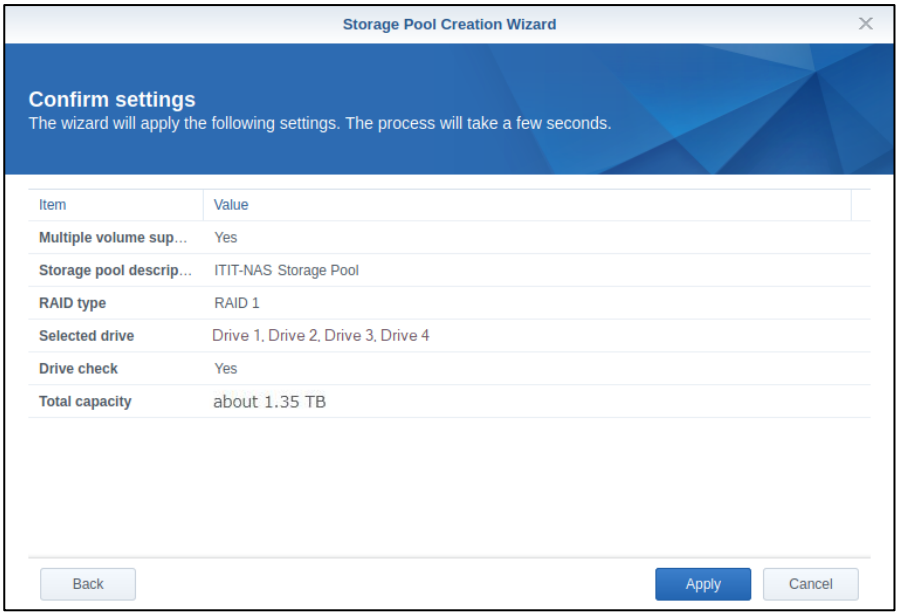


Figure 8.54

08. The storage pool will now begin creation, wait until the disks are scanned for bad sectors, this can take a considerable amount of time. Once the scanning has finished, the storage pool creation is complete.

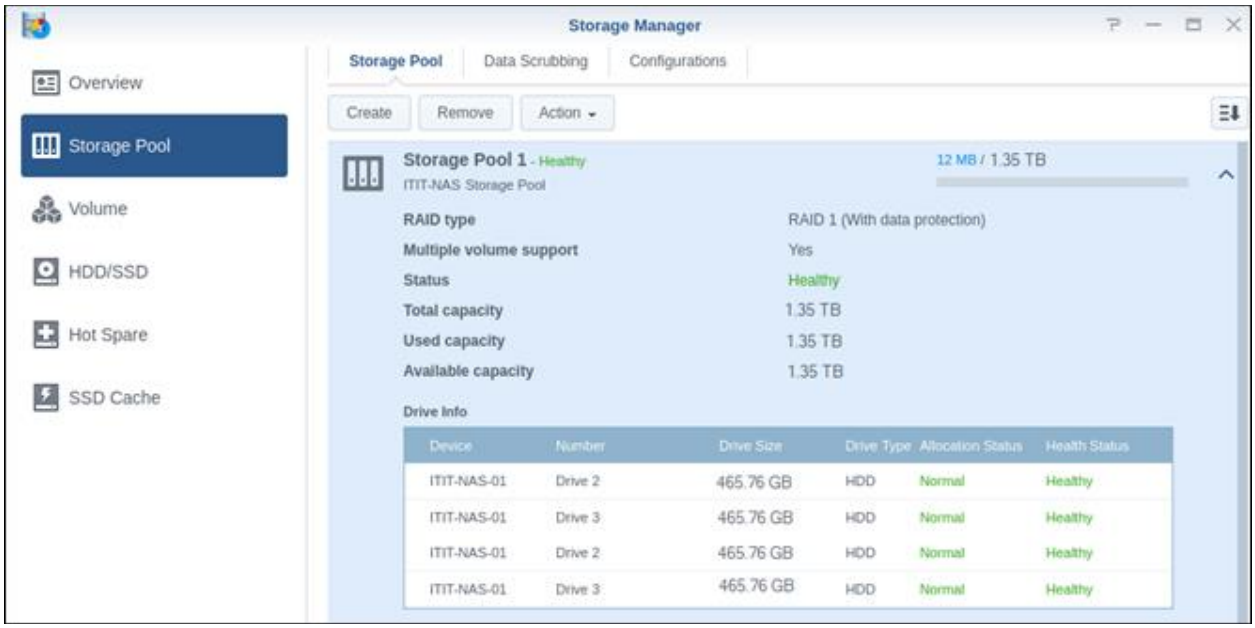


Figure 8.55

The next step would be to create a new storage volume on the pool. In order for data to be stored a volume has to be created. No data can be saved directly on the pool. The creation steps of the storage volume is as follows.

01. In the Storage Manager, select “Volume” and click on the Create button.

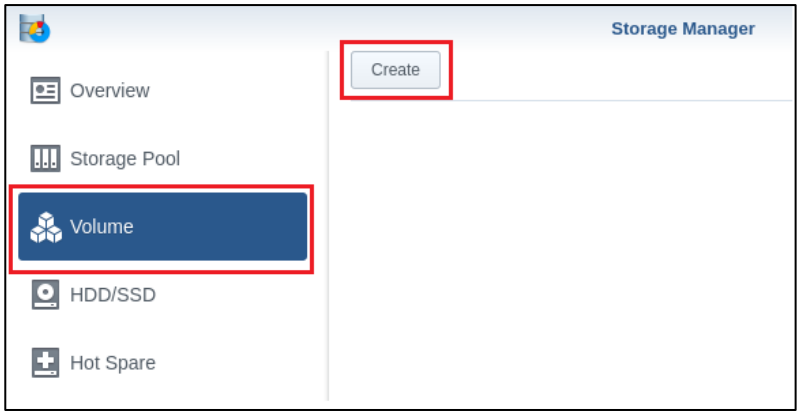


Figure 8.56

02. Select a storage pool for the new volume. Here, the storage pool "ITIT-NAS Storage Pool" created in the previous section is chosen.

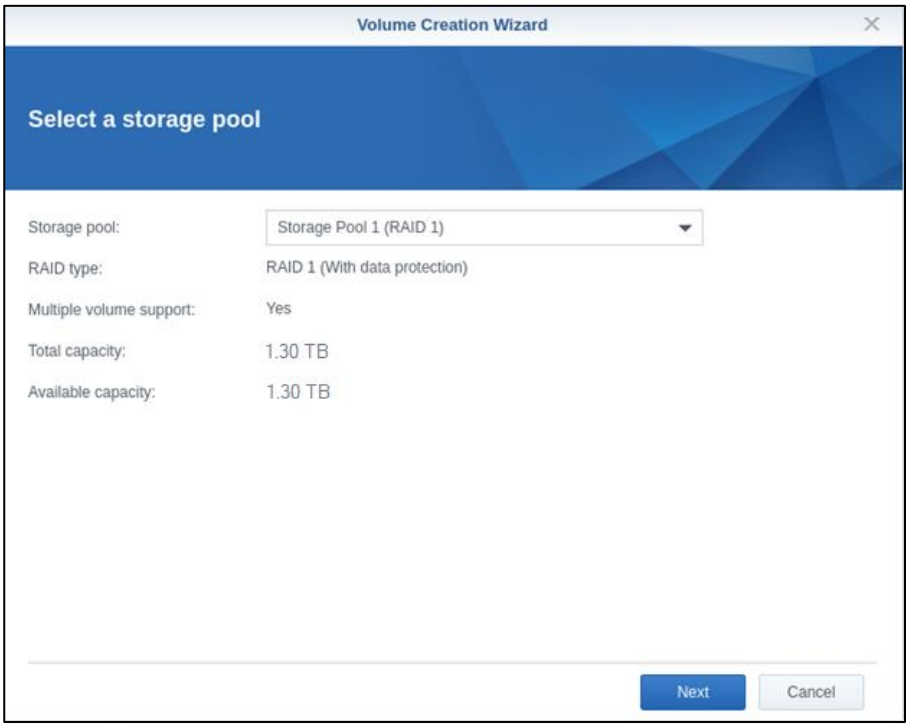


Figure 8.57

03. Select the file system type for the volume. For the current configuration the Btrfs file system was used due to its unique features over ext4 such as self-healing, snapshots, copy-on-write, background file system checks, online defragmentation, and much more. However, ext4 is known to be much more stable and speedy when compared to the much newer Btrfs file system. Click "Next" after selecting the appropriate file system.

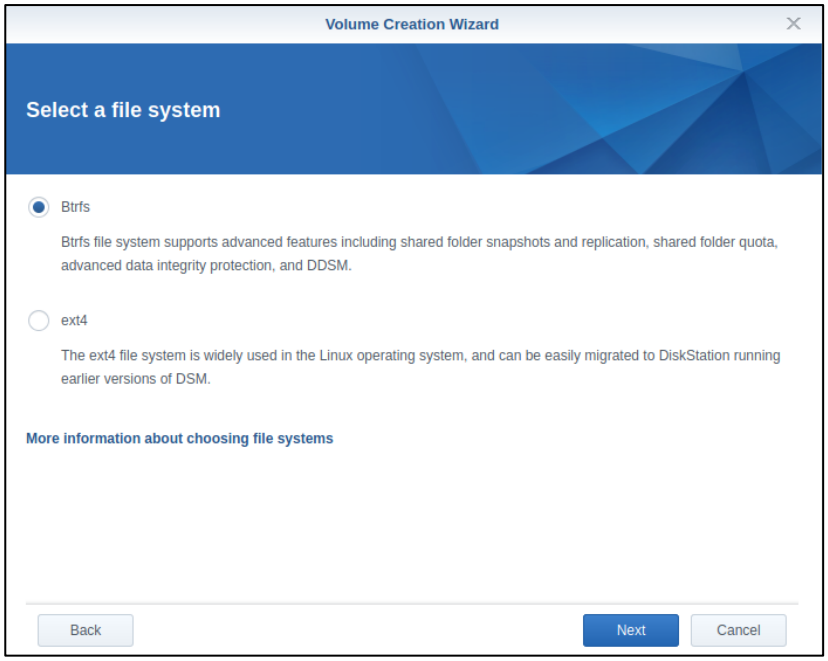


Figure 8.58

04. Allocate the capacity for the volume. The total amount of free capacity in the pool is being allocated to the volume.

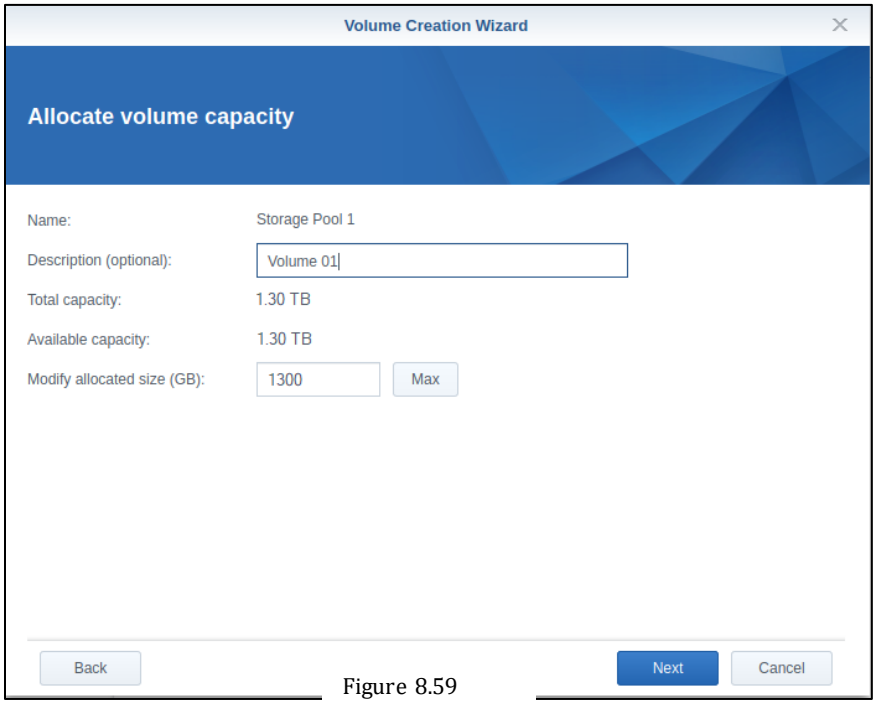


Figure 8.59

05. Review the settings of the volume before it is created. Click on Apply to create the volume when done.

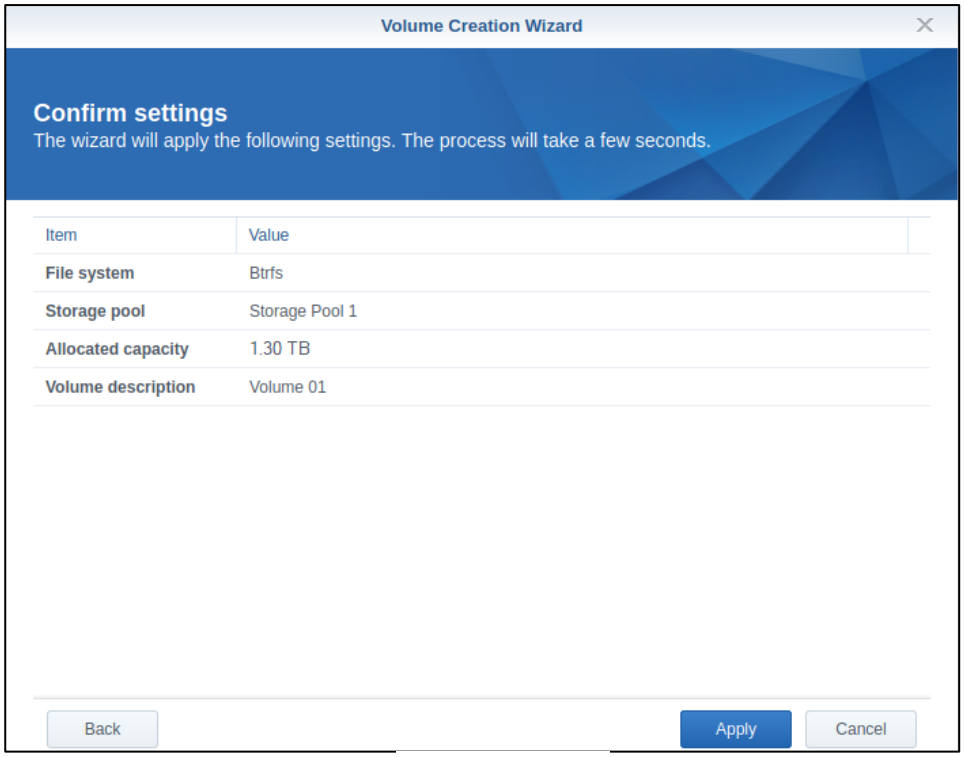


Figure 8.60

06. The volume will be created within a few seconds and will be visible in the Storage Manager's Volume section as follows.

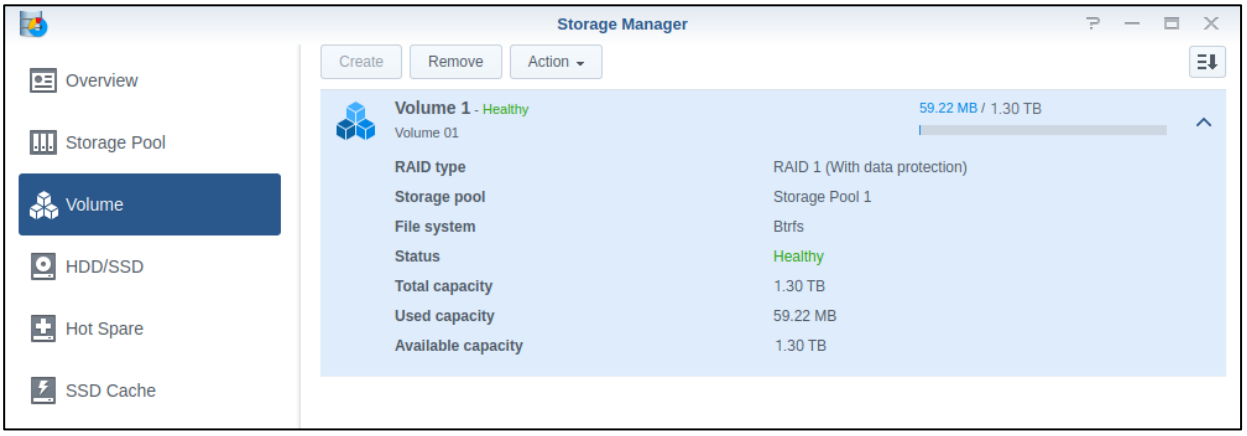


Figure 8.61

The next step is to create an iSCSI LUN (Logical Unit Number) and targets for them. A LUN is needed to act as an identifier for the storage volume and the iSCSI target service links the data storage over an IP network.

01. To begin creating an iSCSI LUN, click on the DSM menu icon, and select iSCSI Managers.

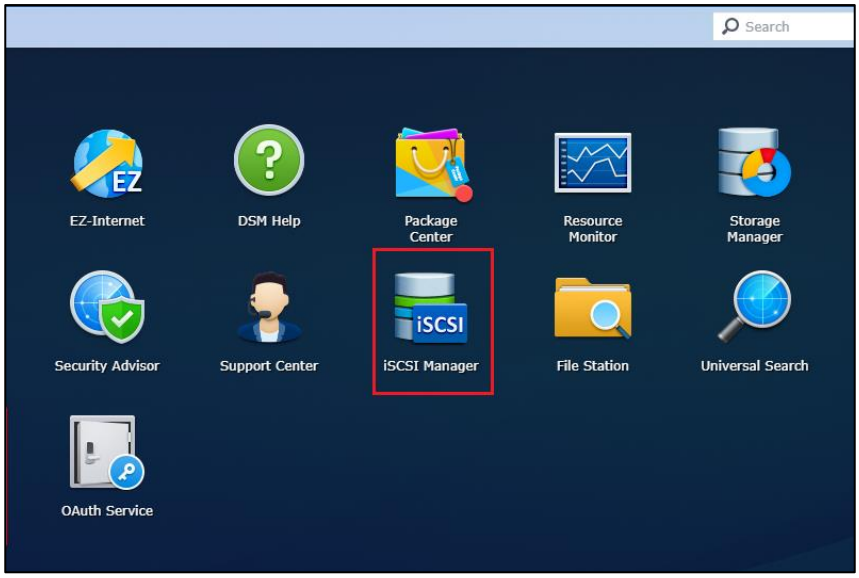


Figure 8.62

01. In the iSCSI Manager window select “LUN” from the menu and click on “Create”.

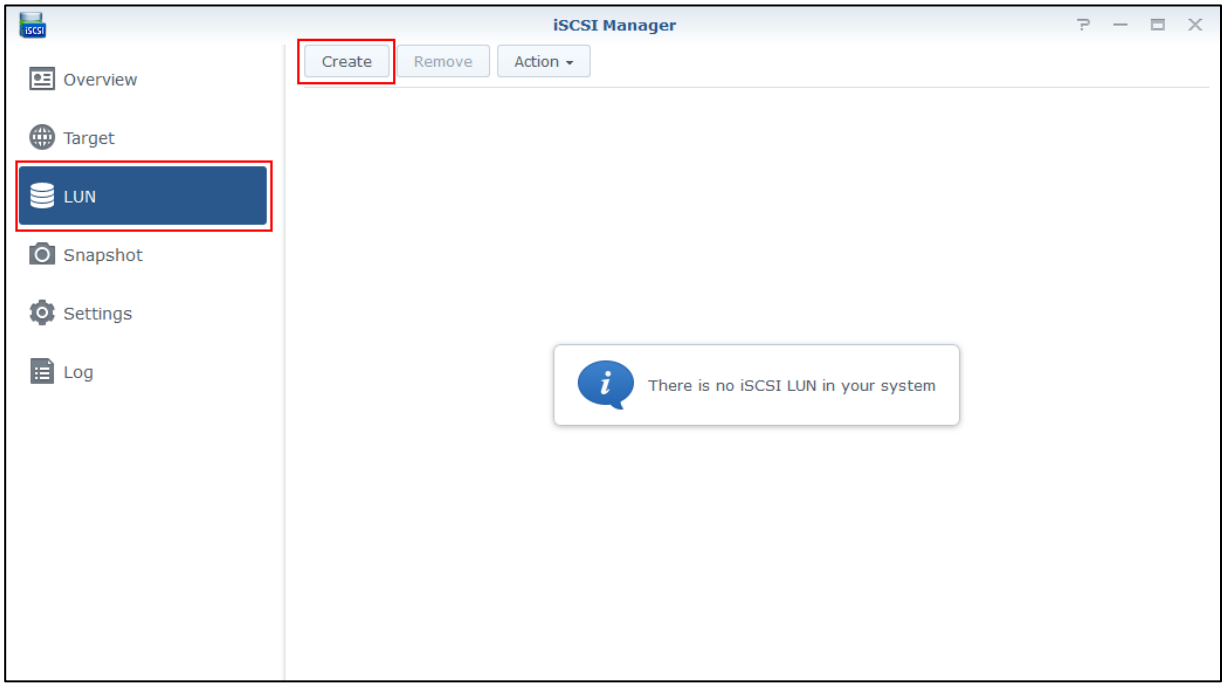


Figure 8.63

02. Provide a name (ITIT-LUN-01), a location (Volume 1), size of the LUN (600GB), and set a space allocation method- Thick or Thin Provisioning. Thin provisioning provides dynamic sizing- that is it only consumes the space that is only needed and can grow or "stretch" when more data is stored. The total capacity here refers to the upper limit of the drive's growth.

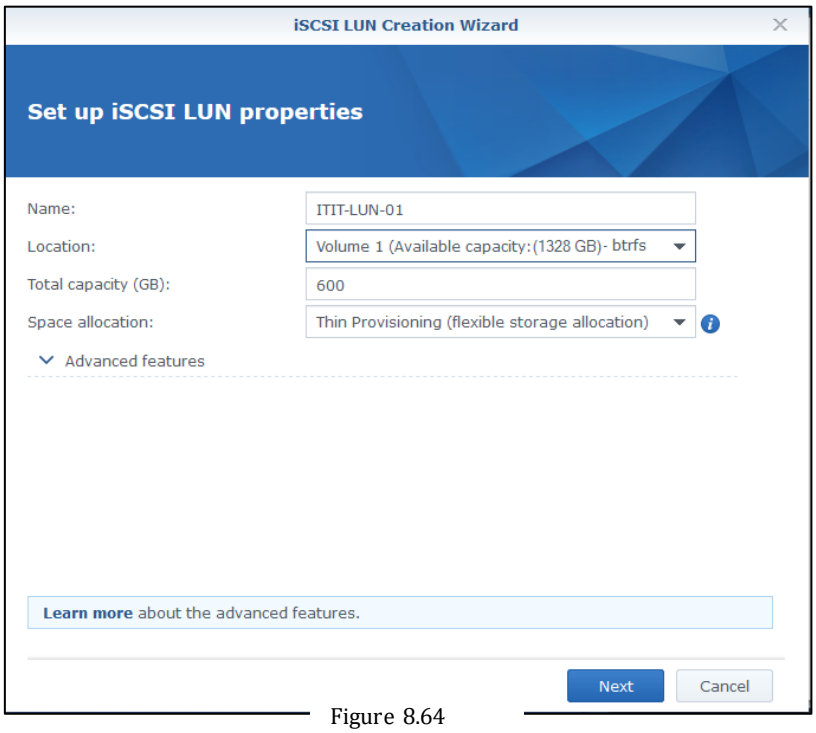


Figure 8.64

03. Select "Create a new iSCSI target" to start the creation of a new iSCSI LUN and target.

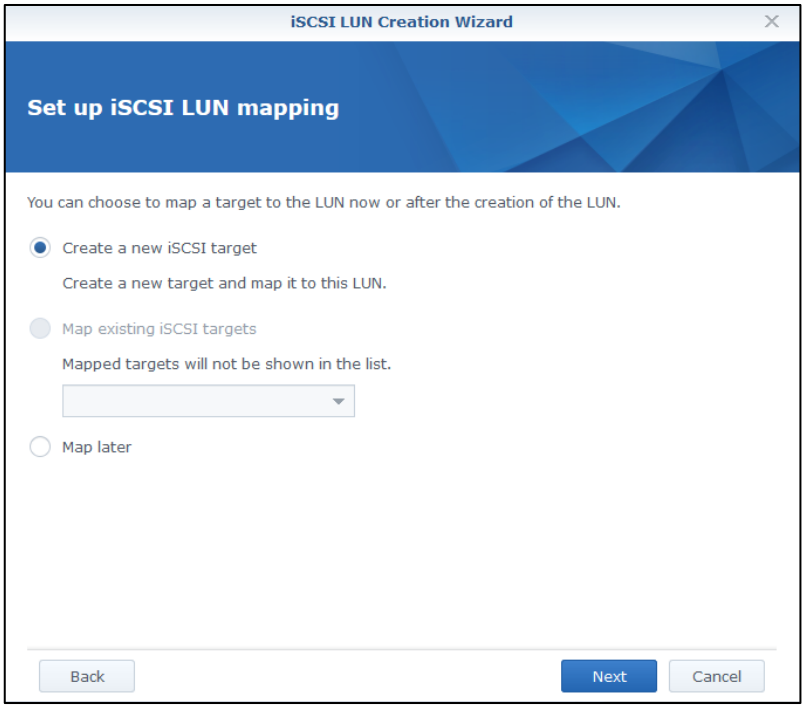


Figure 8.65

04. Enter a name to identify the iSCSI target. An IQN (iSCSI Qualified Number) number is automatically generated with the format "iqn.yyyy-mm.domain:device.ID.", this number is used to identify the iSCSI target. In this scenario, CHAP was not configured to authenticate the iSCSI initiator.

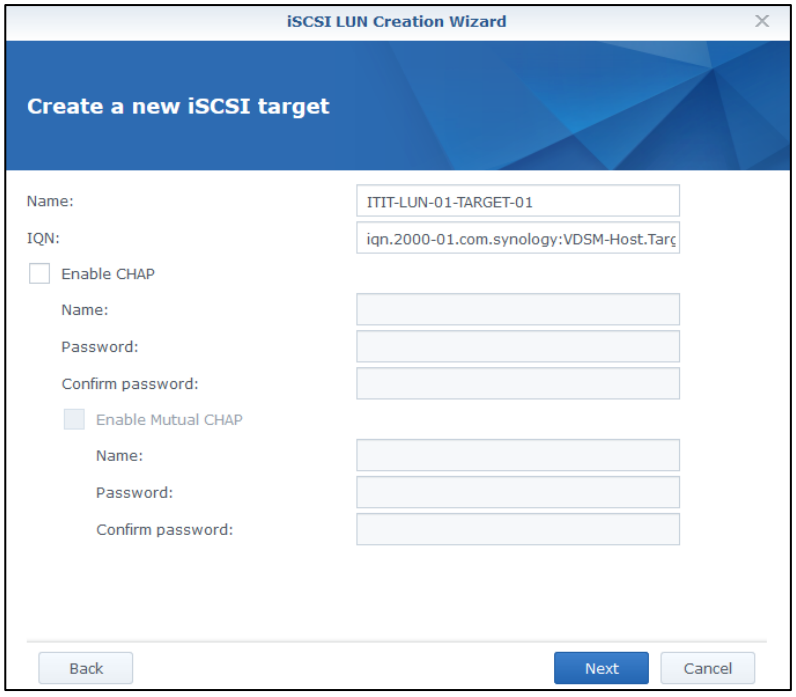


Figure 8.66

05. Confirm the settings and click "OK". The iSCSI LUN and its target will be created in a few seconds.

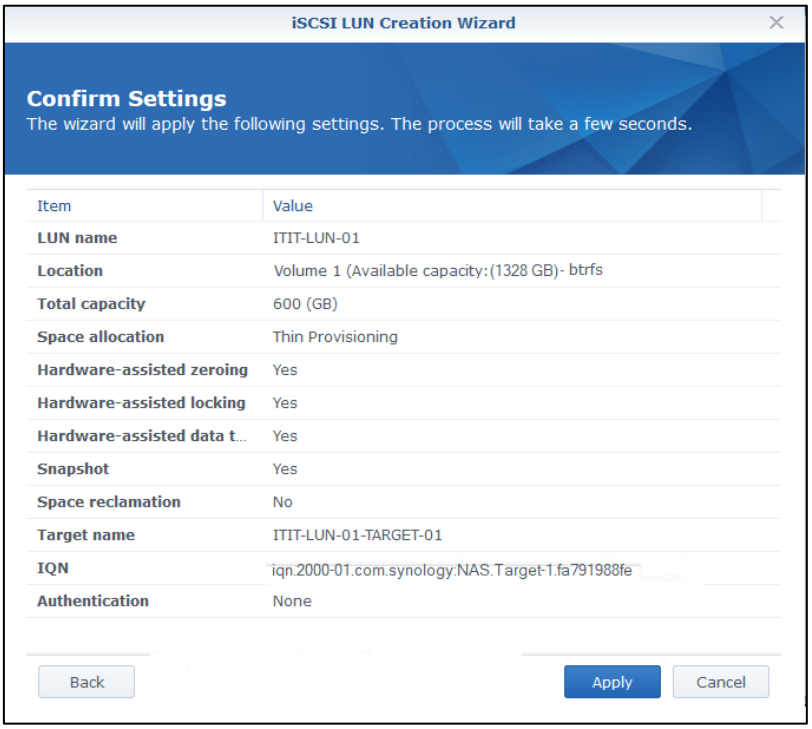


Figure 8.67

06. The newly created iSCSI LUN and target will now be visible in the iSCSI Manager.

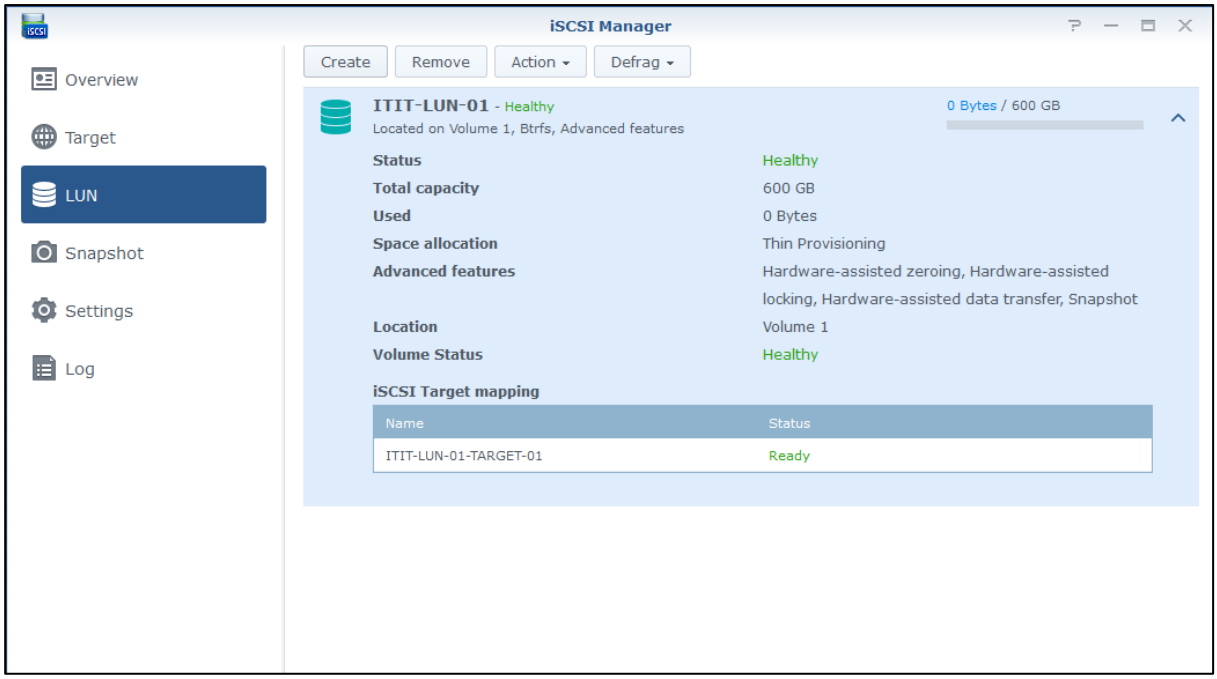


Figure 8.68

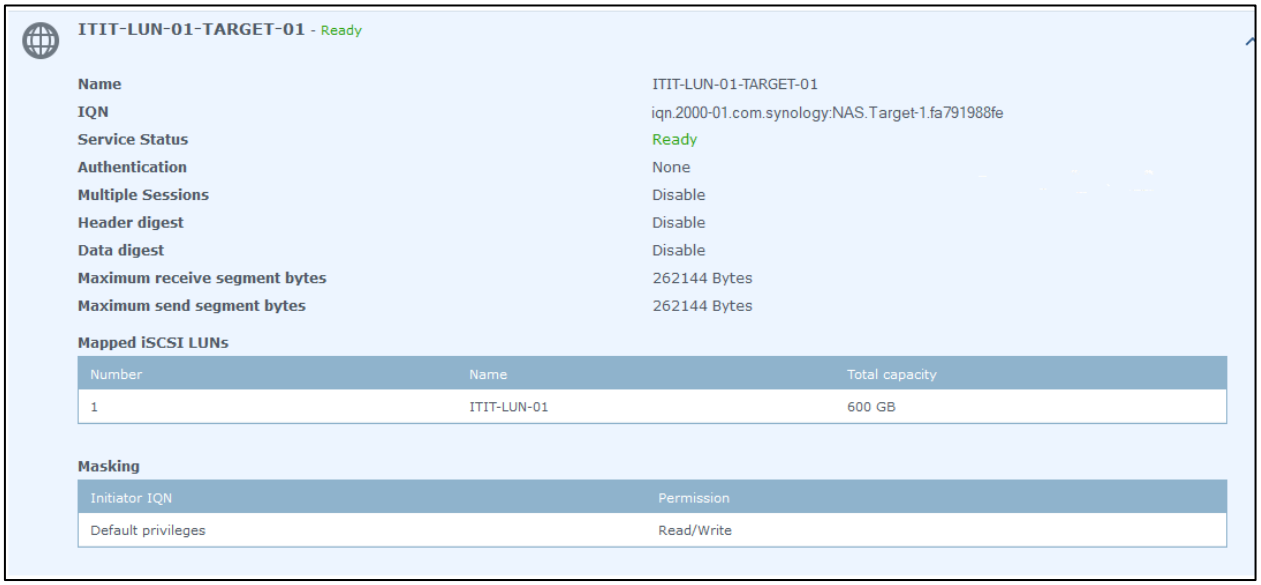


Figure 8.69

In this implementation, two iSCSI LUNs were created on the volume as **ITIT-LUN-01** and **ITIT-LUN-02** with their iSCSI targets set as **ITIT-LUN-01-TARGET-01** and **ITIT-LUN-02-TARGET-02**. The second LUN (ITIT-LUN-02) was configured in the same procedure as the first. The configuration schemes for the second LUN are shown in the next page.

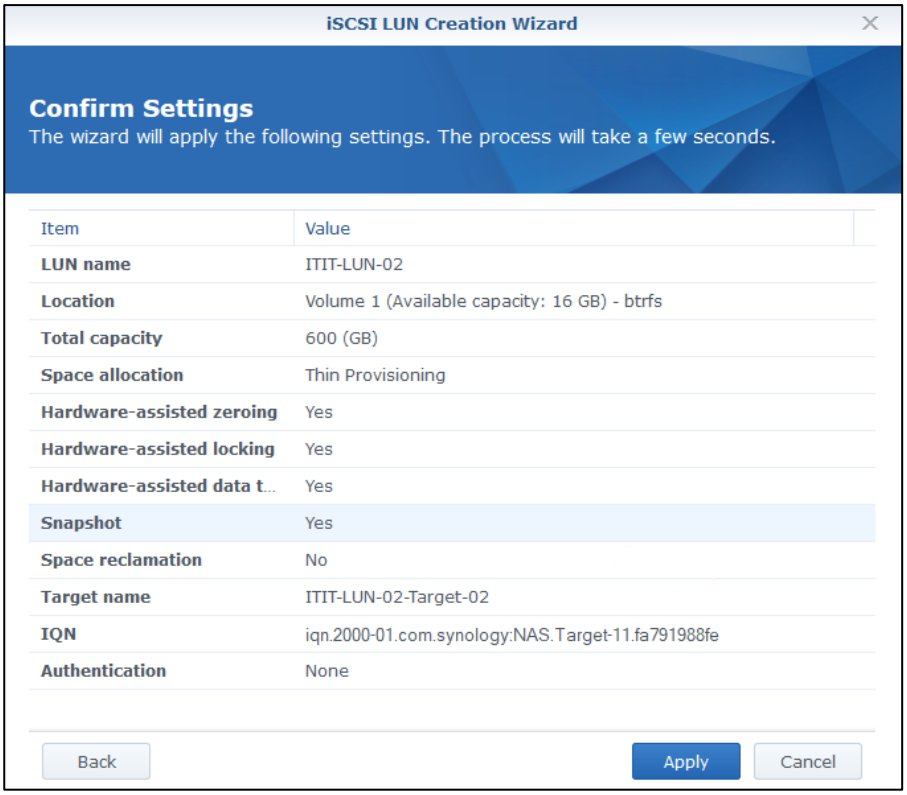


Figure 8.70

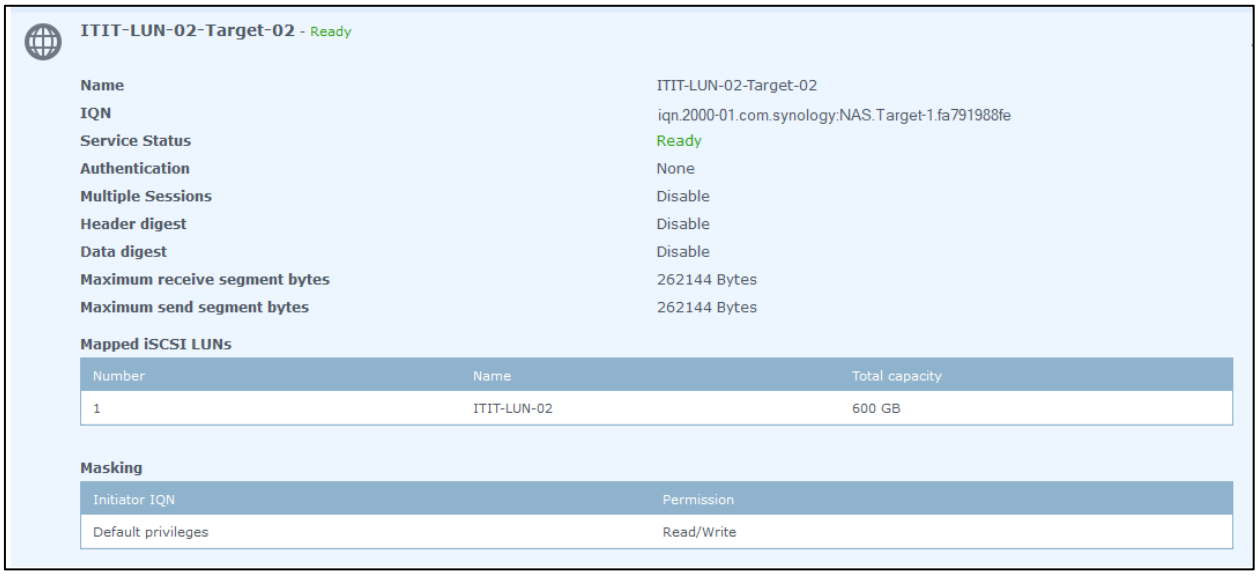


Figure 8.71

In the above configurations of both iSCSI LUN targets the "Multiple Sessions" feature has been disabled. This must be enabled to allow multiple iSCSI initiators (in this case, the two VMWare ESXi hosts) to connect to the same target concurrently. Do note that in order to enable this feature and to get it work without errors the file system must be hooked onto a cluster aware file system.

07. In the Target section click on the "Action" button and select the "Edit" option.

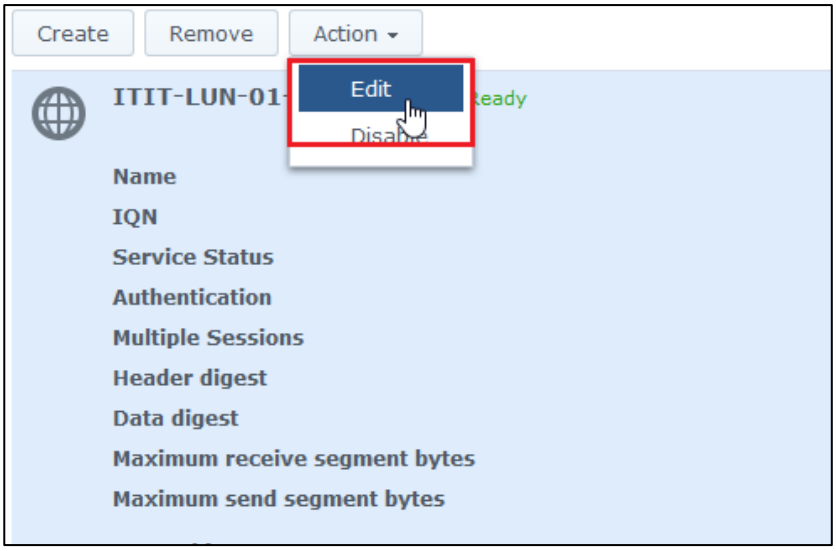


Figure 8.72

08. Click on the "Advanced" tab and tick the option to allow Multiple Sessions and click OK.

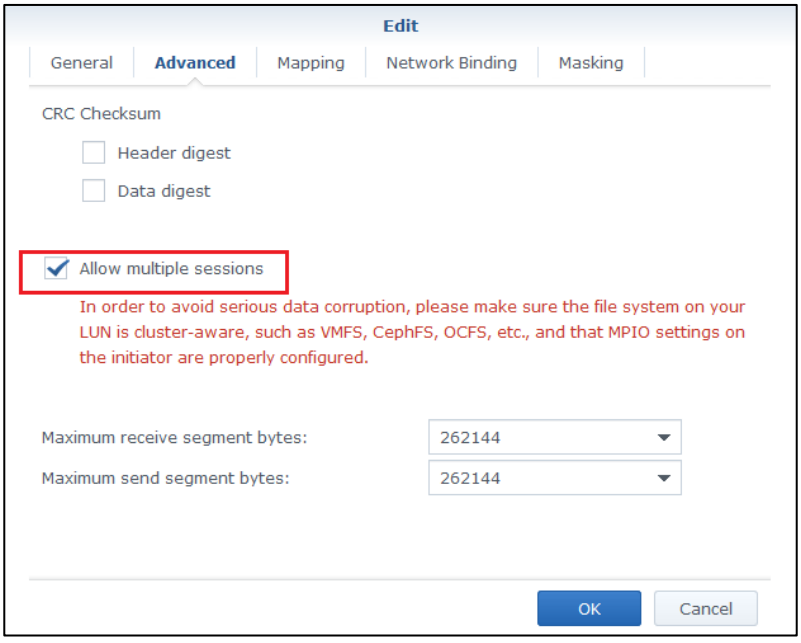


Figure 8.73

09. Enable multiple sessions in the second LUN target (ITIT-LUN-02-TARGET-02) in the same manner.

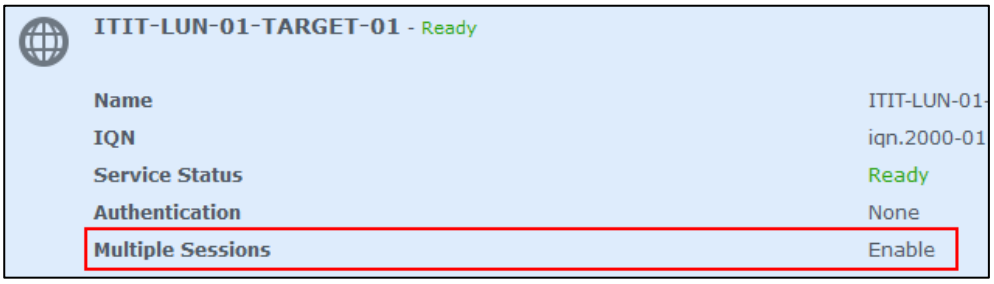


Figure 8.74

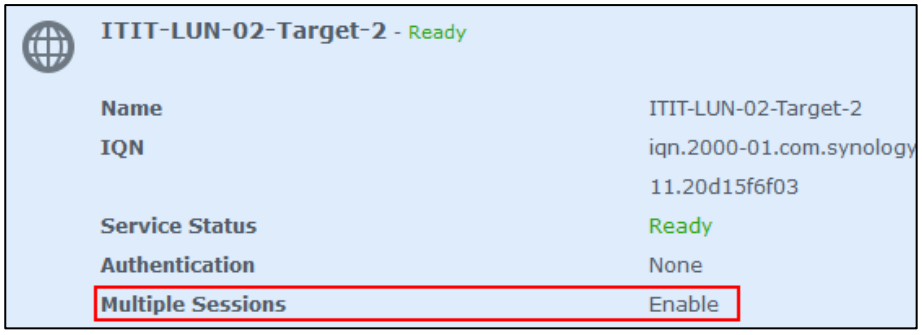


Figure 8.75

Visit the Overview section of the iSCSI manager to ensure that the iSCSI LUNs are performing well and have no issues.

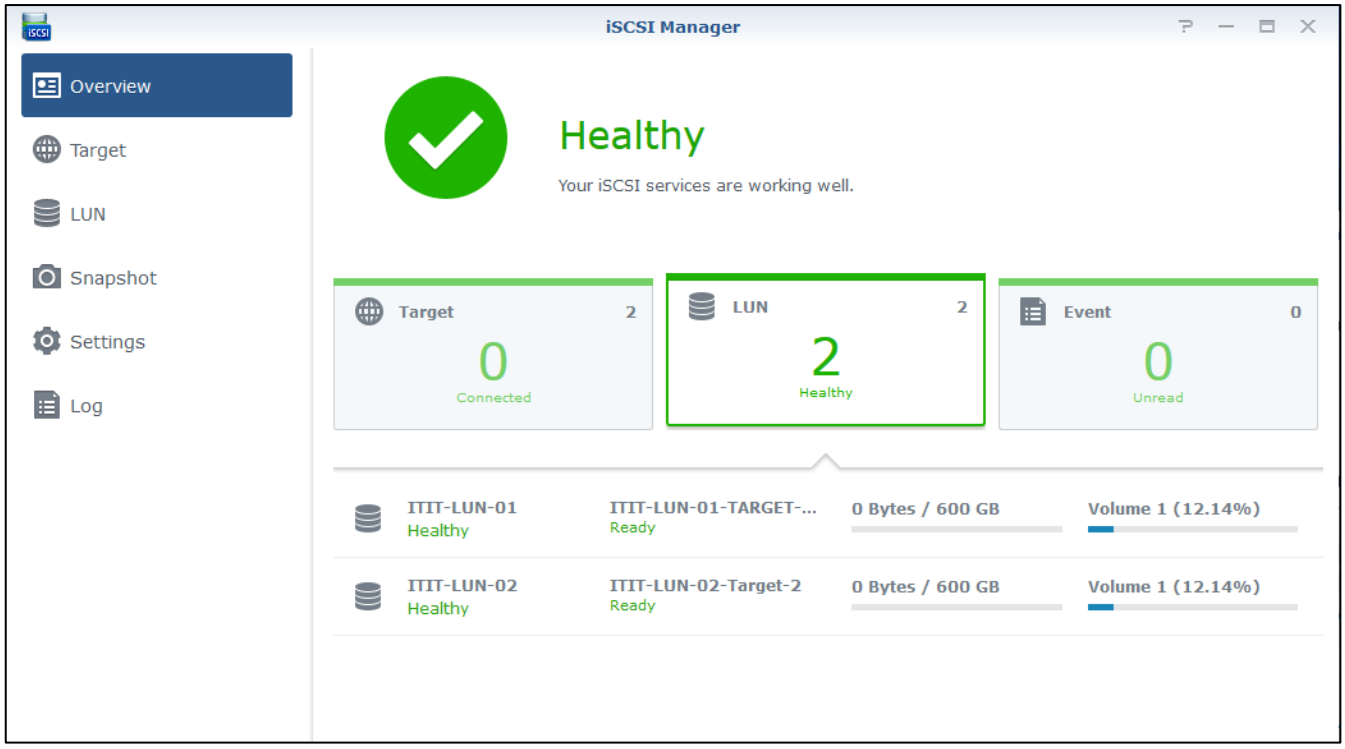


Figure 8.76

The iSCSI LUNS are now available to be used as datastores for the VMWare ESXi hosts.

Configuring Network Attached Storage as VMWare Datastores

Before setting up the network attached storage cluster as a datastore for the ESXi hosts the following prerequisites must be satisfied.

- Initial configurations of the NAS has been configured properly (page XX-XX)
- Created iSCSI LUNs and targets (page XX-XX)
- Installed VMWare ESXi 6.7.0 on the hosts (page XX-XX)
- Access to the ESXi Web UI

If the above mentioned settings are already made, the following configurations can be made to configure the storage cluster as a datastore.

01. Log in to the Web UI of the ESXi host

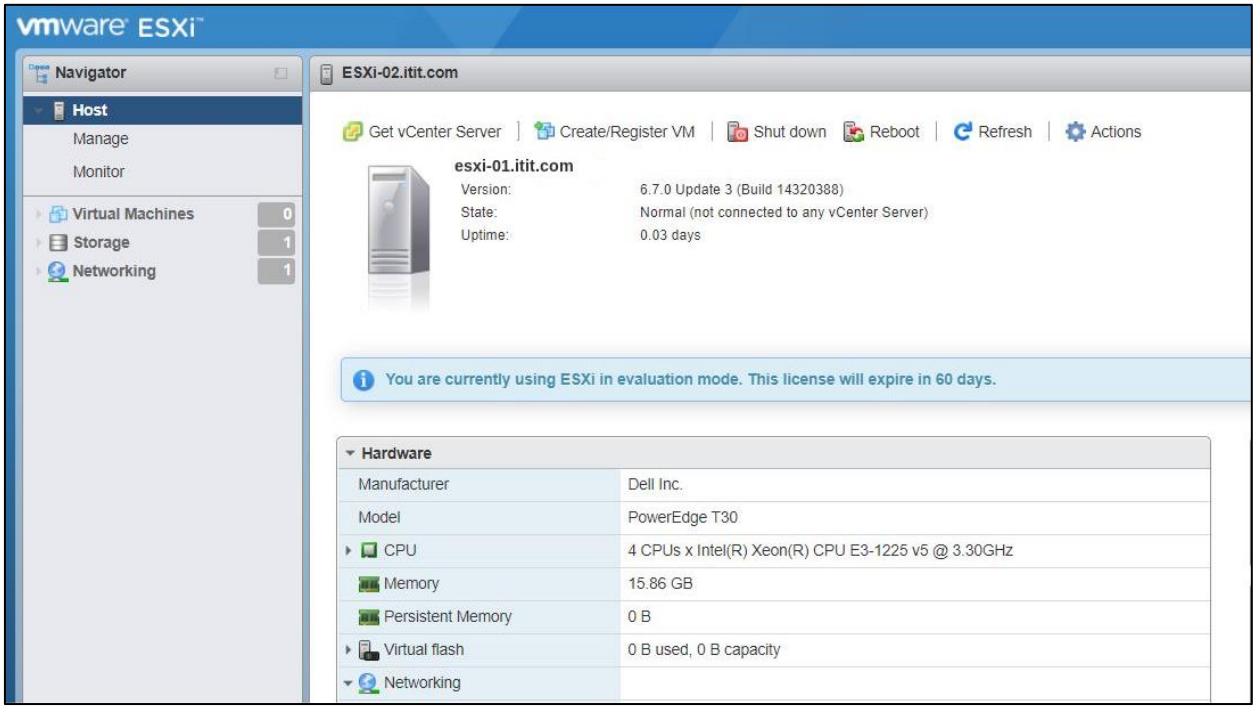


Figure 8.77

Note that in the Navigator section, it shows that 1 datastore is available under Storage, this is the local hard drive of the ESXi host.

02. From the Navigation menu, select "Networking", click on the "Virtual Switches" tab, and then click "Add standard virtual switch" to create a new virtual switch.

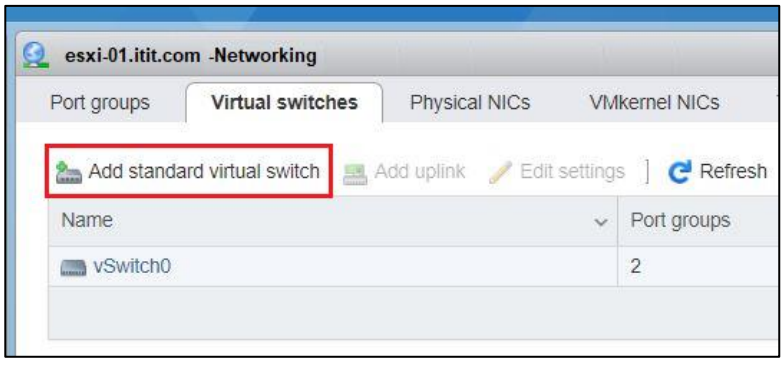


Figure 8.78

03. Set a name for the virtual switch ("ITIT-VSW-01"), set an MTU size (In general, an MTU of 1500 is set for a 1GbE network and 9000 is set for a 10GbE network), and select an uplink for the virtual switch. Click the add button to complete the virtual switch creation.

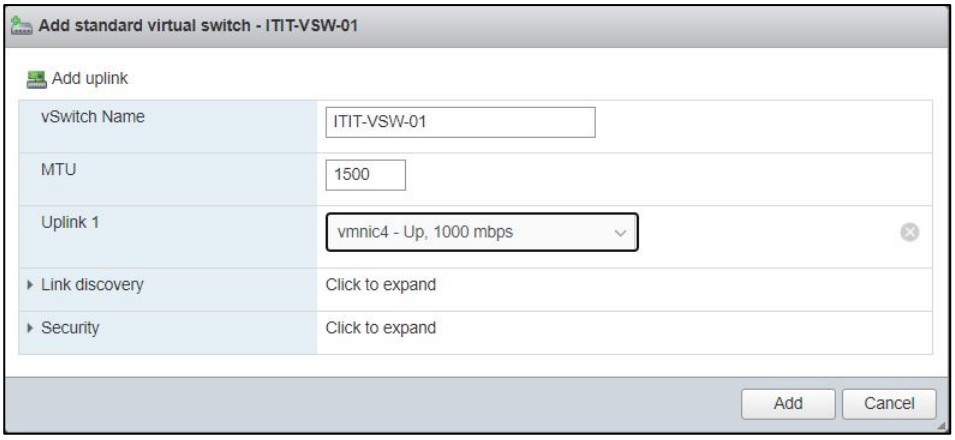


Figure 8.79

04. Once the virtual switch is created, Select the Port groups tab, and click on "Add port group"

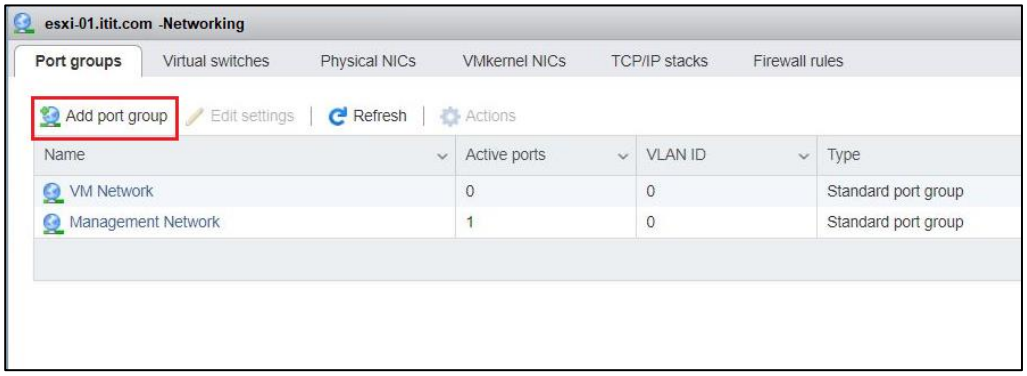


Figure 8.80

05. Enter a name for the new port group (ITIT-PG-01), and the name of the virtual switch name (ITIT-VSW-01) to be associated with the port group. The port group facilitates the addition of virtual switch ports available in a virtual switch. Click "Add" button to create the new port group.

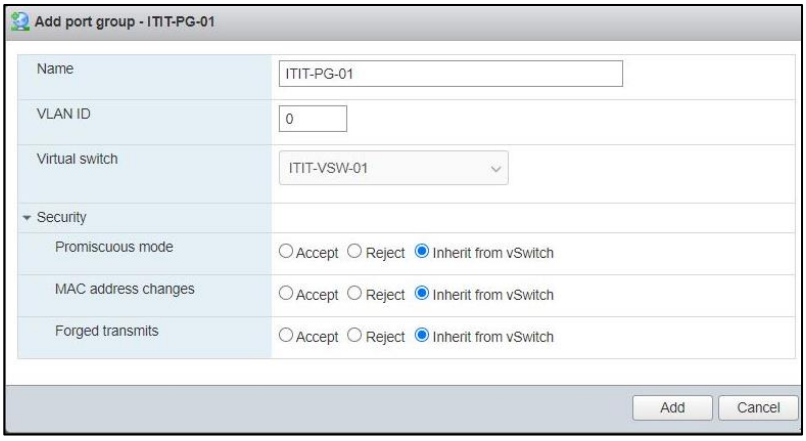


Figure 8.81

06. Click the VMKernel NICs tab and click on the Add "VMkernel NIC" to create a new VMkernel network adapter. VMkernel adapters are required to connect the virtual hosts to the physical network, to handle management traffic and IP-based storage traffic.

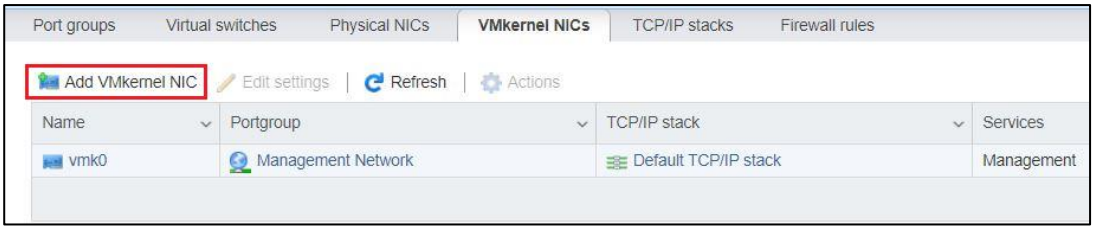


Figure 8.82

07. Enter a name of an interface to be addressed to the port group (ITIT-PG-01) and a static IP address for the VMkernel adapter. It is through this adapter that the communications between the virtual network and the physical storage network will happen. Click "Create" to iniitiate the creation of the VMkernel NIC. In the below configuration vMotion has not been enabled, vMotion is required to provide migration of virtual resources from one location to another.

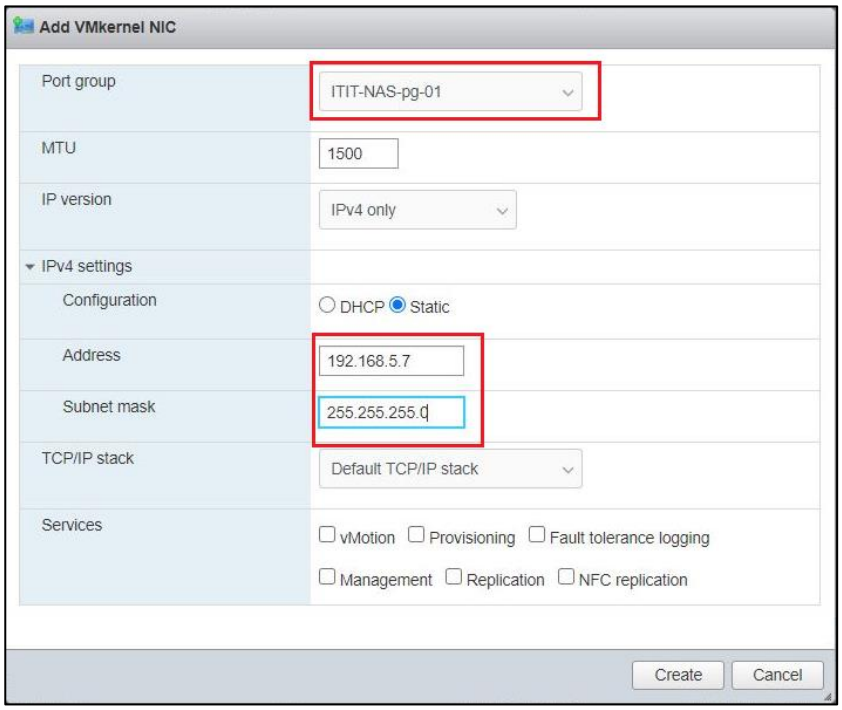


Figure 8.83

Now the ESXi host is ready to connect to the NAS devices through an iSCSI software interface. The next step will be to mount the iSCSI targets on to the host.

08. In the Navigation menu, click "Storage" and select the "Adapters" tab and click on the "Configure iSCSI" option.

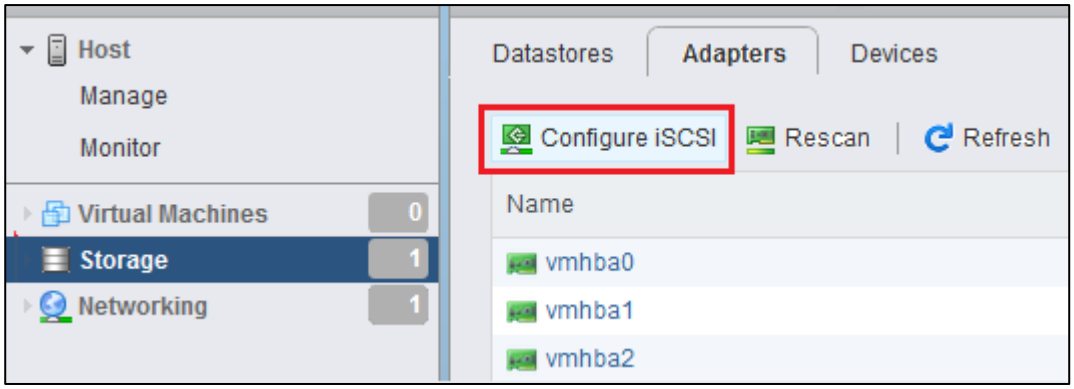


Figure 8.84

09. In the next window, enable iSCSI by selecting the "enabled" option. Once the iSCSI initiator has been enabled, click "**Add port binding**" to add the VMkernel NIC that will be used to connect to the storage cluster. Set the storage cluster address (192.168.5.5) as the dynamic target. Dynamic targeting shows all the available iSCSI targets for a given IP address or hostname. Click the "Save Configuration" button to finsish.

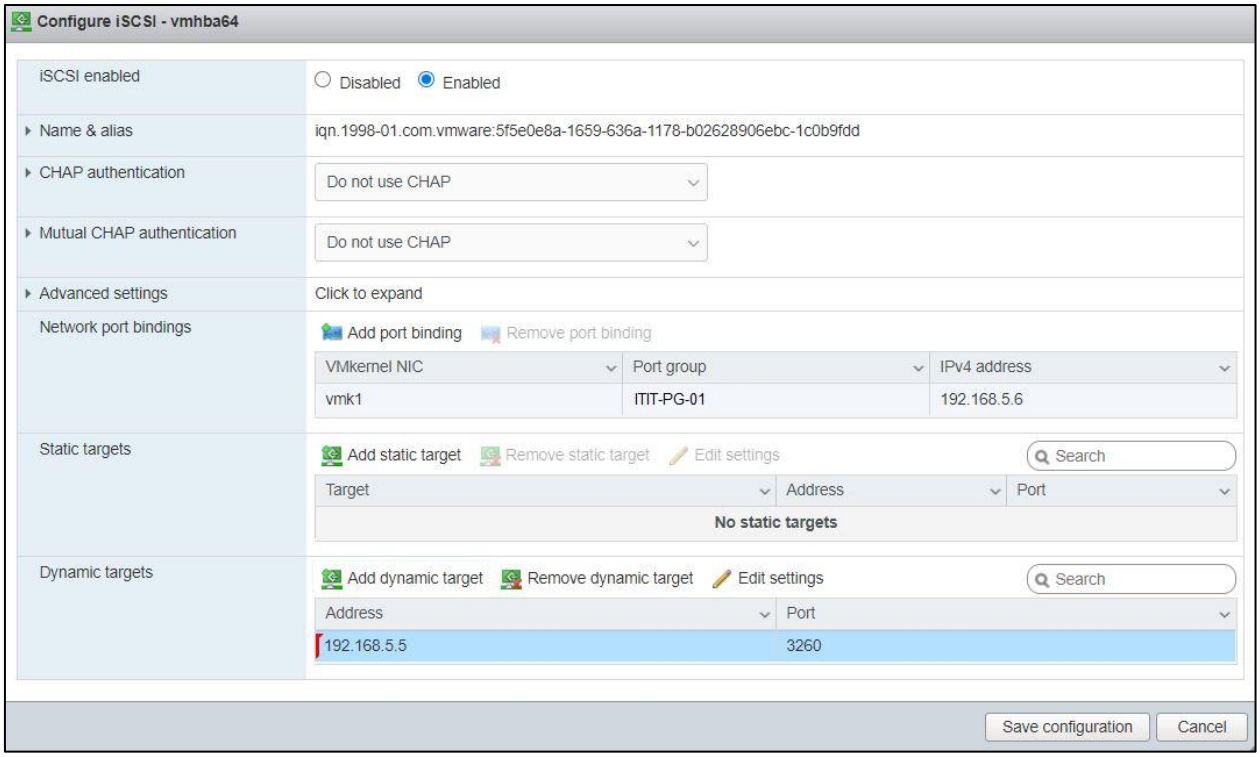


Figure 8.85

10. Click on the "Devices" tab in the Storage menu and hit the Refresh button. Confirm that the new devices are scanned and visible by iSCSI adapter.

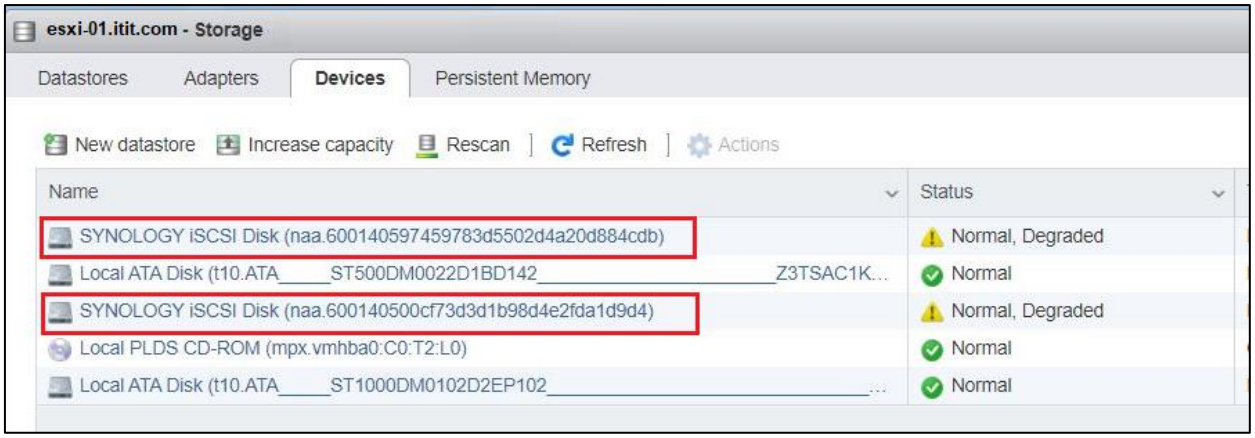


Figure 8.86

11. The next step is to set up the iSCSI targets as datastores for the ESXi hosts. Go to the Datastores tab and click "New Datastore". The datastore creation wizard will start and guide throughout the process. In the first step, select "Create new VMFS Datastore" and click Next.

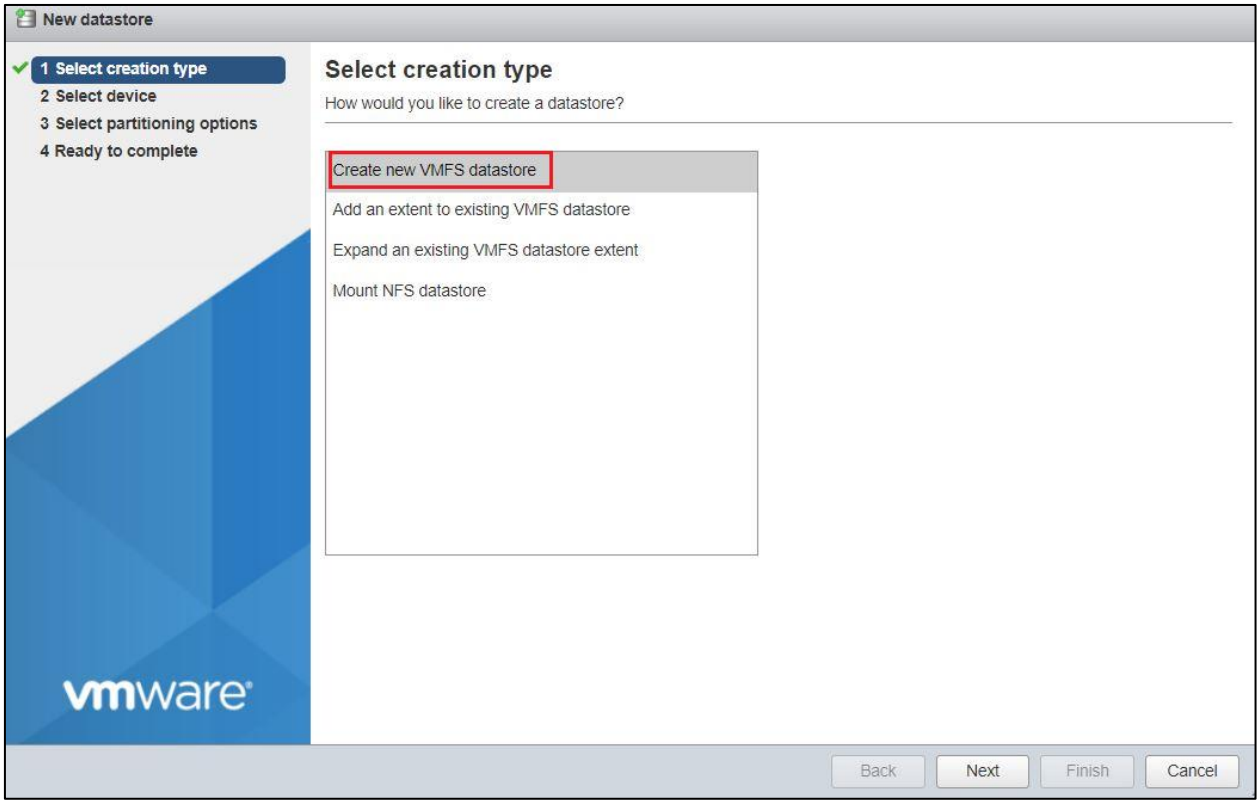


Figure 8.87

12. Select an iSCSI LUN and provide a name for the datastore.

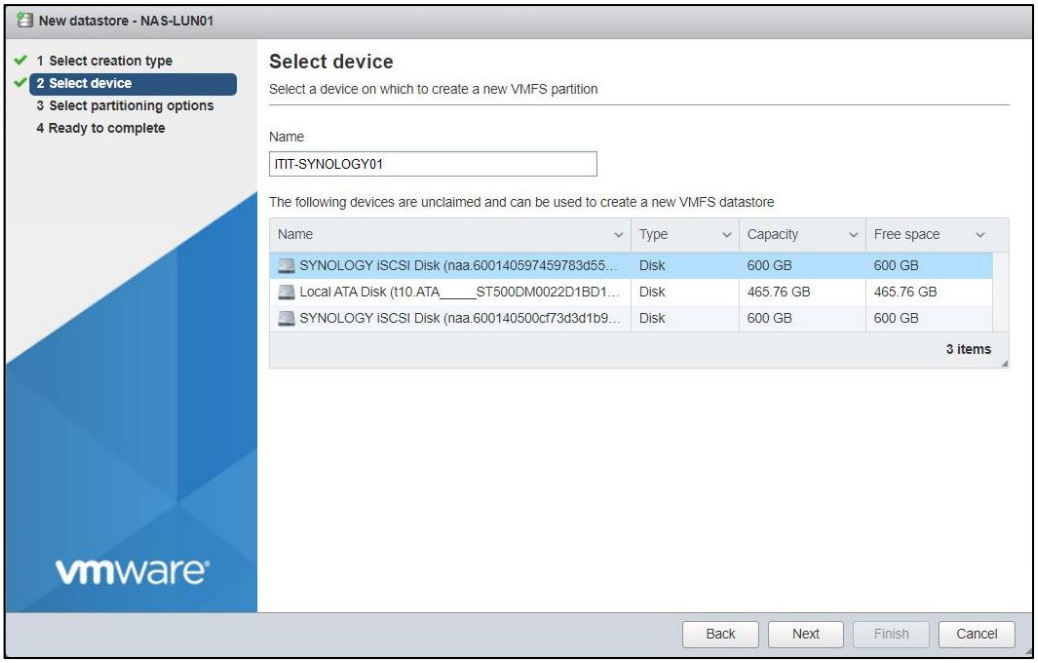


Figure 8.88

13. Check the current disk layout and click Next. In this implementation, the iSCSI LUN was not partitioned and the full disk was used.

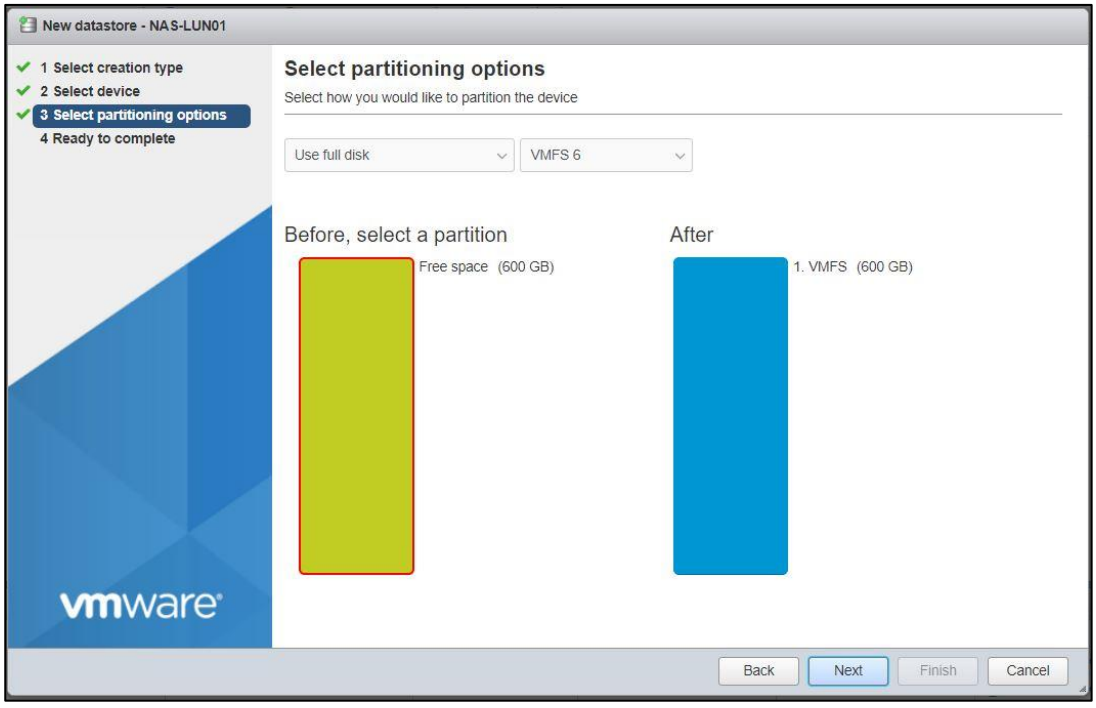


Figure 8.89

14. Check the datastore settings and click Finish to confirm settings.

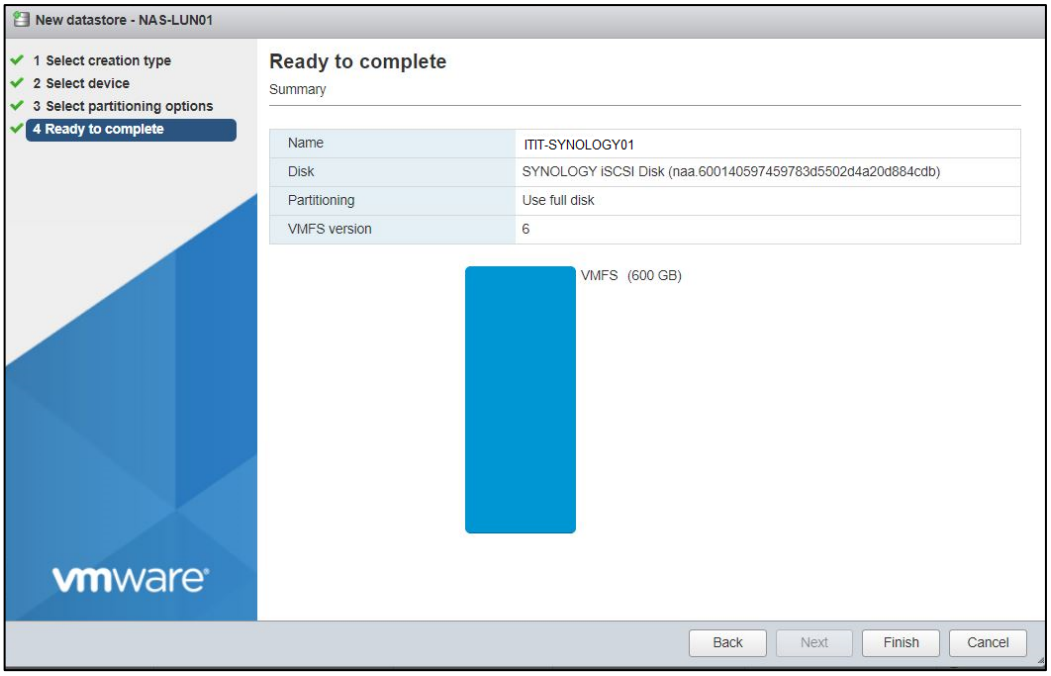


Figure 8.90

15. Create another VMFS datastore for the second iSCSI LUN as shown in steps 11-14. Once done, the two units should be shown as datastores under the devices tab as available disks.

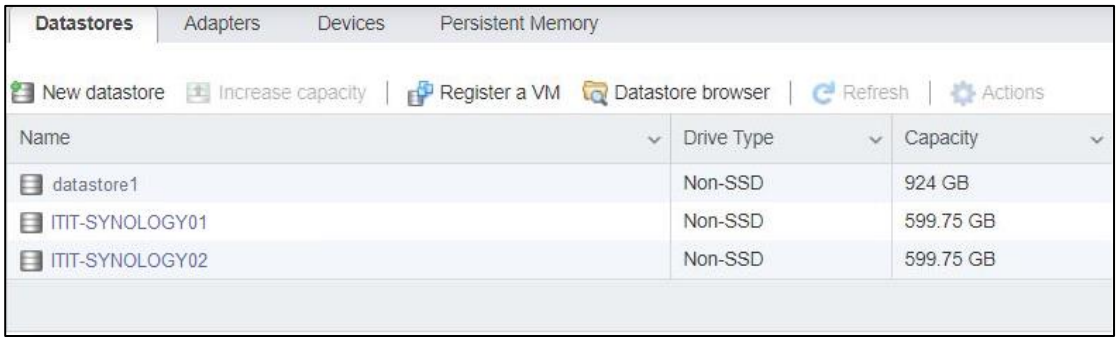


Figure 8.90

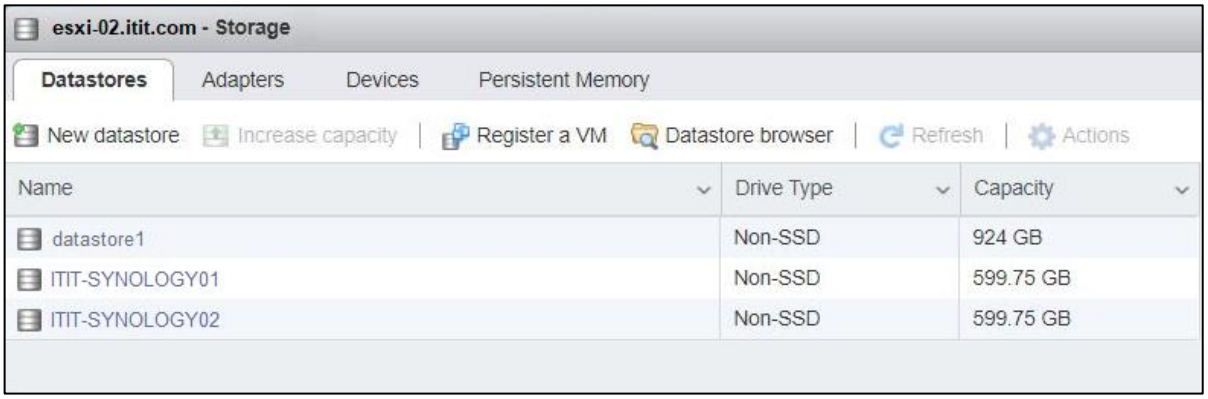


Figure 8.91

16. The other VMWare ESXi host must also be configured with a virtual switch, a port group for that virtual switch, and add a VMKernel NIC of the same storage subnet into the port group. There will be no need to create another datastore, the two datastores (ITIT-SYNOLOGY01

and ITIT-SYNOLOGY02) will be available for use. Below shows the two datastores shared into the second ESXi host (esxi-02.itit.com)

The reason for this is once a datastore has been configured from one ESXi hosts to act as a shared iSCSI datastore it will automatically be shared among the other ESXi hosts with the same virtual network settings configured.

Installation of vCenter Server

vCenter Server is a centralized management utility that is used to manage and monitor virtual machines and multiple ESXi hosts in a VMWare virtualized environment. vCenter Server provides many advanced features vSphere High Availability, vSphere Fault Tolerance, vSphere Distributed Resource Scheduler (DRS), VMware vSphere vMotion, and VMware vSphere Storage vMotion.

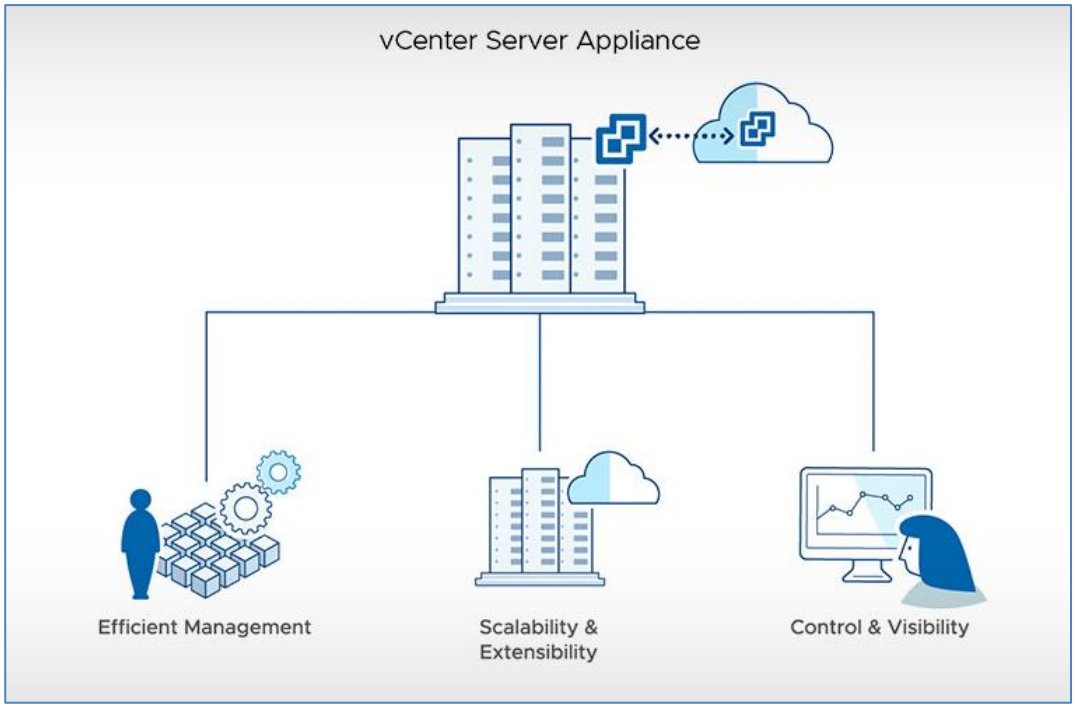


Figure 8.92

The deployment process of the vCenter Server Appliance is for the campus server network is documented below.

01. From the VMware Web site (https://my.vmware.com/web/vmware/downloads), download the vCenter Server Appliance installer.

02. Run the installer within a network client machine and provide inputs needed for the appliance deployment.

03. The first stage of the deployment is called OVA Deployment. This stage completes the deployment of the OVA file on the target server with the deployment type and appliance settings provided in the deployment wizard.

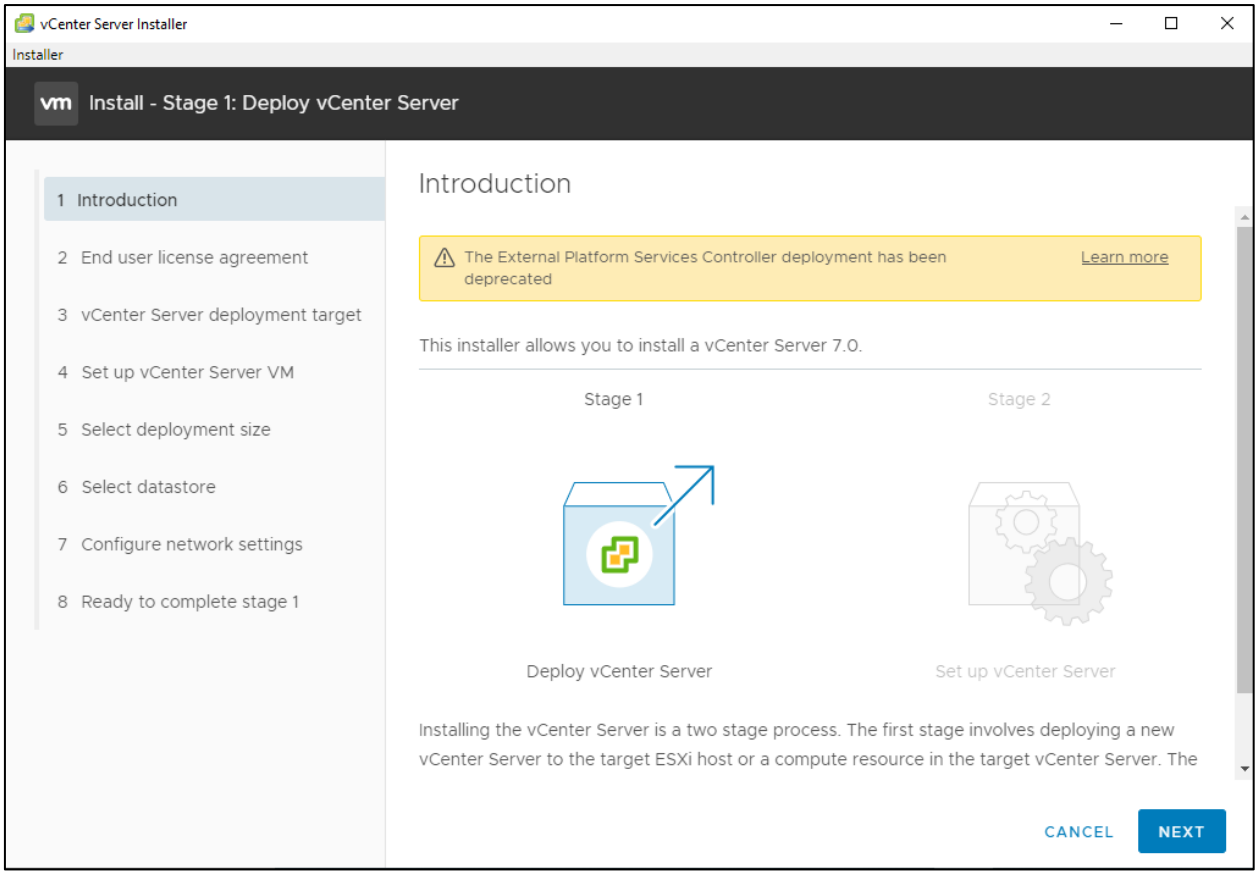


Figure 8.93

04. Read and accept the End User License Agreement.

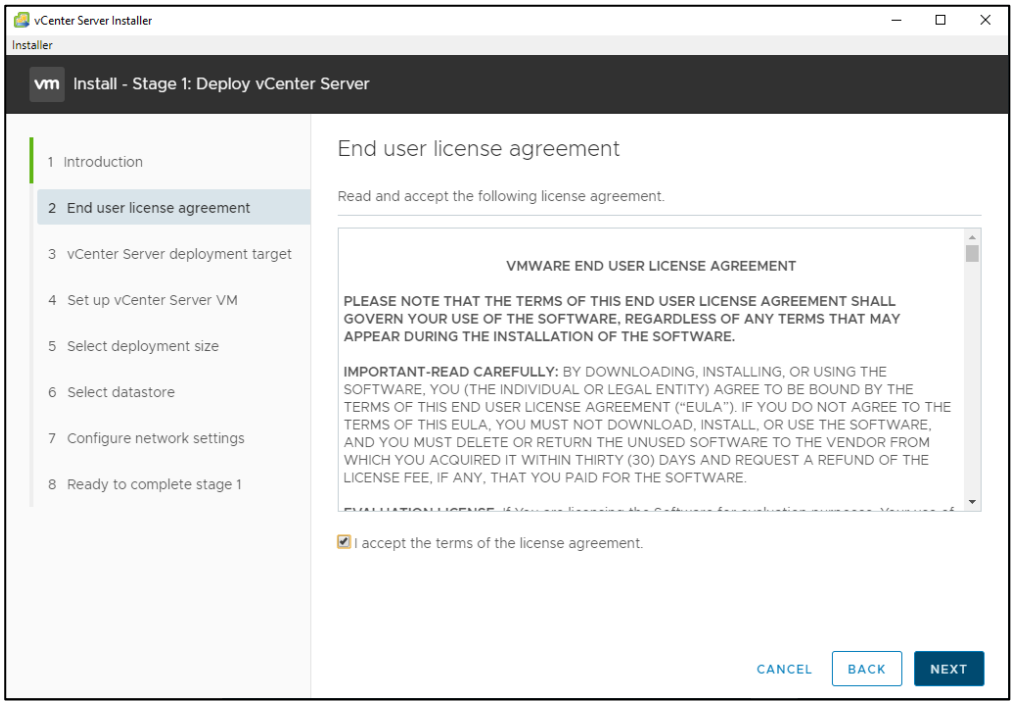


Figure 8.94

05. Enter the target to deploy the vCenter Server. In this scenario, the first ESXi host (esxi-01.itit.com) was used.

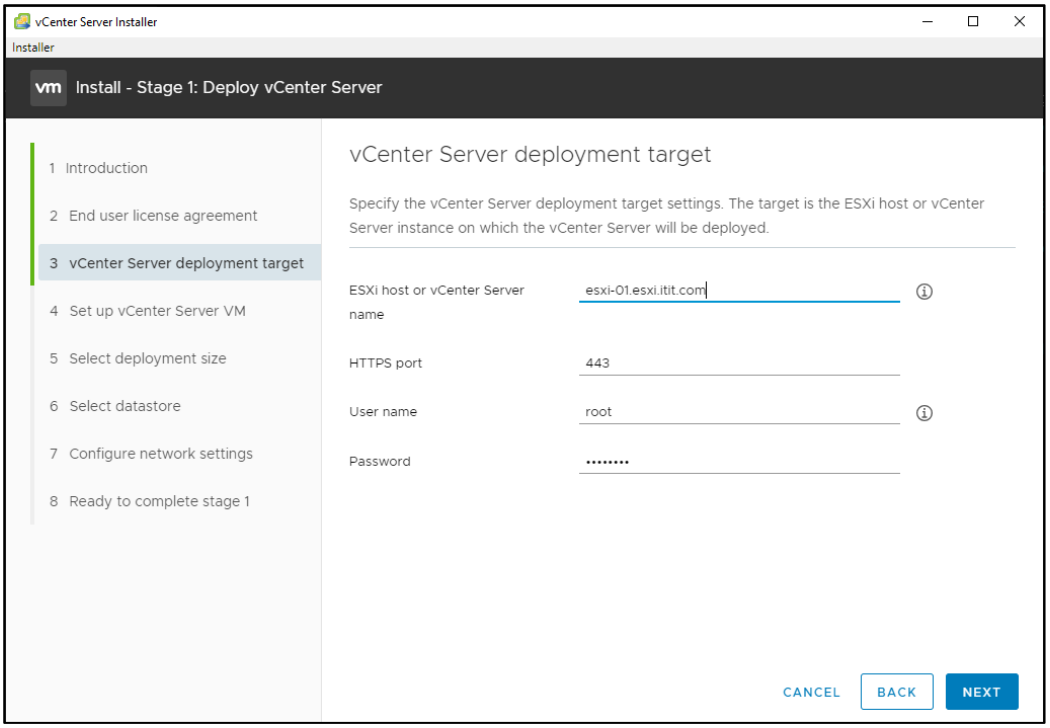


Figure 8.94

06. Enter the details for the vCenter virtual machine. The vCenter Server will run on this virtual machine which will run on the before mentioned ESXi host.

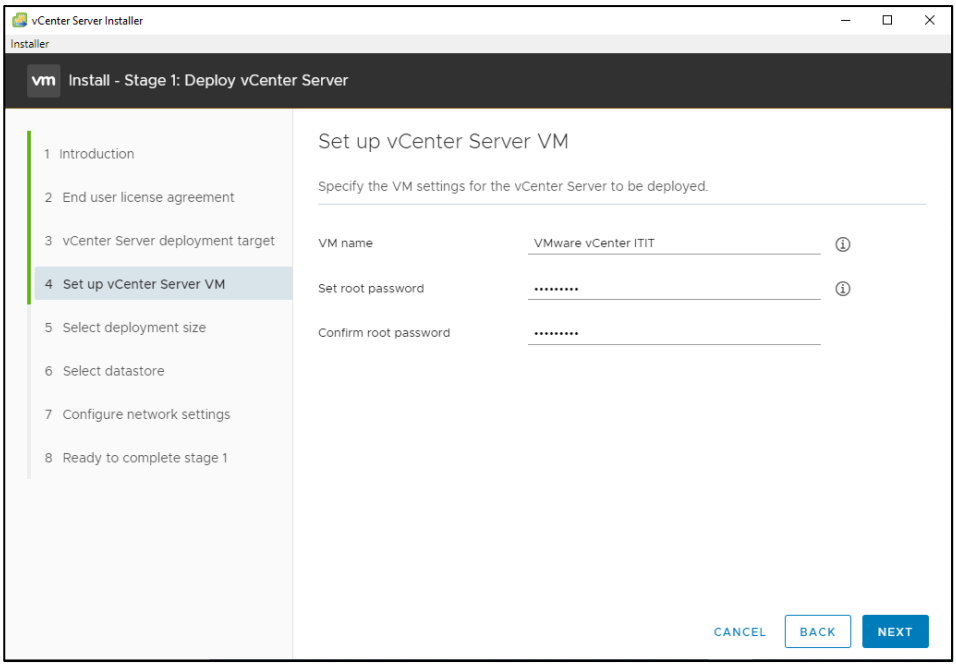


Figure 8.95

07. Choose a deployment size for the vCenter server. For this implementation, the size “Tiny” was chosen. The Tiny deployment is suitable for environments up to 10 hosts or a 100 virtual machines.

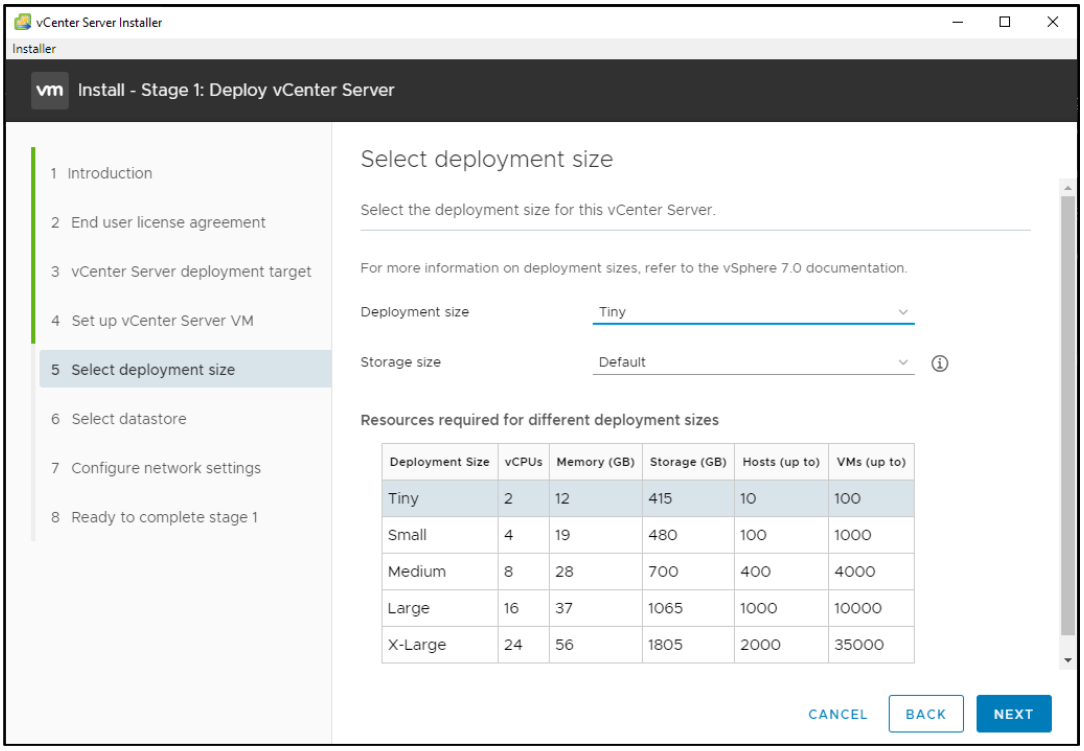


Figure 8.96

08. Select a datastore to provide storage for the vCenter virtual machine, be sure to select one of the iSCSI LUNs from the network storage devices.

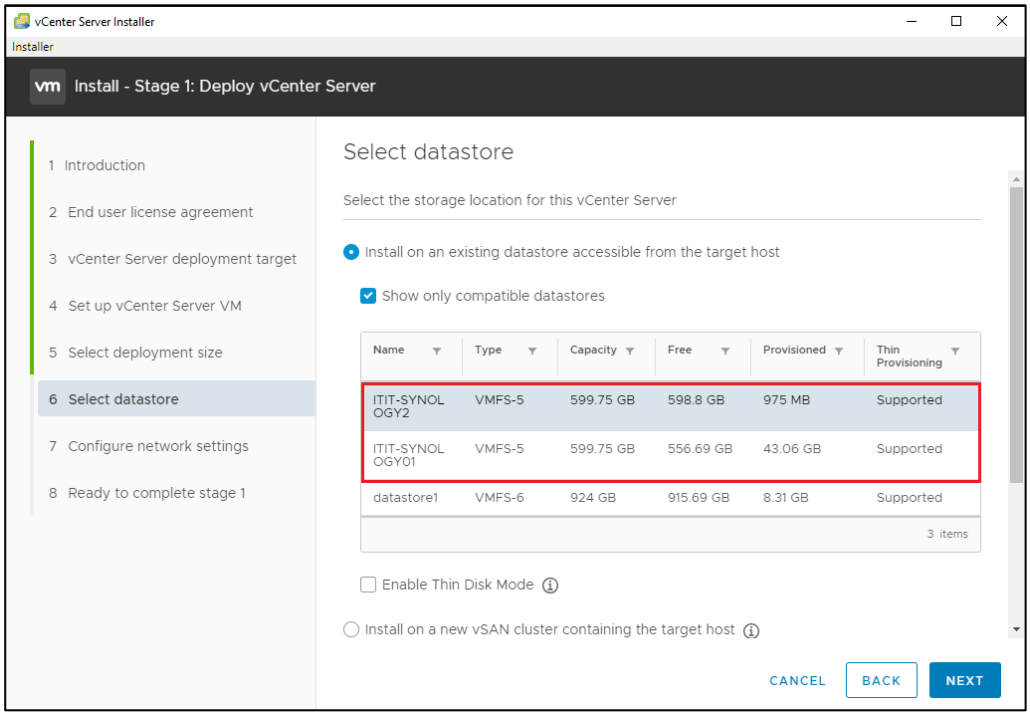


Figure 8.97

09. Configure the network settings for the vCenter Server virtual machine. Be sure to provide it with static IP addressing.

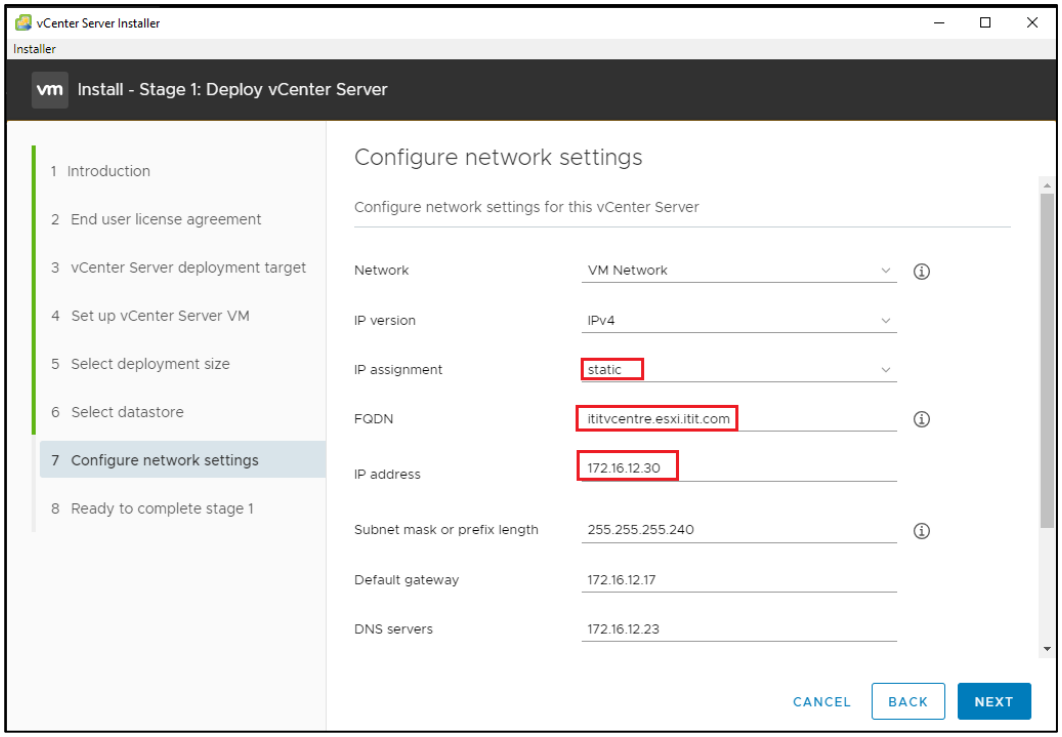


Figure 8.98

10. The vCenter Server will now be ready to be deployed. Check if theinformation provided is correct and click Finish to start the deployment.

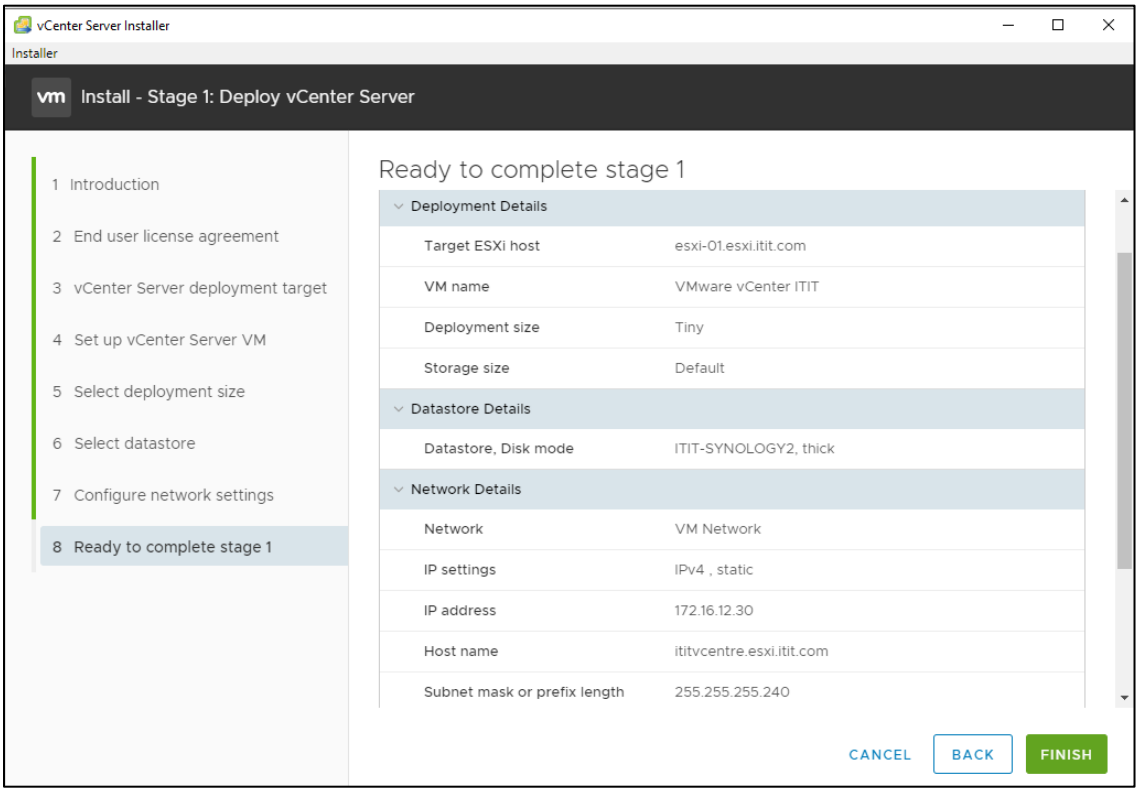


Figure 8.99

11. The deployment process will begin, this can take up to 30 minutes, and take caution to not power off the target host and storage devices. Once the deployment is completed, a success message will be displayed on the screen.

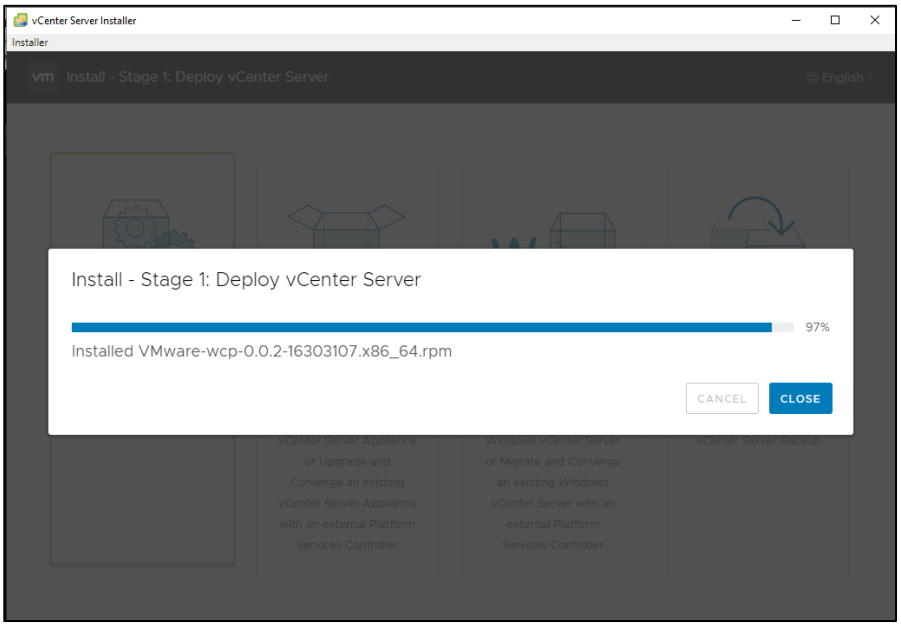


Figure 8.100

Click Continue to begin the stage 2 of the deployment process, vCenter Server Setup or the setup can be exited in which case the second stage can be setup by logging in to the vCenter Server management interface at https://itivtcentre.exsi.itit.com:5480/

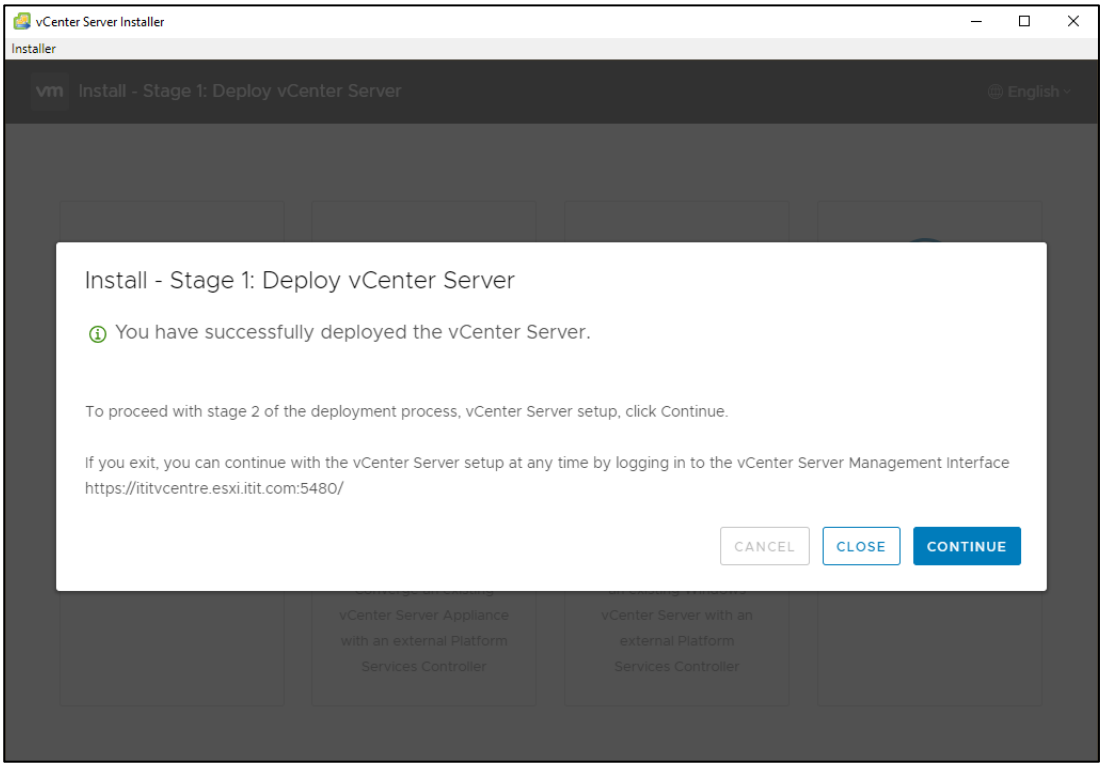


Figure 8.101

The second stage of the vCenter deployment consists of setting up the newly deployed vCenter Server Appliance with an Embedded Platform Services Controller (EPSC). This is a new service that that handles the infrastructure security functions such as vCenter Single Sign-On, licensing, certificate management and server reservation. EPSC provides one appliance for system administrators for centralized management of these common infrastructure services. When deploying the vCenter Server with an EPSC all services that

are bundled with the Platform Services Controller are deployed together with the vCenter Server services on the same virtual machine or physical server.

The second stage of the vCenter deployment is documented below. This is the continuation from the last step in the previous page.

12. Hit Next to begin the second stage of the vCenter deployment.

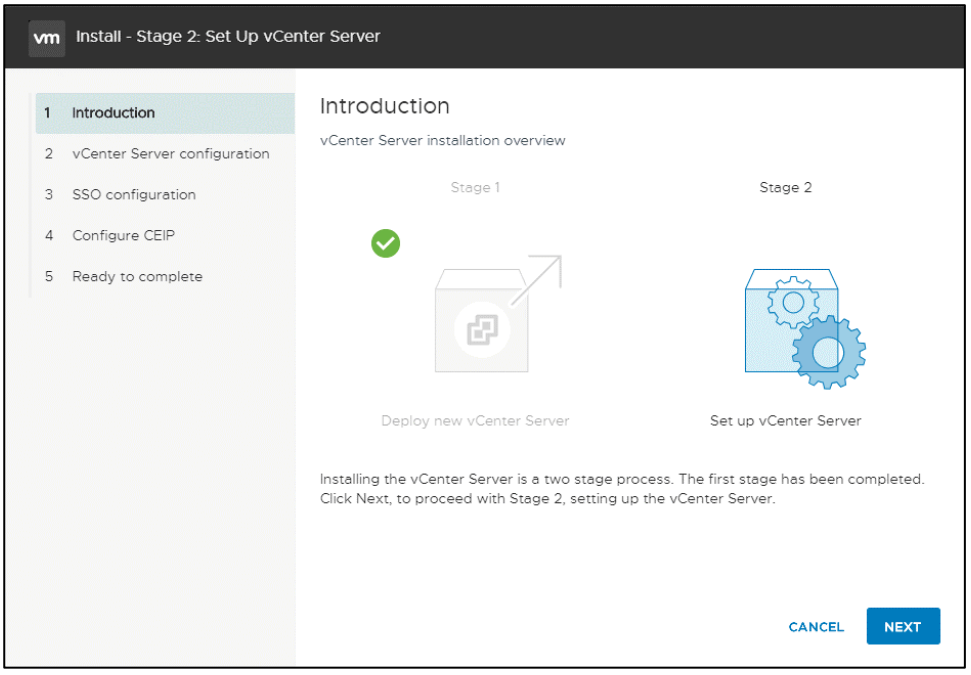


Figure 8.102

13. Select the time synchronization setting to synchronize the clock with the ESXi host and enable SSH access, this is required to enable vCenter Server High Availability.

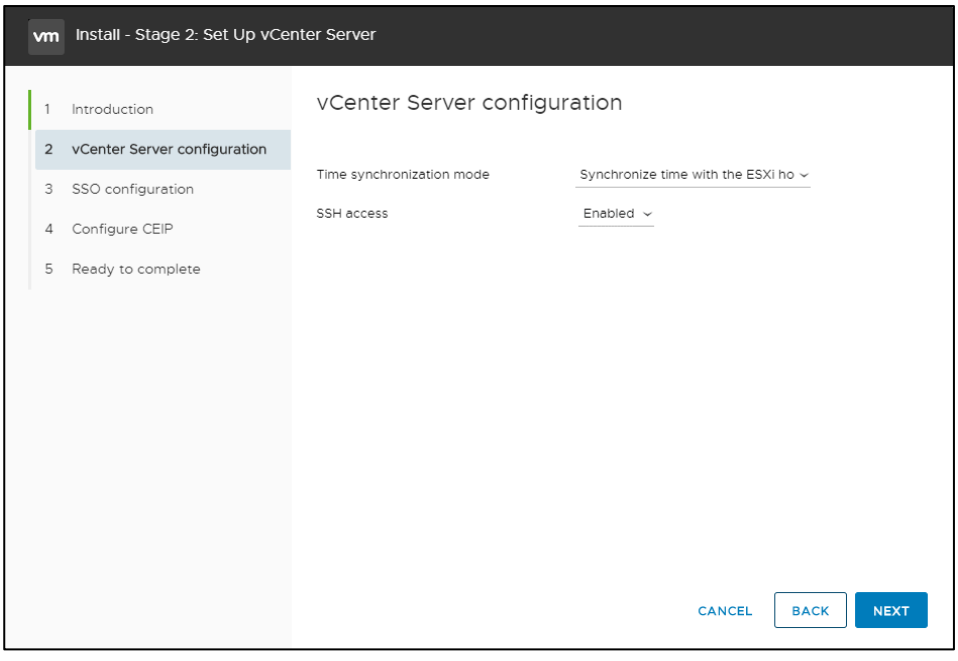


Figure 8.103

14. Create a new vCenter single sign-on domain by providing a domain name, password for the vCenter single sign-on administrator account.

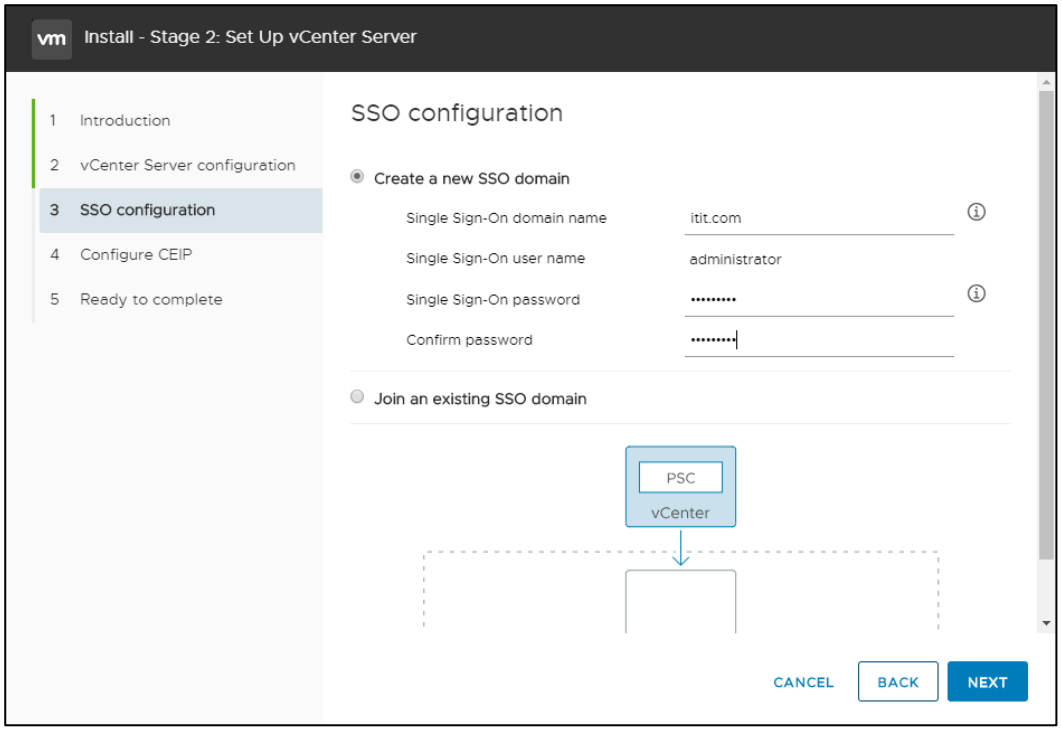


Figure 8.104

15. Deselect the checkbox to prevent sending statistics information to VMWare and to not take part in the VMWare Customer Experience Improvement Program.

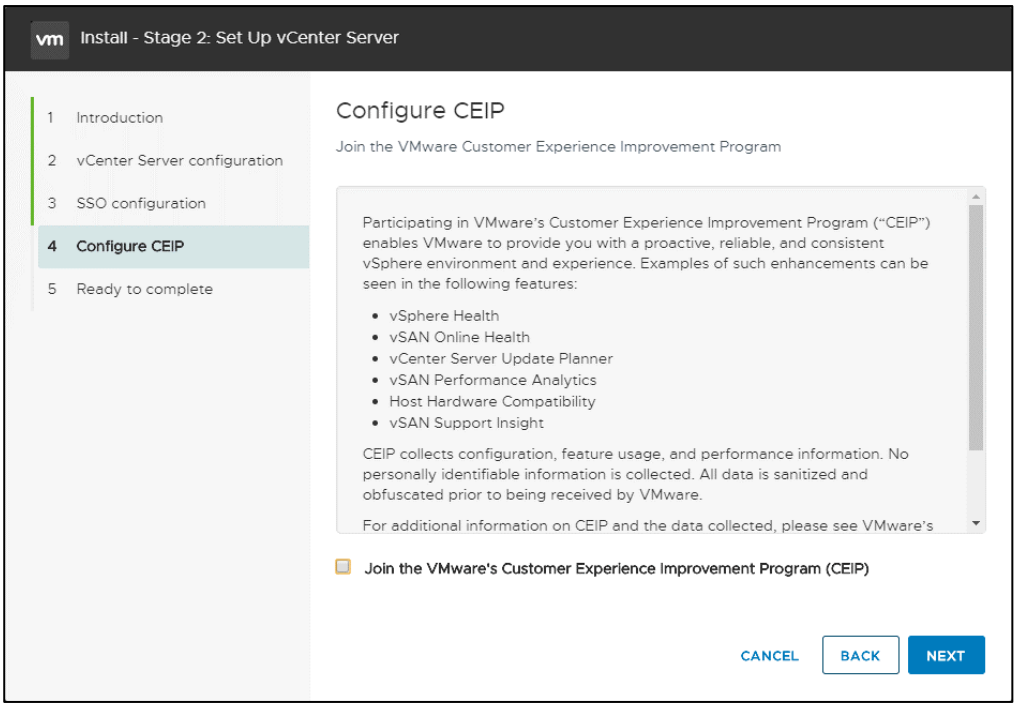


Figure 8.105

16. Review the settings and click Finish to begin the vCenter Server setup process.

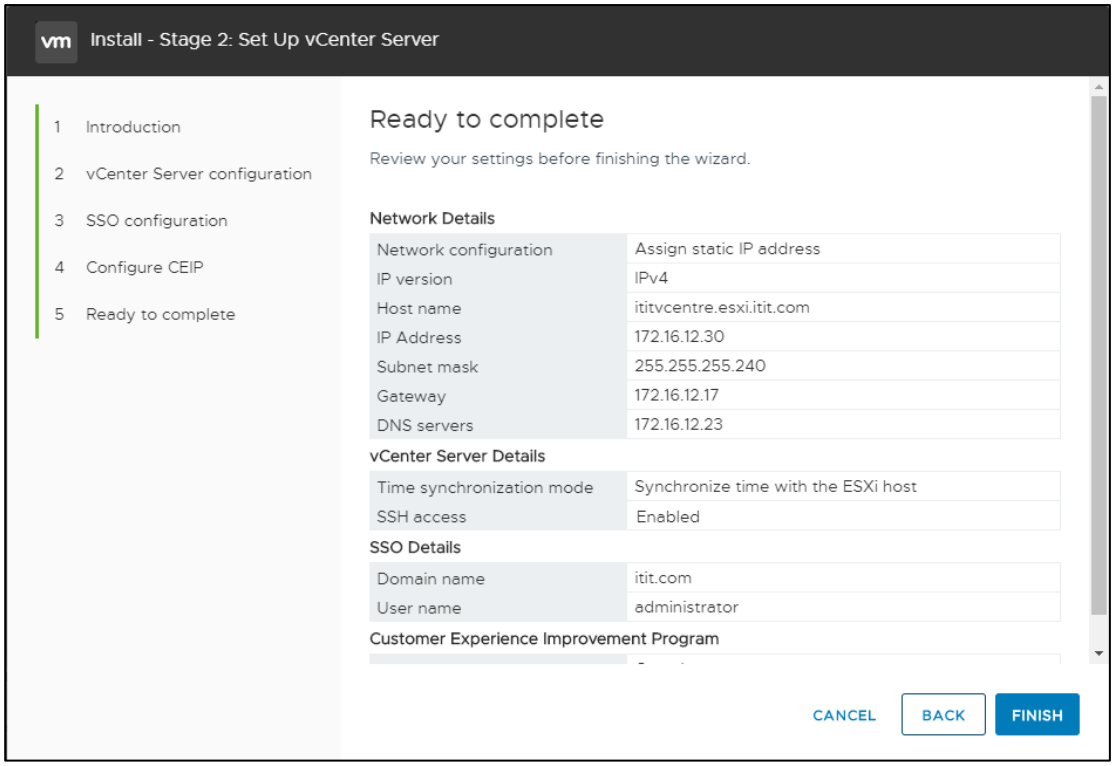


Figure 8.106

Do not terminate the installation process. Termination will require the user to reinstall vCenter Server from the beginning. The installation process will take up to 20 minutes.

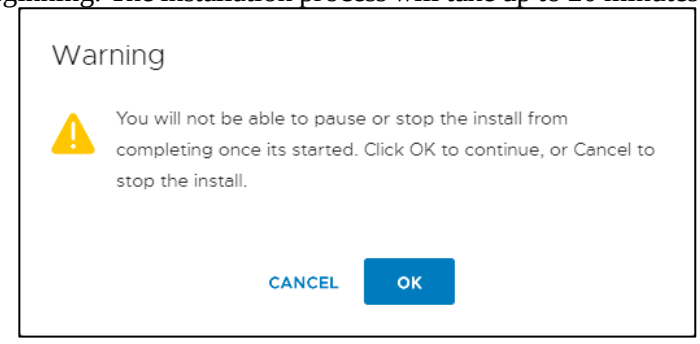


Figure 8.107

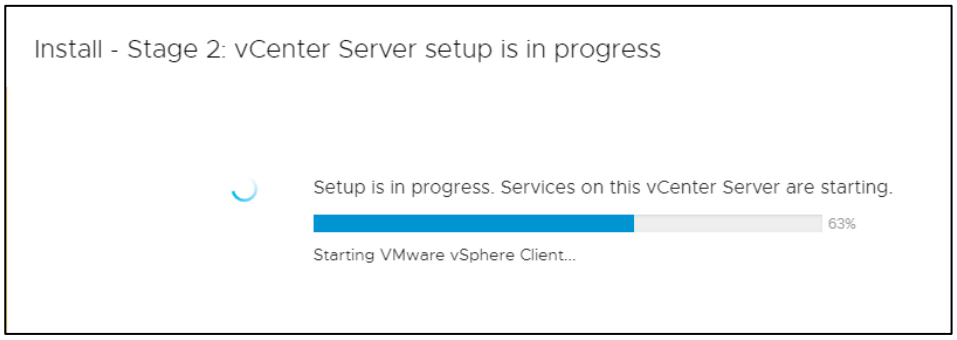


Figure 8.108

Once the installation is completed, visit the vCenter Server web interface at https://ititvcenter.esxi.itit.com:443/

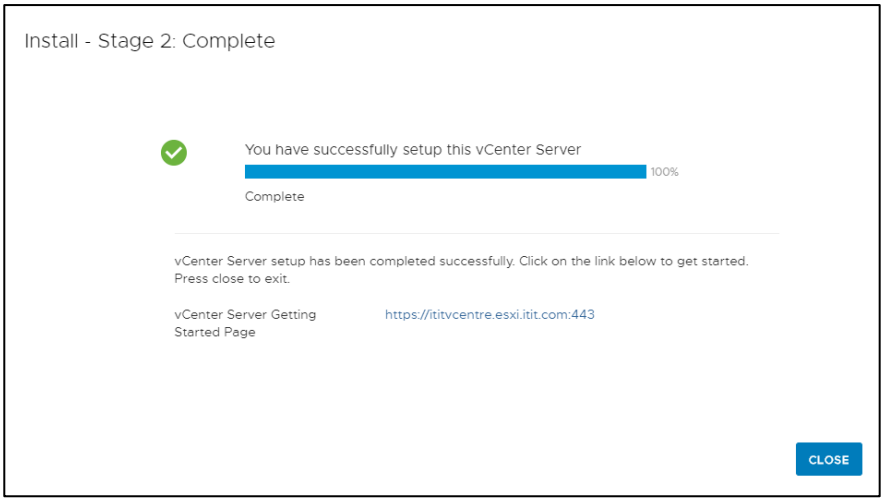


Figure 8.109

Figure 8.110

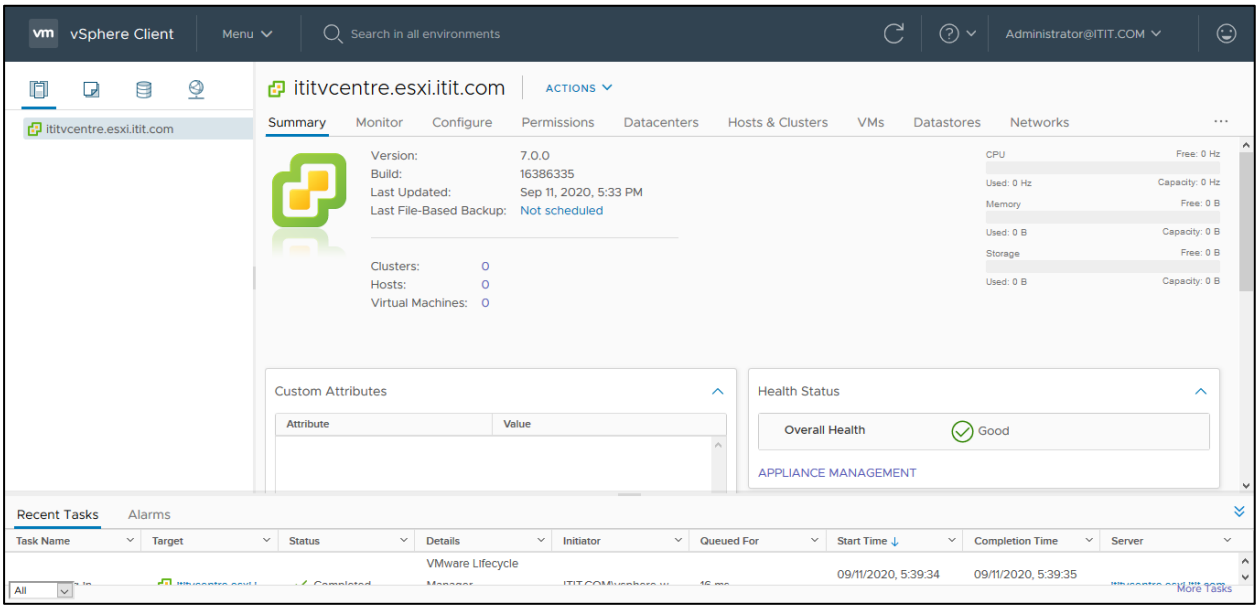


Figure 8.111

Once vCenter Server has been deployed and configured the vSphere environment can be configured, virtual machines can be created and various vSphere services can be set up.

Configuring vSphere Client for Centralized Server Management

The vSphere Web Client is a web application installed on a machine with network access to your vCenter Server installation. It can be used as an interface to manage and administer VMWare ESXi hosts and the vCenter Server. Below is the configuration made via the vSphere Web Client to setup server management, administration, monitoring and high availability.

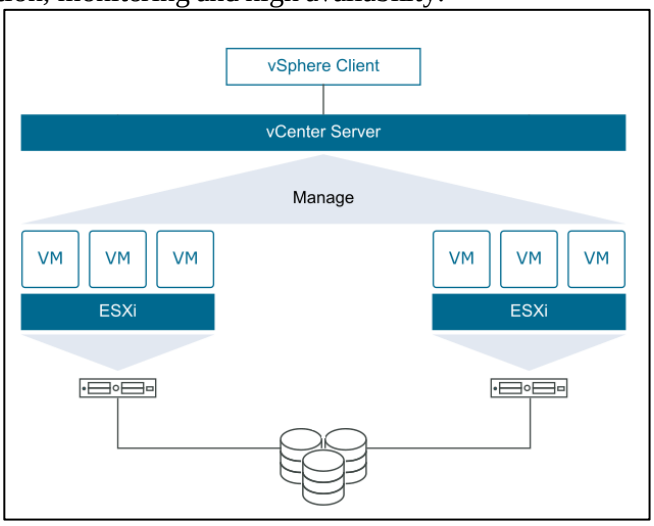


Figure 8.112

01. For the organization of the ESXi hosts a Datacenter needs to be created first. A Datacenter is a logical unit that can be used to aggregate and organize various objects that are to work in a virtual infrastructure. To create a Datacenter click on Actions and select the New Datacenter option.

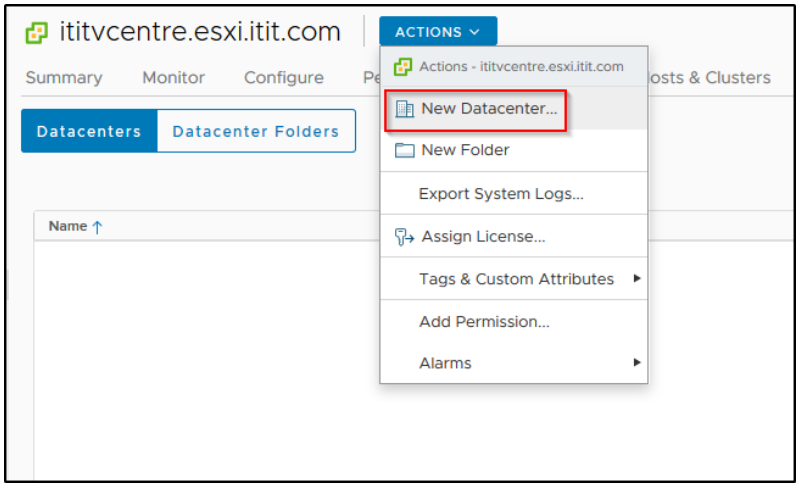


Figure 8.113

02. Enter a name for the datacenter (ITIT-DC) and click OK. The newly created datacenter should now be visible at the management page.

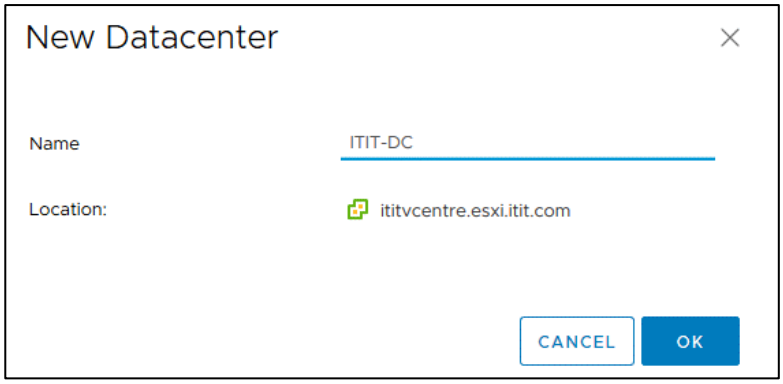


Figure 8.114

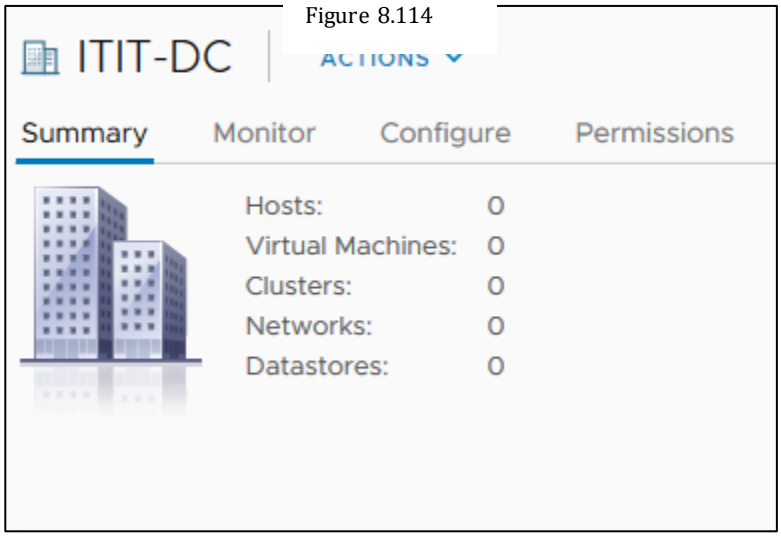


Figure 8.115

Before adding the ESXi hosts into the Datacenter, a cluster should be created. A cluster is a collection of ESXi hosts and associated virtual machines intended to work together as a unit. When you add a host to a cluster, the host's resources become part of the cluster's resources. vCenter Server manages the resources of all hosts in a cluster as one unit. The cluster manages the resources of all hosts within it. Clusters enable powerful features such as vSphere High Availability (HA), vSphere Distributed Resource Scheduler (DRS), and the VMware vSAN features.
The next sections of the report provide details on configuring the rest of the vSphere 7 environment, which includes creating a cluster for hosts, adding ESXi hosts in to the cluster, creating and managing virtual machines, and setting up high availability and fault tolerance into the cluster.

03. Select the Datacenter, expand the Actions menu, and click on New Cluster. Provide a name and click OK to create the cluster.

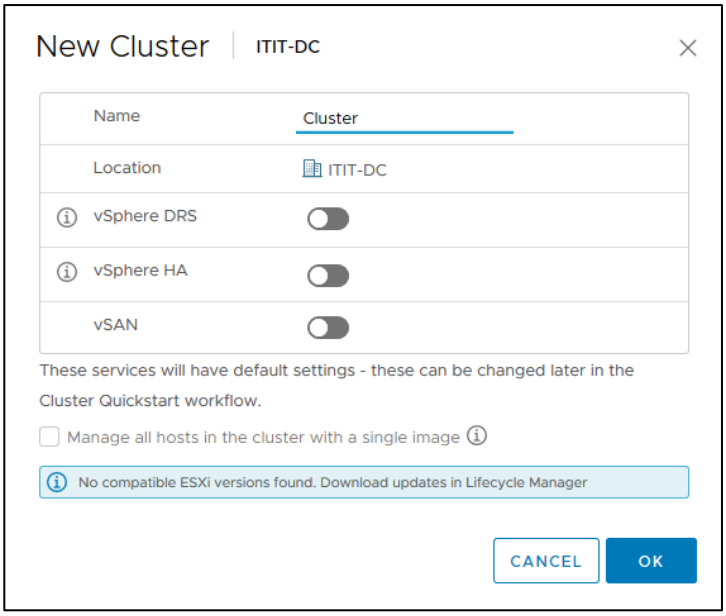


Figure 8.116

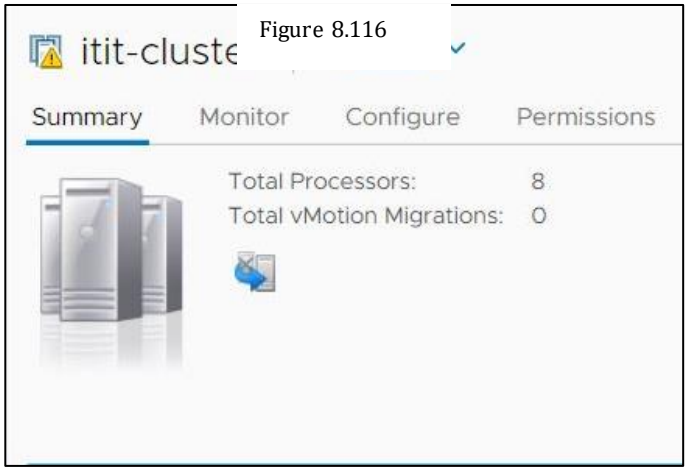


Figure 8.117

04. To add the ESXi hosts into the cluster, click on the cluster name from the menu, expand the Actions menu, and click Add Host.

05. Enter the IP addresses or the fully qualified domain names along with the administrator account details.

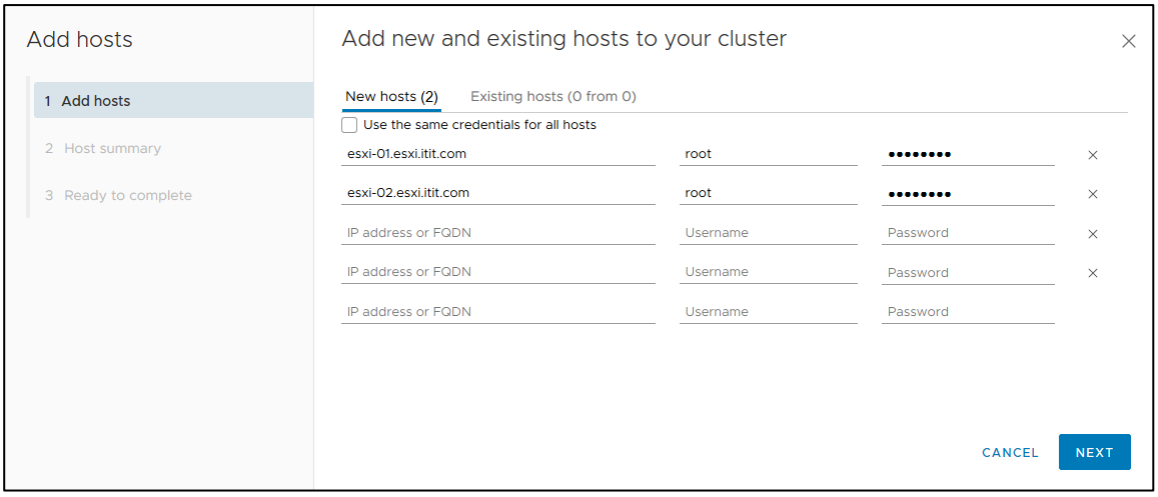


Figure 8.118

06. vCenter Server will be unable to verify the certificates provided by the ESXi hosts and display the below alert, this a common warning with self-signed certificates present on the hosts by default. Click OK to accept the certificates and add the ESXi hosts into the cluster. Upon completion the hosts would be visible under the cluster.

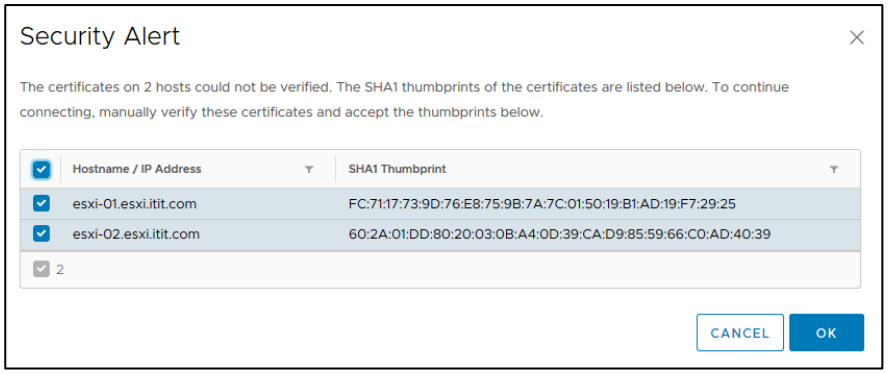


Figure 8.119

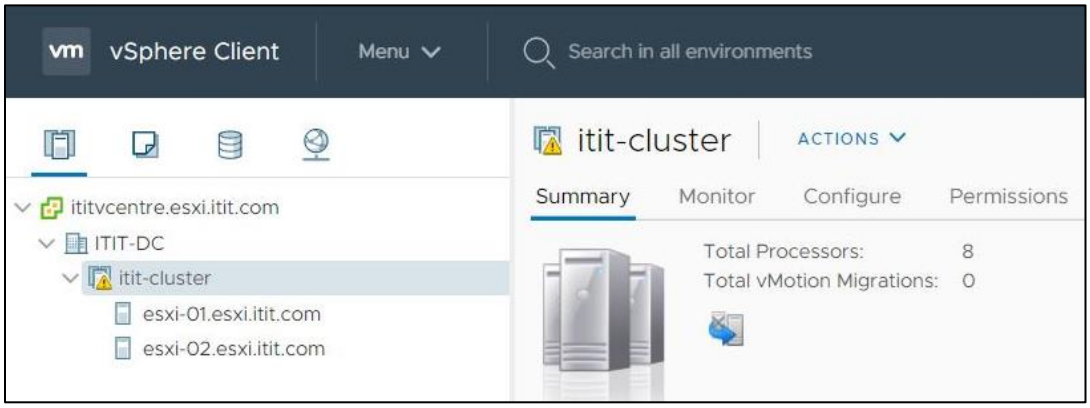


Figure 8.120

Configuring a vSphere High Availability Cluster

vSphere High Availability (HA) aggregates multiple ESXi hosts configured as a cluster to provide rapid recovery from outages and cost-effective high availability for applications running in virtual machines. vSphere HA provides the following services to ensures application availability in the following ways,

- In case of a server failure, the virtual machines on it are restarted on another host within the HA cluster.
- Continuously monitors virtual machines and resets it in the event of failure detection.
- Protects against datastore accessibility failures by restarting affected virtual machines on other hosts which still have access to their datastores.

While vSphere HA provides base level protection by restarting virtual machines in the event of a host failure, vSphere Fault Tolerance provides an additional level of availability by allowing users to protect any virtual machine from host failure with no loss of data, transactions, or connections.

01. Browse the cluster and click the configure tab. From the side menu, select vSphere Availability and click the Edit button.

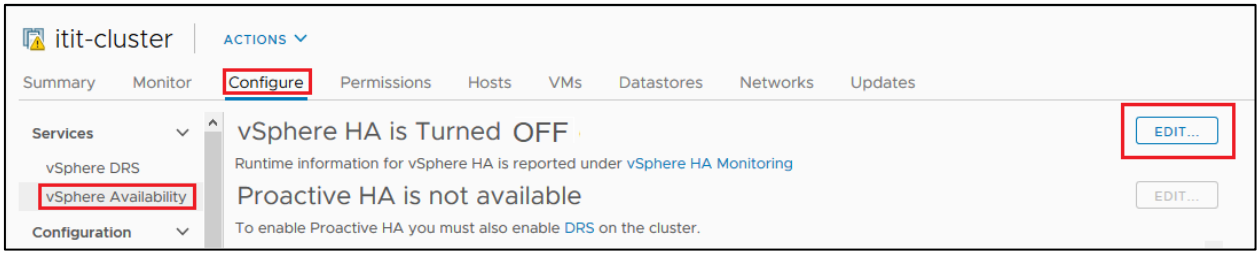


Figure 8.121

02. Select vSphere HA and under Failures and responses select Enable Host Monitoring.

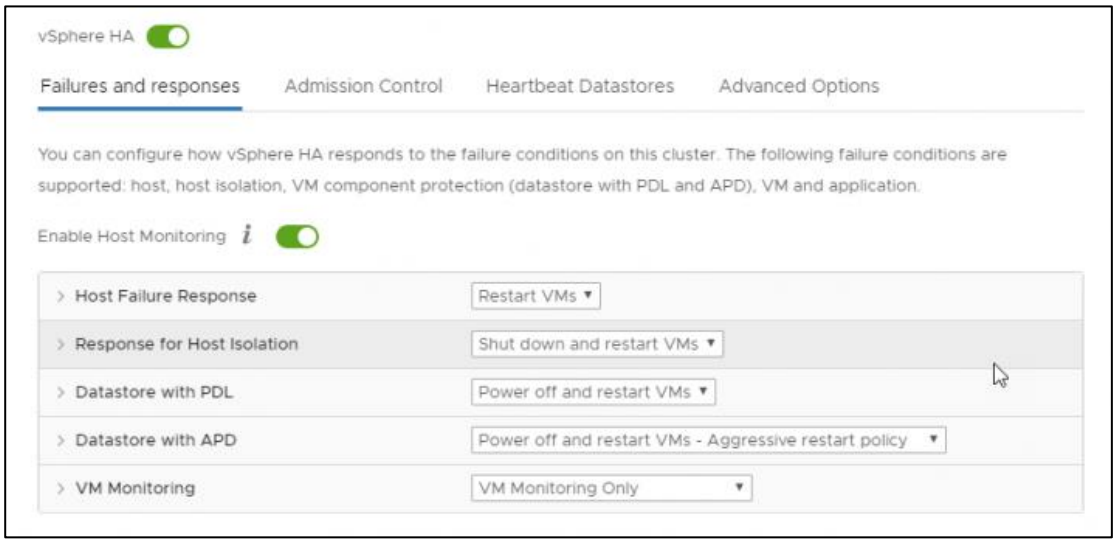


Figure 8.122

03. Select Heartbeat Datastores and instruct vSphere HA about how to select the datastores. In this scenario, vSphere has been instructed to use the datastores from a specified list. Select datastores that need to be used for heartbeating and click OK.

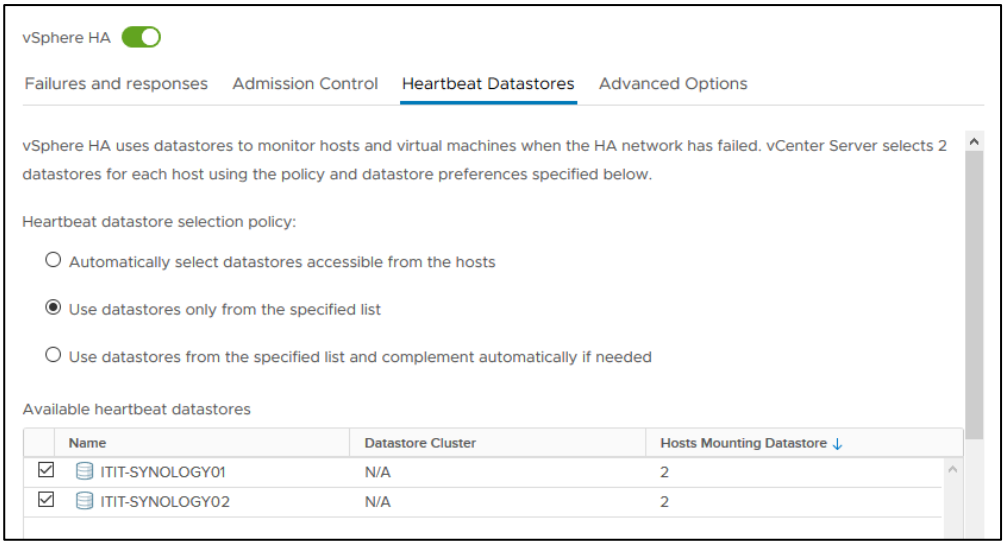


Figure 8.123

The cluster should now display that the vSphere High Availability feature is ON.

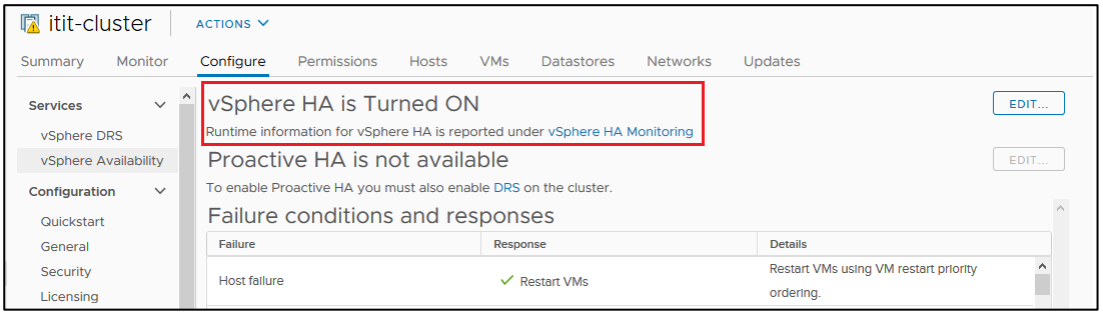


Figure 8.124

The cluster has now been configured with vSphere HA. In order to enable vSphere Fault Tolerance (FT), virtual machines must be created first, since the function of vSphere FT is to provide continuity and data protection for virtual machines. This is done by identical virtual machines run on separate hosts, one virtual machine remains active while the other is continuously available to replace it in the event of a failover situation.

Creating a Virtual Machine with vSphere Web Client

The deployment, administering, and management of virtual machines can be done using the vCenter Server via the vSphere Web client. Below is the methodology used to deploy a virtual machine using the vSphere Web client, the machine being deployed here is a Windows Server 2016 Datacenter machine which is supposed to serve the role as the Active Directory Domain Controller of the campus network.

01. Browse to the cluster and expand the Actions menu. Select New Virtual Machine.

02. On the Select a creation type page, select “Create a new virtual machine” and click Next.

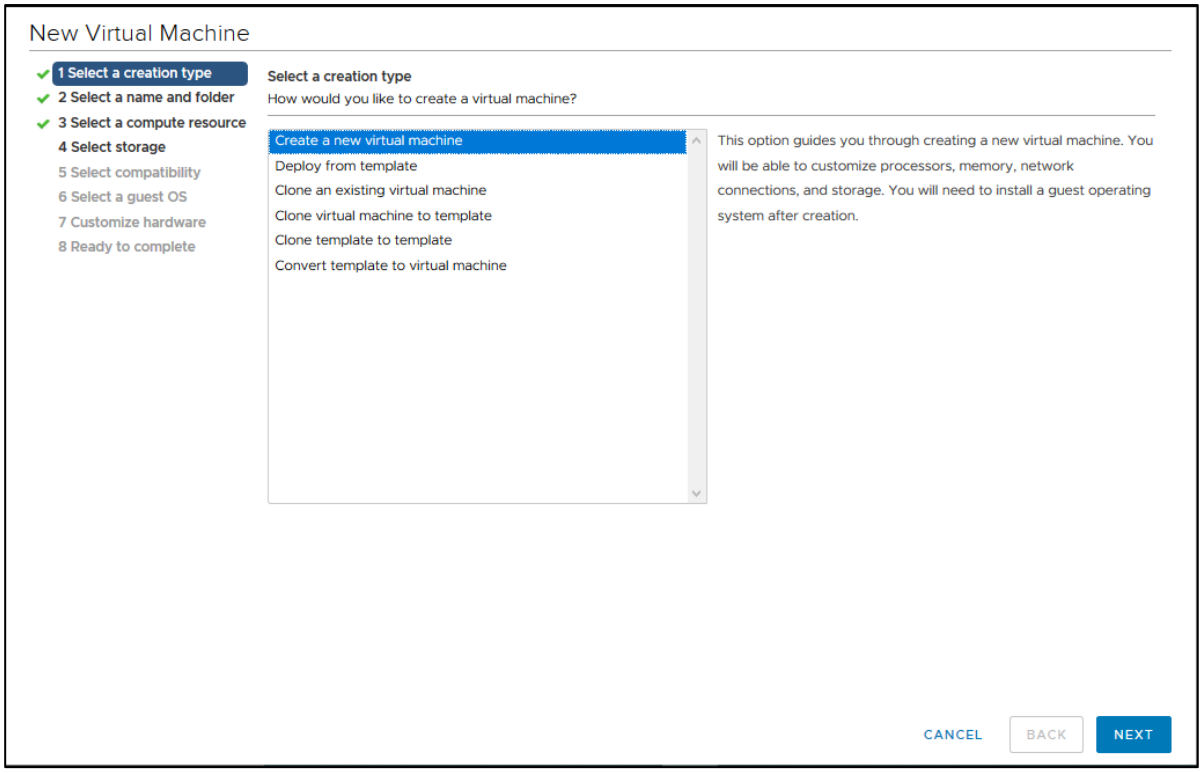


Figure 8.125

03. Enter a unique name for the virtual machine and select a deployment location.

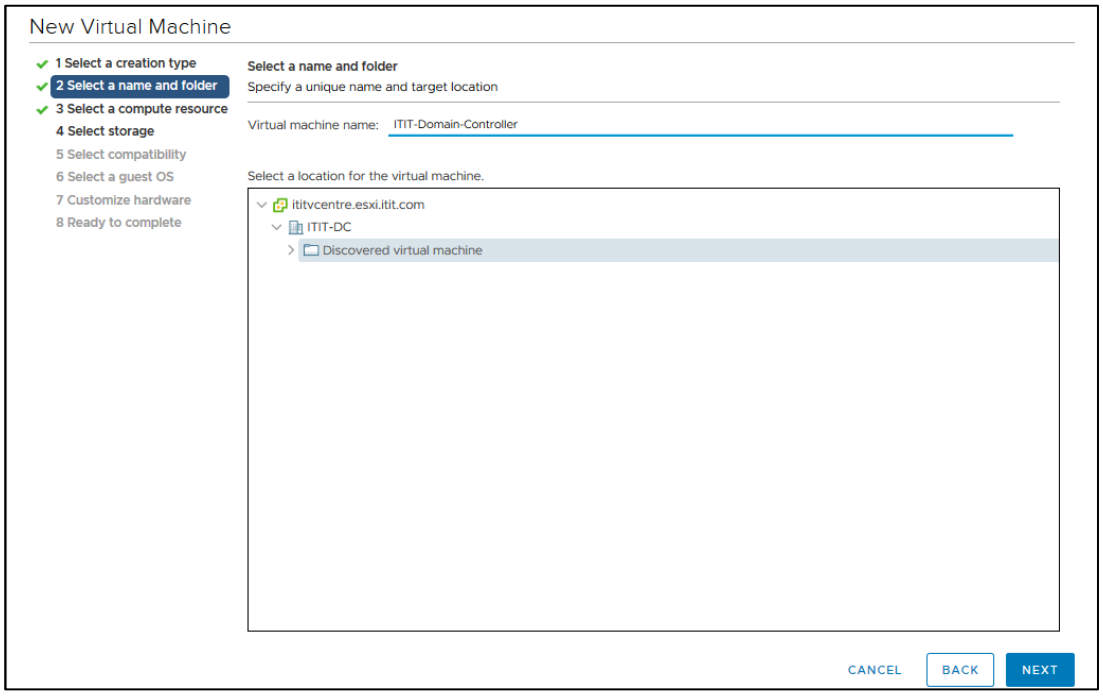


Figure 8.126

04. On the Select a compute resource page, select the ESXi host the virtual machine will run on.

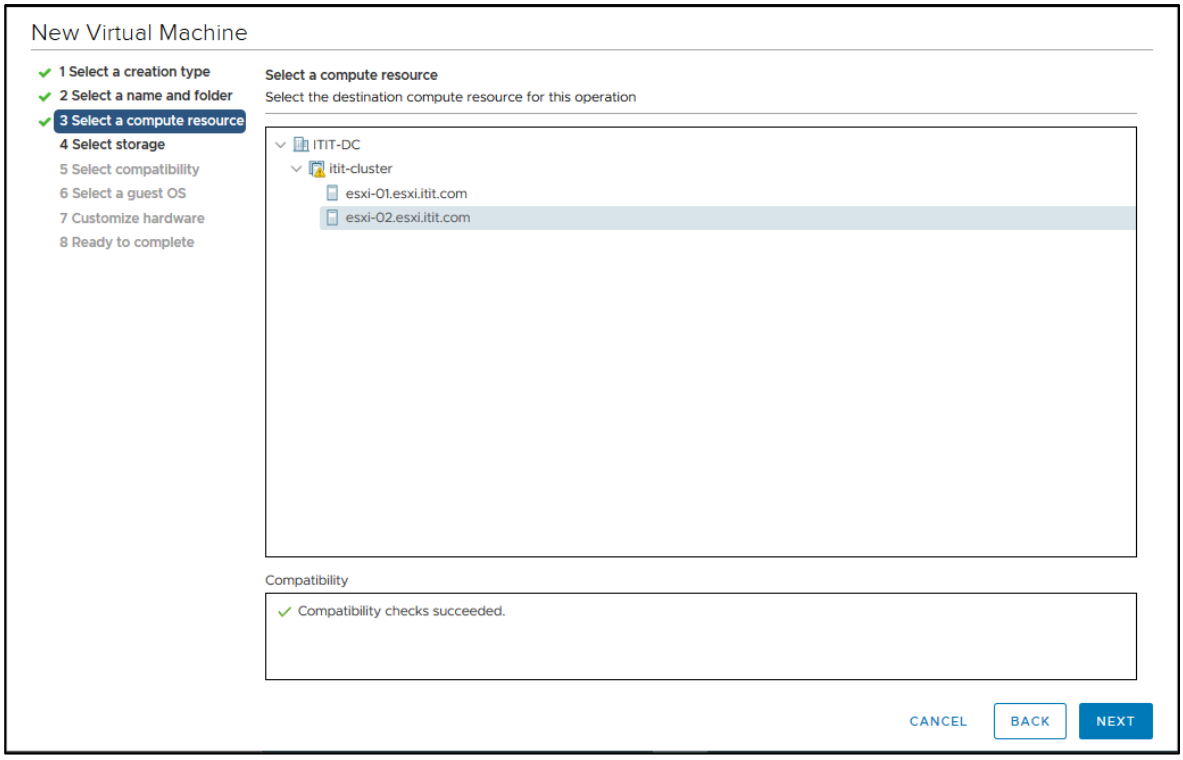


Figure 8.127

05. Select a datastore to provide storage for the virtual machine. (Check the compatibility pane below to verify if the compatibility checks have succeeded)

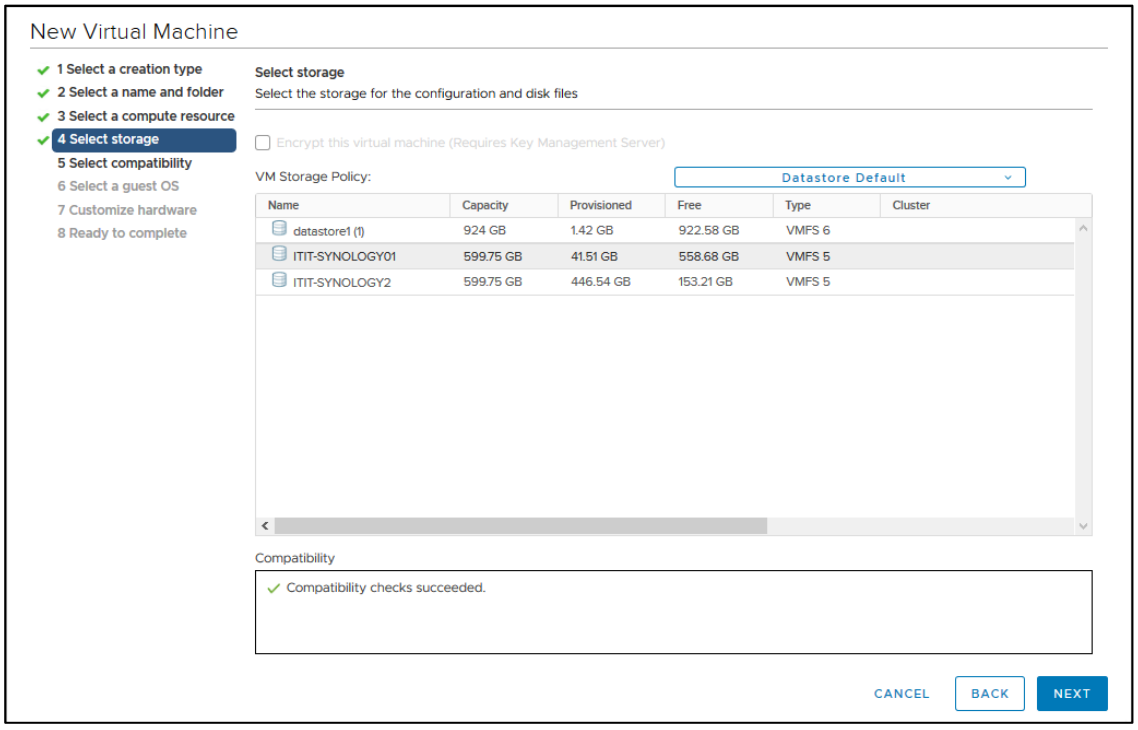


Figure 8.128

06. Select the virtual machine compatibility with the available ESXi host versions and click Next.

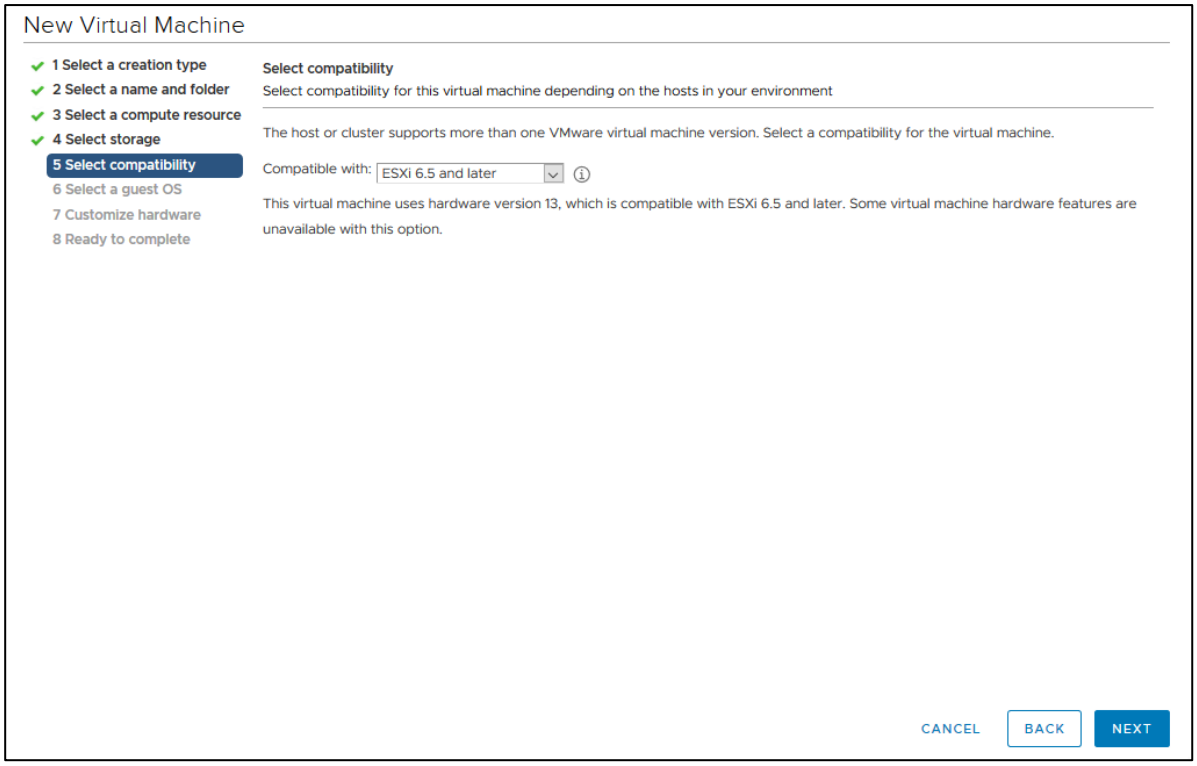


Figure 8.129

07. Select the operating system to be installed in the virtual guest. In this configuration Microsoft Windows Server 2016 (64 bit) edition was selected. Click Next to continue

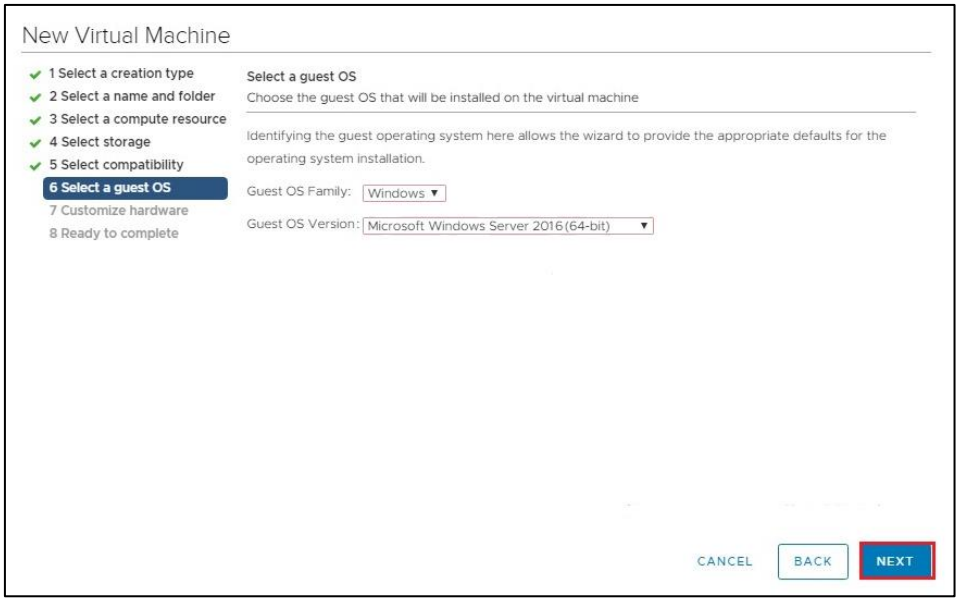


Figure 8.130

08. Configure the virtual machine hardware as required and click Next.

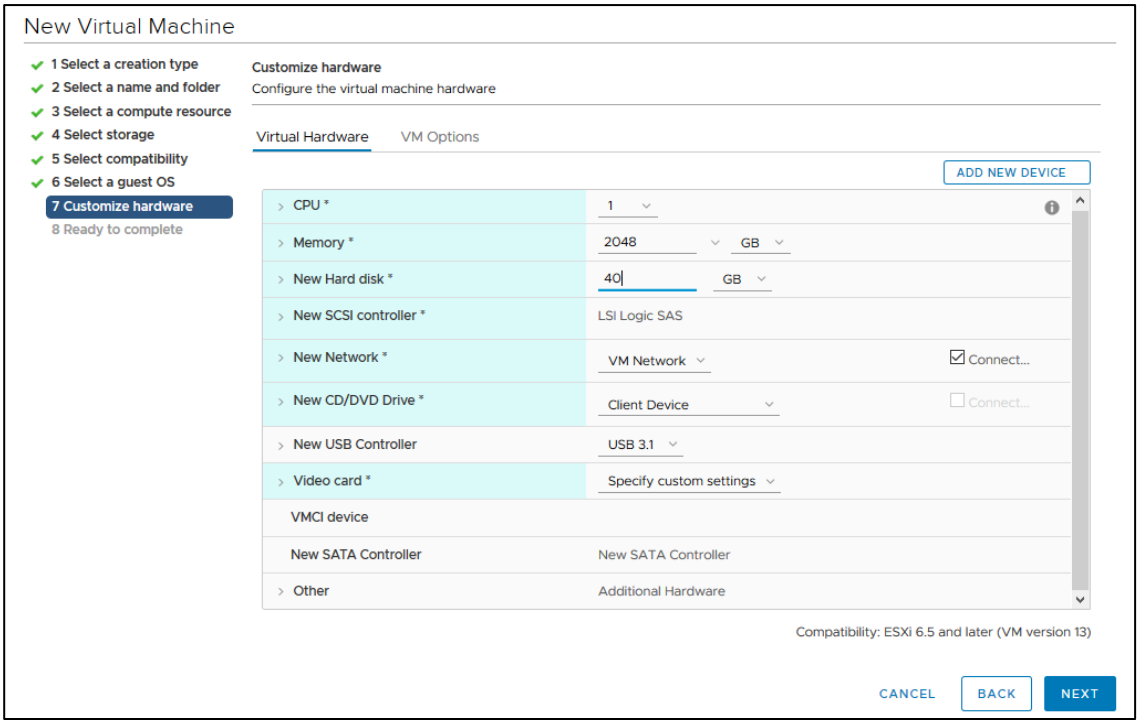


Figure 8.131

09. Review the configurations and click Finish to complete the creation of the virtual machine.

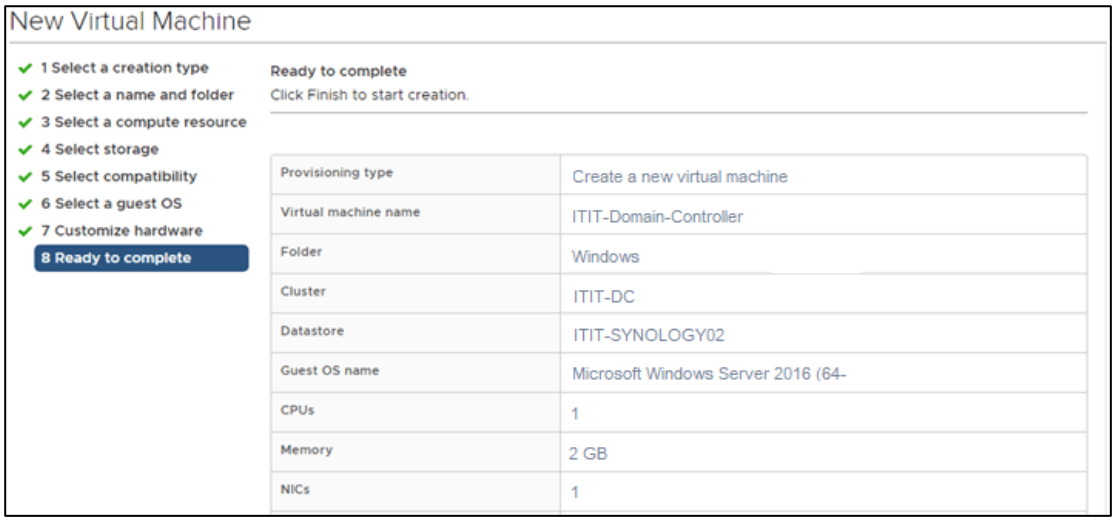


Figure 8.131

10. Once the virtual machine has been created, launch the remote console to begin the installation of the guest operating system. Before continuing ensure that the guest operating system image is loaded and ready to be used for installation.

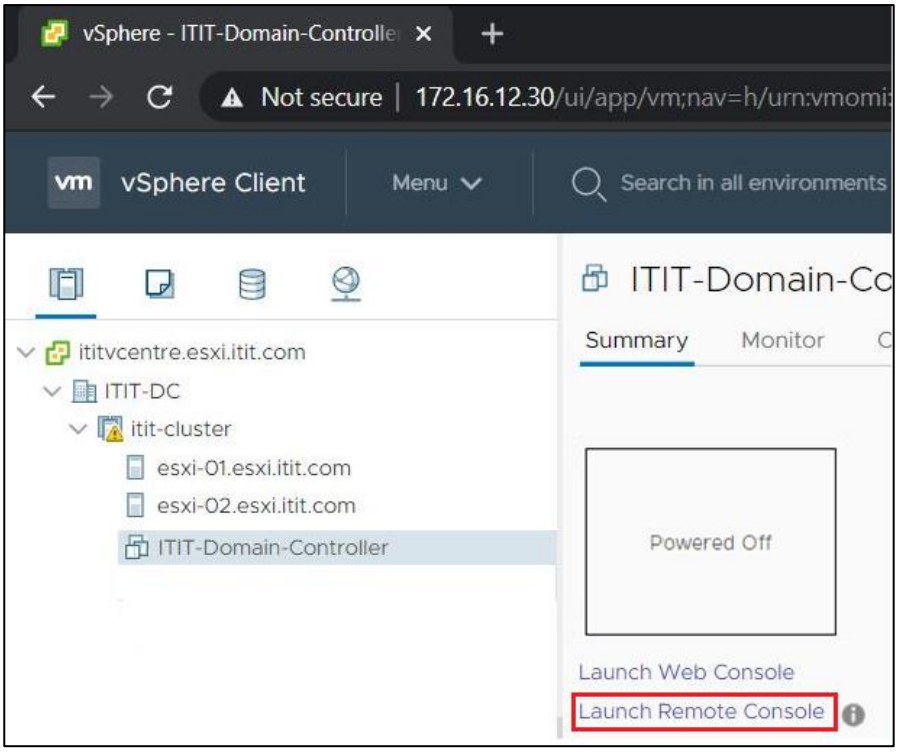


Figure 8.132

11. Once the remote console application starts, click on the VMRC button, select the Removable Device option from the drop down menu, and then the CD/DVD drive, and then select the Connect to Disk Image File (iso) option.

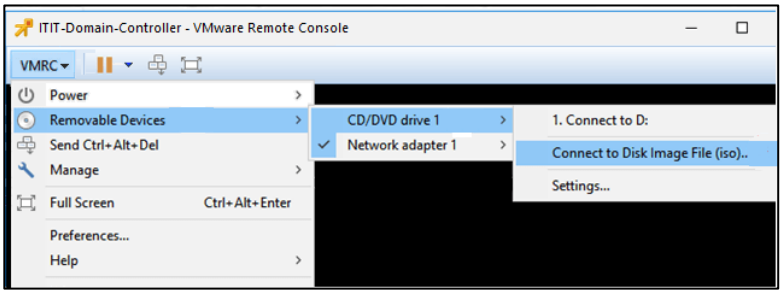


Figure 8.133

12. From the VMRC drop down menu select Send Ctrl+Alt+Del to send the key combination into the virtual machine. The Windows installer will now begin.

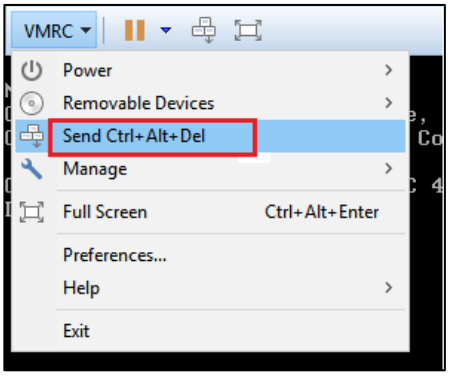


Figure 8.134

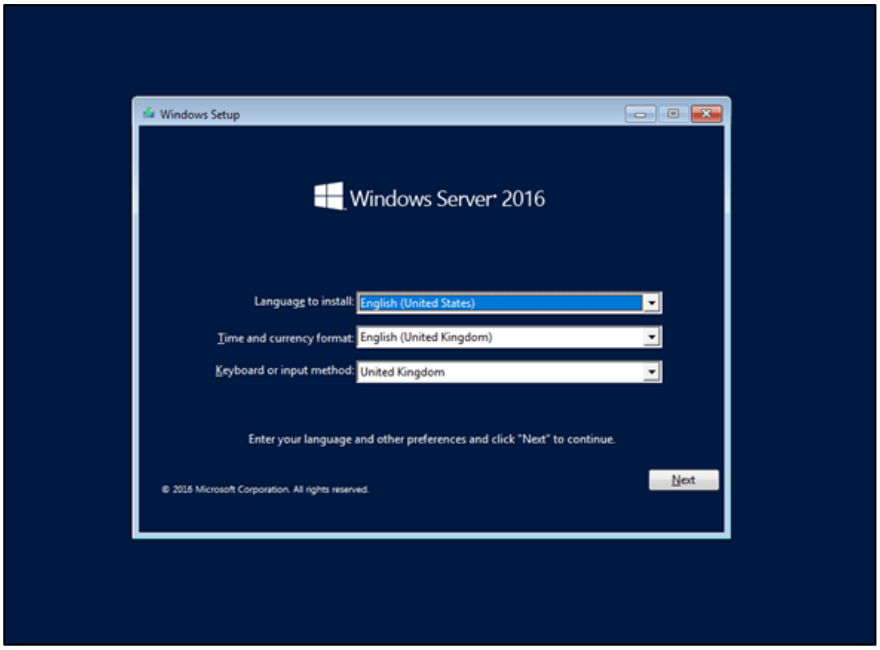


Figure 8.135

Providing Fault Tolerance for Virtual Machines

Before using vSphere Fault tolerance a few prerequisites must be fulfilled.

- A vSphere HA cluster is created and enabled
- Fault Tolerant logging and VMotion networking must be enabled.

01. To enable fault tolerance and VMotion in the ESXi host's NIC, browse to the ESXi host, click Configure tab and click Networking and click on VMkernel adapters. Select VMkernel the interface to enable Fault Tolerance logging and vMotion services, and click Edit.

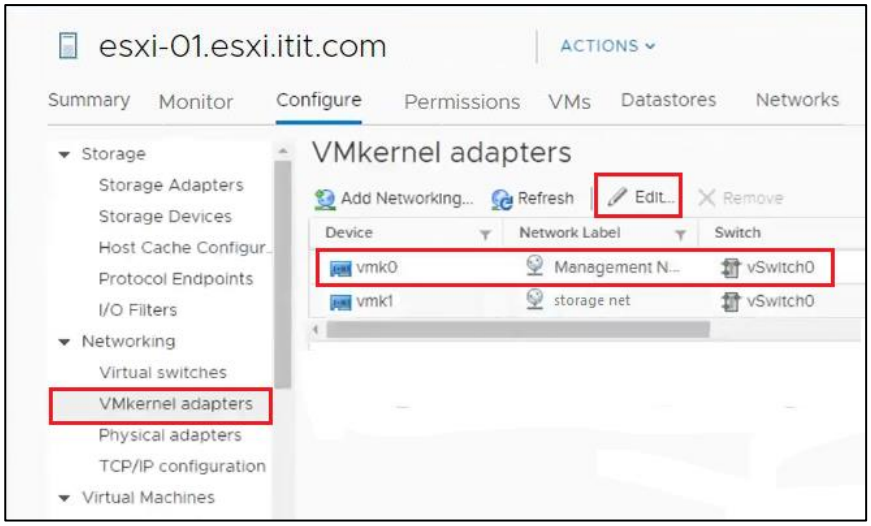


Figure 8.136

02. Enable vMotion and Fault Tolerance logging. Click OK when done.

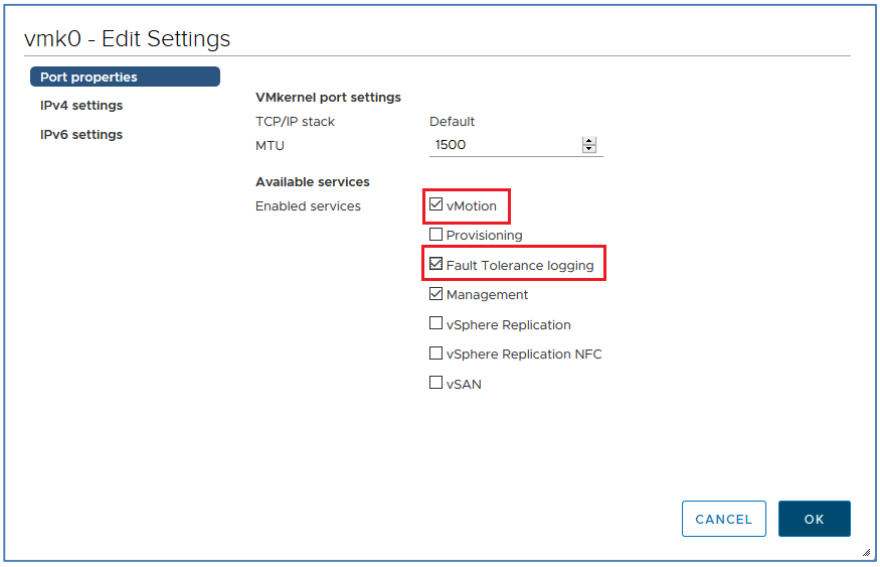


Figure 8.137

03. Right click on the virtual machine, select Fault Tolerance, and then select Turn On Fault Tolerance.

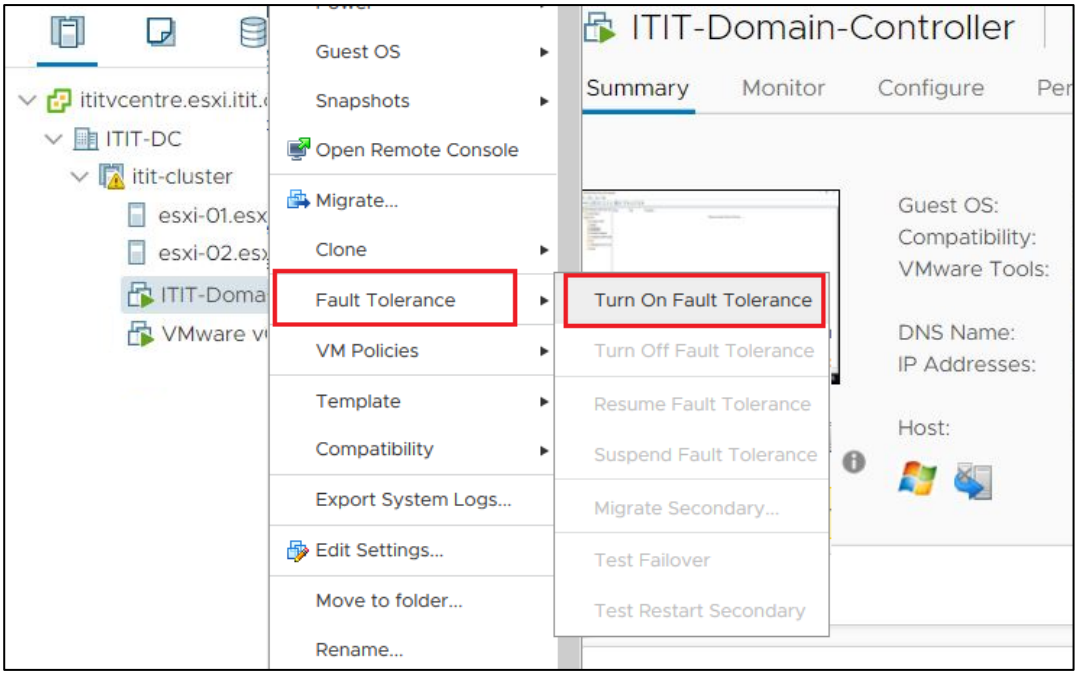


Figure 8.138

04. vSphere Fault Tolerance works by creating a secondary or shadow virtual machine on a secondary ESXi server. In the below step select a datastore to provide storage for the secondary virtual machine. Per best practices, choose a datastore that is different from the one that holds the primary virtual machine.

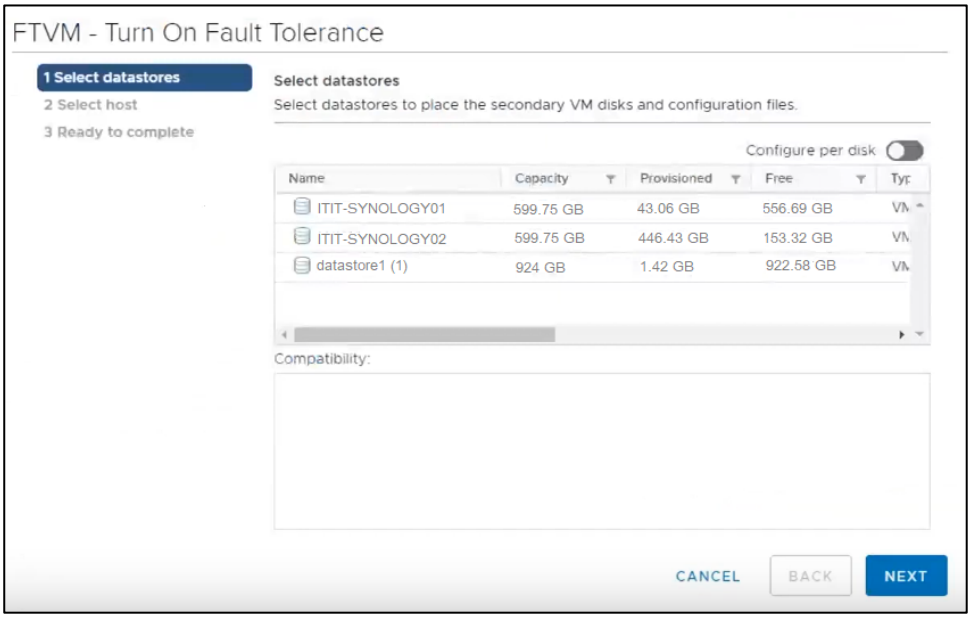


Figure 8.139

05. Select an ESXi host to provide a host for the secondary virtual machine.

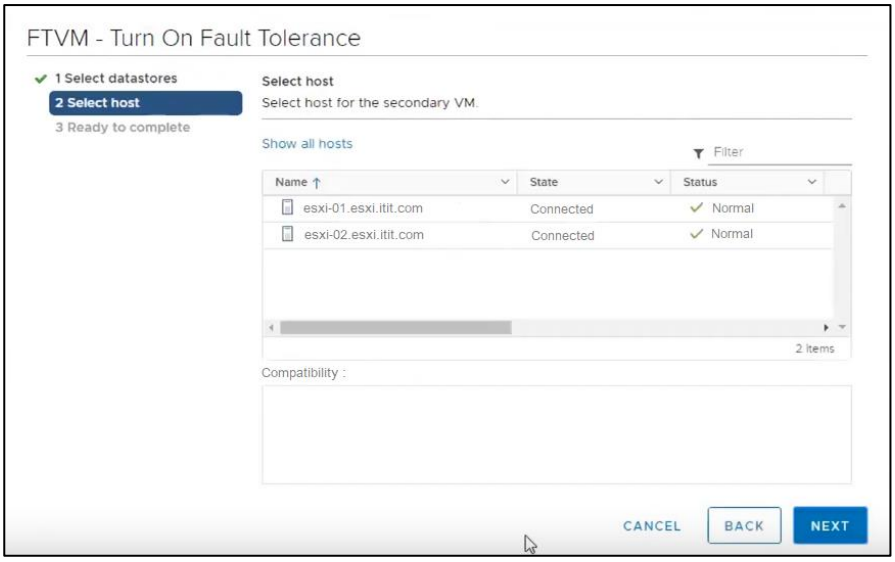


Figure 8.140

06. Review the options selected and click Finish to turn on Fault Tolerance for the virtual machine.

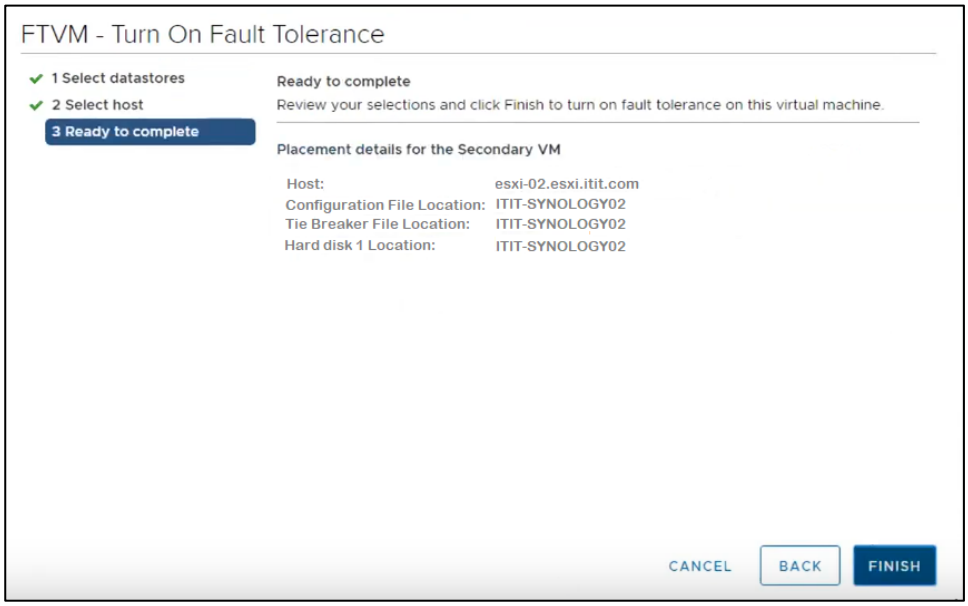


Figure 8.141

All the virtual servers that will be running on the ESXi hosts to provide services for the campus network will be configured as such. In summary, these virtual servers will be stored in the network storage cluster, run on the highly available ESXi hosts with fault tolerant capabilities. The next section of the report will provide documentation of configuring the individual services provided by these virtual servers.

Installation and Configuration of Virtual Servers

As part of the network upgrade program, the campus network required the implementation of a new internal mail server, a file server, and a web server. These server were installed and configured from scratch as virtual machines. The upgrade objectives were as follows,

- Implement an internal mail server utilizing Zimbra Collaboration 9 Network Edition for sending, receiving, and storing e-mails.
- Implement a Samba file server utilizing Samba for centralized storage and file sharing.
- Integrate the mail and file server with this existing domain
- Deploy a web server using Apache in the new demilitarized zone to host the two websites of the campus
- Deploy a reverse proxy server using Nginx to sit and provide load balancing in front of the web servers
- Provide full redundancy at the demilitarized zone.

The IP addressing of the newly created virtual servers are represented in the below table.

Server Name	Fully Qualified Domain Name	IP address
ITIT Mail Server	itit-mail-server.itit.com	172.16.12.25
ITIT File and Printer Share	itit-file-server.itit.com	172.16.12.26
ITIT Web Server (Primary)	itit-web01.itit.com	10.10.10.2
ITIT Web Server (Secondary)	itit-web02.itit.com	10.10.10.3
ITIT Reverse Proxy (Primary)	Itit-revproxy-01.itit.com	10.10.10.5
ITIT Reverse Proxy (Secondary)	itit-revproxy-02.itit.com	10.10.10.6

Table 7.0

All the virtual servers that will be configured below were created using the vSphere Web Client as shown in the previous section. The virtual machines are stored in the network storage cluster and configured to run on the highly available ESXi hosts and has vSphere Fault Tolerance enabled.

Figure 8.141

Implementing the Internal Mail Server

01. The system information of the mail server is shown below. Ensure that the mail server is set up with a valid hostname.

```
[itit-administrator@itit-mail-server ~]$ hostnamectl
   Static hostname: itit-mail-server
         Icon name: computer-vm
           Chassis: vm
        Machine ID: 8eafc99838684fd1bcd7e7766be7321f
           Boot ID: 48be98c1d6c64b5d803011652e0f472e
    Virtualization: microsoft
  Operating System: CentOS Linux 8 (Core)
       CPE OS Name: cpe:/o:centos:centos:8
            Kernel: Linux 4.18.0-193.6.3.el8_2.x86_64
      Architecture: x86-64
[itit-administrator@itit-mail-server ~]$ sudo yum install unzip net-tools syssta
t openssh-clients perl-core libaio nmap-ncat libstdc++.so.6 wget -y
CentOS-8 - AppStream                            29 MB/s | 5.6 MB     00:00
CentOS-8 - Base                                 22 MB/s | 2.2 MB     00:00
CentOS-8 - Extras                               20 kB/s | 7.9 kB     00:00
CentOS-8 - openlogic packages for x86_64        53 kB/s | 3.0 kB     00:00
```

Figure 8.142

02. First ensure that an A record and an MX record pointing to the mail server IP address is present. In this implementation the DNS server is set as the internal DNS server residing at 172.16.12.21.

```
[itit-administrator@itit-mail-server ~]$ dig -t A itit-mail-server.itit.com

; <<>> DiG 9.11.13-RedHat-9.11.13-5.el8_2 <<>> -t A itit-mail-server.itit.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 44482
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
;itit-mail-server.itit.com.     IN      A

;; ANSWER SECTION:
itit-mail-server.itit.com. 3600 IN      A       172.16.12.25

;; Query time: 2 msec
;; SERVER: 172.16.12.21#53(172.16.12.21)
;; WHEN: Mon Sep 14 21:39:05 UTC 2020
;; MSG SIZE  rcvd: 70
```

Figure 8.143

```
[itit-administrator@itit-mail-server ~]$ dig -t MX itit-mail-server.itit.com

; <<>> DiG 9.11.13-RedHat-9.11.13-5.el8_2 <<>> -t MX itit-mail-server.itit.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 33989
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
;itit-mail-server.itit.com.     IN      MX

;; ANSWER SECTION:
itit-mail-server.itit.com. 3600 IN      MX      10 itit-mail-server.itit.com.

;; ADDITIONAL SECTION:
itit-mail-server.itit.com. 3600 IN      A       172.16.12.25

;; Query time: 2 msec
;; SERVER: 172.16.12.21#53(172.16.12.21)
;; WHEN: Mon Sep 14 21:39:47 UTC 2020
;; MSG SIZE  rcvd: 86
```

Figure 8.144

The above outputs confirm that the DNS records are correctly configured for the mail server.

03. Download the Zimbra Collaborstion 9 (latest) package into the opt folder for installation using the link *https://files.zimbra.com/downloads/9.0.0_GA/zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300.tgz*

```
[itit-administrator@itit-mail-server ~]$ sudo mkdir -p /opt/zimbra && cd /opt/zi
mbra
[itit-administrator@itit-mail-server zimbra]$ sudo wget https://files.zimbra.com
/downloads/9.0.0_GA/zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300.tgz
--2020-09-14 12:04:37--  https://files.zimbra.com/downloads/9.0.0_GA/zcs-NETWORK
-9.0.0_GA_3954.RHEL8_64.20200629045300.tgz
Resolving files.zimbra.com (files.zimbra.com)... 99.86.226.218
Connecting to files.zimbra.com (files.zimbra.com)|99.86.226.218|:443... connecte
d.
HTTP request sent, awaiting response... 200 OK
Length: 537151560 (512M) [binary/octet-stream]
Saving to: ‘zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300.tgz’

zcs-NETWORK-9.0.0_G 100%[===================>] 512.27M   102MB/s    in 5.3s

2020-09-14 12:04:43 (97.1 MB/s) - ‘zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.2020062904
5300.tgz’ saved [537151560/537151560]

[itit-administrator@itit-mail-server zimbra]$
```

Figure 8.145

04. Extract the downloaded archive using the tar command (will require super user privileges)

```
[itit-administrator@itit-mail-server zimbra]$ sudo tar zxvpf zcs-NETWORK-9.0.0_G
A_3954.RHEL8_64.20200629045300.tgz
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/bin/
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/bin/checkLicense.pl
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/bin/checkService.pl
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/bin/get_plat_tag.sh
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/bin/zmValidateLdap.pl
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/bin/zmdbintegrityreport
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/bin/checkValidBackup
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/data/
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/data/versions-init.sql
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/data/backup-version-init.sql
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/docs/
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/docs/en_US/
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/docs/en_US/admin.pdf
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/docs/en_US/Fedora Server Confi
g.pdf
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/docs/en_US/Import_Wizard_Outlo
ok.pdf
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/docs/en_US/Migration_Exch_Admi
n.pdf
zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629045300/docs/en_US/MigrationWizard_Dom
ino.pdf
```

Figure 8.146

05. Enter into the extracted directory and run the “install.sh” script to install Zimbra. (superuser privileges required to run the script) The installer will search for any existing Zimbra packages first.

```
[itit-administrator@itit-mail-server zimbra]$ cd zcs-NETWORK-9.0.0_GA_3954.RHEL8
_64.20200629045300/
[itit-administrator@itit-mail-server zcs-NETWORK-9.0.0_GA_3954.RHEL8_64.20200629
045300]$ sudo ./install.sh

Operations logged to /tmp/install.log.gtFedEtW
Checking for existing installation...
    zimbra-drive...NOT FOUND
    zimbra-imapd...NOT FOUND
    zimbra-modern-ui...NOT FOUND
    zimbra-modern-zimlets...NOT FOUND
    zimbra-patch...NOT FOUND
    zimbra-mta-patch...NOT FOUND
    zimbra-proxy-patch...NOT FOUND
    zimbra-license-tools...NOT FOUND
    zimbra-license-extension...NOT FOUND
    zimbra-network-store...NOT FOUND
    zimbra-network-modules-ng...NOT FOUND
    zimbra-chat...NOT FOUND
    zimbra-connect...NOT FOUND
    zimbra-talk...NOT FOUND
    zimbra-ldap...NOT FOUND
    zimbra-logger...NOT FOUND
    zimbra-mta...NOT FOUND
```

Figure 8.147

06. Press Y to accept the software license agreement

```
SYNACOR, INC.
END USER LICENSE AGREEMENT
FOR ZIMBRA NETWORK EDITION


(c) Copyright 2020 Synacor, Inc. All rights reserved.




Do you agree with the terms of the software license agreement? [N] Y
```

Figure 8.148

07. Press Y to configure Zimbra Package Repository and install its components. Press Y to confirm the installation of each individual package.

```
Use Zimbra's package repository [Y] Y
```

Figure 8.149

```
Select the packages to install

Install zimbra-ldap [Y] Y

Install zimbra-logger [Y] Y

Install zimbra-mta [Y] Y

Install zimbra-dnscache [Y] Y

Install zimbra-snmp [Y] Y

Install zimbra-store [Y] Y

Install zimbra-apache [Y] Y

Install zimbra-spell [Y] Y

Install zimbra-convertd [Y] Y

Install zimbra-memcached [Y] Y

Install zimbra-proxy [Y] Y
```

Figure 8.150

```
Install zimbra-archiving [N] N

Install zimbra-drive [Y] Y

Install zimbra-imapd (BETA - for evaluation only) [N] N

Install zimbra-network-modules-ng [Y] Y

Install zimbra-connect [Y] Y
```

Figure 8.151

08. The packages to be installed will be displayed along with a prompt asking for permission to modify the system (download and installation), hit Y to accept.

```
Installing:
    zimbra-core
    zimbra-ldap
    zimbra-logger
    zimbra-mta
    zimbra-dnscache
    zimbra-snmp
    zimbra-store
    zimbra-apache
    zimbra-spell
    zimbra-convertd
    zimbra-memcached
    zimbra-proxy
    zimbra-drive
    zimbra-modern-ui
    zimbra-modern-zimlets
    zimbra-patch
    zimbra-mta-patch
    zimbra-proxy-patch
    zimbra-license-tools
    zimbra-license-extension
    zimbra-network-store
    zimbra-connect
    zimbra-network-modules-ng
```

Figure 8.152

```
The system will be modified.  Continue? [N] Y
```

Figure 8.153

09. The downloading of the packages will begin. The downloading of all the packages will only take a few minutes.

```
Beginning Installation - see /tmp/install.log.ylpSZNvU for details...

                        zimbra-core-components will be downloaded and installe
d.
                        zimbra-common-core-jar will be installed.
                       zimbra-common-core-libs will be installed.
                       zimbra-common-mbox-conf will be installed.
                  zimbra-common-mbox-conf-attrs will be installed.
                   zimbra-common-mbox-conf-msgs will be installed.
                 zimbra-common-mbox-conf-rights will be installed.
                          zimbra-common-mbox-db will be installed.
                        zimbra-common-mbox-docs will be installed.
                  zimbra-common-mbox-native-lib will be installed.
                           zimbra-timezone-data will be installed.
                                    zimbra-core will be installed.
                        zimbra-ldap-components will be downloaded and installe
d.
                                    zimbra-ldap will be installed.
                                  zimbra-logger will be installed.
                         zimbra-mta-components will be downloaded and installe
d.
                                     zimbra-mta will be installed.
```

Figure 8.154

10. After successful installation the below message will be displayed.

```
Running Post Installation Configuration:
Operations logged to /tmp/zmsetup.20200914-215044.log
Installing LDAP configuration database...done.
Setting defaults...     MX: itit-mail-server.itit.com (172.16.12.25)

        Interface: 127.0.0.1
        Interface: ::1
        Interface: 172.16.12.25
                172.16.12.25
                172.16.12.25
                172.16.12.25
done.
Checking for port conflicts
```

Figure 8.155

11. In the resulting numbered menu check for any alerts or warnings with the installation.

```
   1) Common Configuration:
   2) zimbra-ldap:                          Enabled
   3) zimbra-logger:                        Enabled
   4) zimbra-mta:                           Enabled
   5) zimbra-dnscache:                      Enabled
   6) zimbra-snmp:                          Enabled
   7) zimbra-store:                         Enabled
        +Create Admin User:                 yes
        +Admin user to create:              admin@itit-mail-server.itit.com
******* +Admin Password                     UNSET
        +Anti-virus quarantine user:        virus-quarantine.eq9zjbqf@itit-ma
il-server.itit.com
        +Enable automated spam training:    yes
        +Spam training user:                spam.fazarfct@itit-mail-server.it
it.com
        +Non-spam(Ham) training user:       ham.djtftfd4@itit-mail-server.iti
t.com
        +SMTP host:                         itit-mail-server.itit.com
        +Web server HTTP port:              8080
        +Web server HTTPS port:             8443
        +Web server mode:                   https
        +IMAP server port:                  7143
        +IMAP server SSL port:              7993
        +POP server port:                   7110
```

Figure 8.156

An alert is displayed to show that the password for the Zimbra administrator account is missing.

```
        +POP server SSL port:               7995
        +Use spell check server:            yes
        +Spell server URL:                  http://itit-mail-server.itit.com:
7780/aspell.php
        +Enable version update checks:      TRUE
        +Enable version update notifications: TRUE
        +Version update notification email: admin@itit-mail-server.itit.com
        +Version update source email:       admin@itit-mail-server.itit.com
        +Install mailstore (service webapp): yes
        +Install UI (zimbra,zimbraAdmin webapps): yes
******* +License filename:                  UNSET

   8) zimbra-spell:                         Enabled
   9) zimbra-convertd:                      Enabled
  10) zimbra-proxy:                         Enabled
  11) Default Class of Service Configuration:
  12) Enable default backup schedule:       yes
   s) Save config to file
   x) Expand menu
   q) Quit

Address unconfigured (**) items  (? - help)
```

Figure 8.157

In the above image it shows that the Zimbra web interface is available at http://itit-mail-server.itit.com and it also shows that a license has not been provided for this installation.

12. To set the administrator password press 7 at the prompt. A new menu displaying the server settings will be displayed. Press 4 and enter a password for the administrator account.

```
Address unconfigured (**) items  (? - help) 7
```

Figure 8.158

```
   1) Status:                                  Enabled
   2) Create Admin User:                       yes
   3) Admin user to create:                    admin@itit-mail-server.itit.com
** 4) Admin Password                          UNSET
   5) Anti-virus quarantine user:              virus-quarantine.eq9zjbqf@itit-mail
-server.itit.com
   6) Enable automated spam training:          yes
   7) Spam training user:                      spam.fazarfct@itit-mail-server.itit
.com
   8) Non-spam(Ham) training user:             ham.djtftfd4@itit-mail-server.itit.
com
   9) SMTP host:                               itit-mail-server.itit.com
  10) Web server HTTP port:                    8080
  11) Web server HTTPS port:                   8443
  12) Web server mode:                         https
  13) IMAP server port:                        7143
  14) IMAP server SSL port:                    7993
  15) POP server port:                         7110
  16) POP server SSL port:                     7995
  17) Use spell check server:                  yes
  18) Spell server URL:                        http://itit-mail-server.itit.com:77
80/aspell.php
  19) Enable version update checks:            TRUE
  20) Enable version update notifications:     TRUE
  21) Version update notification email:       admin@itit-mail-server.itit.com
  22) Version update source email:             admin@itit-mail-server.itit.com
  23) Install mailstore (service webapp):      yes
  24) Install UI (zimbra,zimbraAdmin webapps): yes
**25) License filename:                        UNSET
```

Figure 8.159

Enter a strong password with at least 6 characters and press Enter.

```
Select, or 'r' for previous menu [r] 4

Password for admin@itit-mail-server.itit.com (min 6 characters): [9ZfILv8ndE]
```

Figure 8.160

Once the password has been set, enter 25 and enter the file location of the Zimba license.

```
Enter the name of the file that contains the license: /tmp/ZCSLicense.xml
```

Figure 8.161

Once the administrator password and the license has been entered, press r to return to the previous menu and then press “S” to save the configuration.

```
   s) Save config to file
   x) Expand menu
   q) Quit

Address unconfigured (**) items  (? - help) s

Save config in file: [/opt/zimbra/config.11611]
Saving config in /opt/zimbra/config.11611...done.
```

Figure 8.162

13. In the next menu, press 'a' to apply changes to the system and confirm the installation process.

```
Main menu

   1) Common Configuration:
   2) zimbra-ldap:                          Enabled
   3) zimbra-logger:                        Enabled
   4) zimbra-mta:                           Enabled
   5) zimbra-dnscache:                      Enabled
   6) zimbra-snmp:                          Enabled
   7) zimbra-store:                         Enabled
   8) zimbra-spell:                         Enabled
   9) zimbra-convertd:                      Enabled
  10) zimbra-proxy:                         Enabled
  11) Default Class of Service Configuration:
  12) Enable default backup schedule:       yes
   s) Save config to file
   x) Expand menu
   q) Quit

*** CONFIGURATION COMPLETE - press 'a' to apply
Select from menu, or press 'a' to apply config (? - help) a
Save configuration data to a file? [Yes] Yes
Save config in file: [/opt/zimbra/config.65479]
Saving config in /opt/zimbra/config.65479...done.
The system will be modified - continue? [No] Yes
```

Figure 8.163

The installation process will finish up. Press Enter to exit the installation when completed.

```
Finished installing network zimlets.
Restarting mailboxd...done.
Creating galsync account for default domain...done.
Checking if the NG started running...done.
Setting up zimbra crontab...done.


Moving /tmp/zmsetup.20200915-105518.log to /opt/zimbra/log


Configuration complete - press return to exit
```

Figure 8.164

14. To enable logging with syslog add the below lines to the file /etc./rsyslog.conf

```
# Provides UDP syslog reception
# for parameters see http://www.rsyslog.com/doc/imudp.html
module(load="imudp") # needs to be done just once
input(type="imudp" port="514")
```

Figure 8.165

15. Start and enable the rsyslog.service and run the zmsysslogsetup to update logging options.

```
[itit-administrator@itit-mail-server ~]$ sudo systemctl start rsyslog
[itit-administrator@itit-mail-server ~]$ sudo systemctl enable rsyslog
[itit-administrator@itit-mail-server ~]$
```

Figure 8.166

```
[itit-administrator@mail ~]$ sudo /opt/zimbra/libexec/zmsyslogsetu
p
updateSyslog: Updating /etc/rsyslog.conf...done.
```

Figure 8.167

16. Login as the Zimbra service user and run the zmupdateauthkeys to populate the SSH keys and restart the Zimbra service.

```
[itit-administrator@mail ~]$ sudo su - zimbra
Last login: Wed Sep 16 09:00:40 UTC 2020 on pts/1
Last failed login: Wed Sep 16 09:01:07 UTC 2020 on pts/1
There was 1 failed login attempt since the last successful login.
[zimbra@mail ~]$ zmupdateauthkeys
Updating keys for mail.itit.com
Fetching key for mail.itit.com
Updating keys for mail.itit.com
Updating /opt/zimbra/.ssh/authorized_keys
[zimbra@mail ~]$ zmcontrol restart
Host mail.itit.com
        Stopping vmware-ha...Done.
        Stopping zmconfigd...Done.
        Stopping zimlet webapp...Done.
        Stopping zimbraAdmin webapp...Done.
        Stopping zimbra webapp...Done.
        Stopping service webapp...Done.
        Stopping stats...Done.
        Stopping mta...Done.
        Stopping spell...Done.
        Stopping snmp...Done.
        Stopping cbpolicyd...Done.
        Stopping archiving...Done.
        Stopping opendkim...Done.
        Stopping amavis...Done.
        Stopping antivirus...Done.
        Stopping antispam...Done.
        Stopping proxy...Done.
        Stopping memcached...Done.
```

Figure 8.168

17. Once Zimbra restarts run the command *zmcontrol status* to check the status of the running services.

```
[zimbra@mail ~]$ zmcontrol status
Host mail.itit.com
        amavis                  Running
        antispam                Running
        convertd                Running
        dnscache                Running
        ldap                    Running
        logger                  Running
        mailbox                 Running
        memcached               Running
        mta                     Running
        opendkim                Running
        proxy                   Running
        service webapp          Running
        snmp                    Running
        spell                   Running
        stats                   Running
        zimbra webapp           Running
        zimbraAdmin webapp      Running
        zimlet webapp           Running
        zmconfigd               Running
```

Figure 8.169

18. Configure the firewall to allow Zimbra via the port 25 (SMTP), 80(HTTP), 443 (HTTPS), 110 (POP3), 389 (LDAP querying)

```
[itit-administrator@mail zimbra]$ sudo firewall-cmd --permanent --add-port=25/tcp
success
[itit-administrator@mail zimbra]$ sudo firewall-cmd --permanent --add-port=80/tcp
success
[itit-administrator@mail zimbra]$ sudo firewall-cmd --permanent --add-port=110/tcp
success
[itit-administrator@mail zimbra]$ sudo firewall-cmd --permanent --add-port=143/tcp
success
[itit-administrator@mail zimbra]$ sudo firewall-cmd --permanent --add-port=389/tcp
success
[itit-administrator@mail zimbra]$ sudo firewall-cmd --permanent --add-port=443/tcp
success
[itit-administrator@mail zimbra]$ sudo firewall-cmd --reload
success
[itit-administrator@mail zimbra]$
```

Figure 8.170

19. Enter the URL of the mail server and log in to gain access to the web interface.

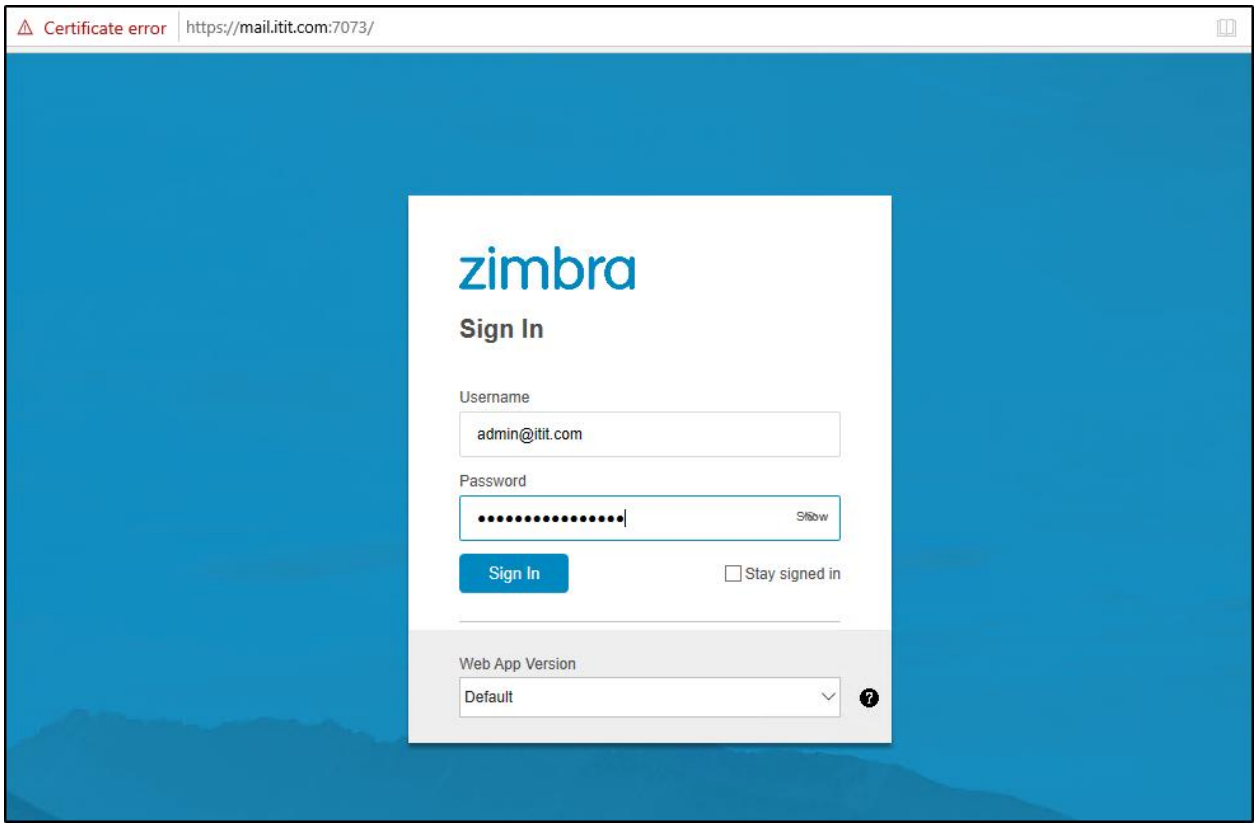


Figure 8.171

The mail of the administrator account with a few log emails are visible once you log in. The administrator account will receive e-mails when any change in the system occurs.

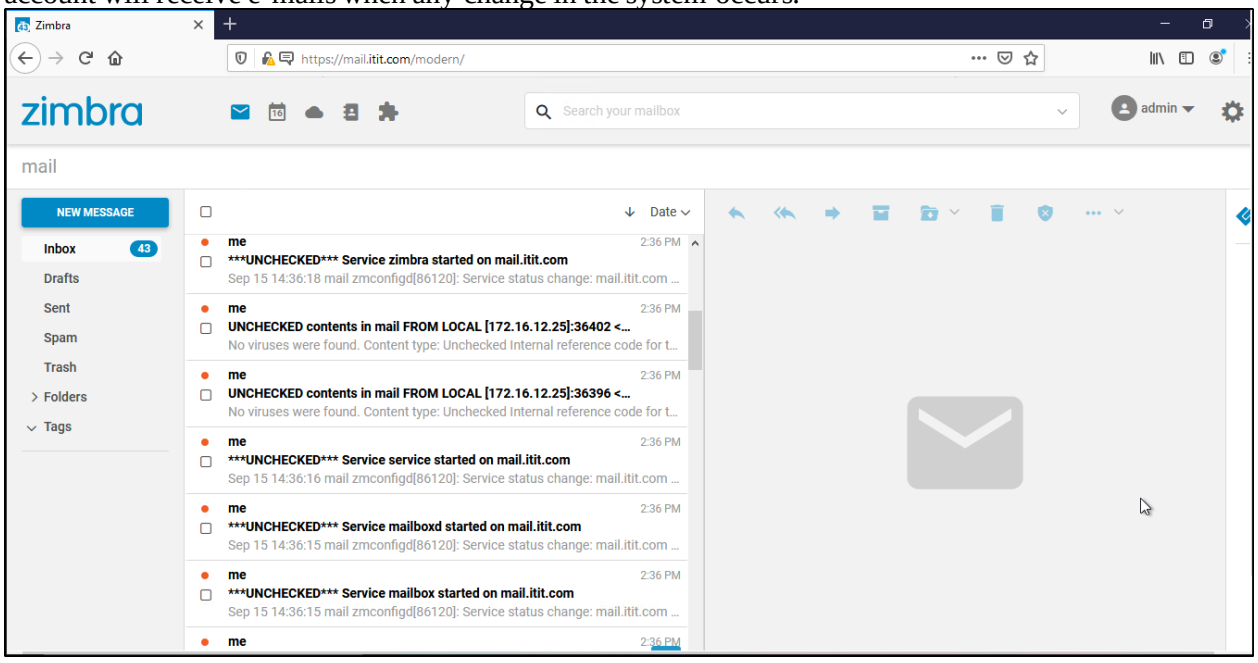


Figure 8.172

20. To integrate the Zimbra mail server with Active Directory login in to the administrator panel at https://mail.itit.com:7071/

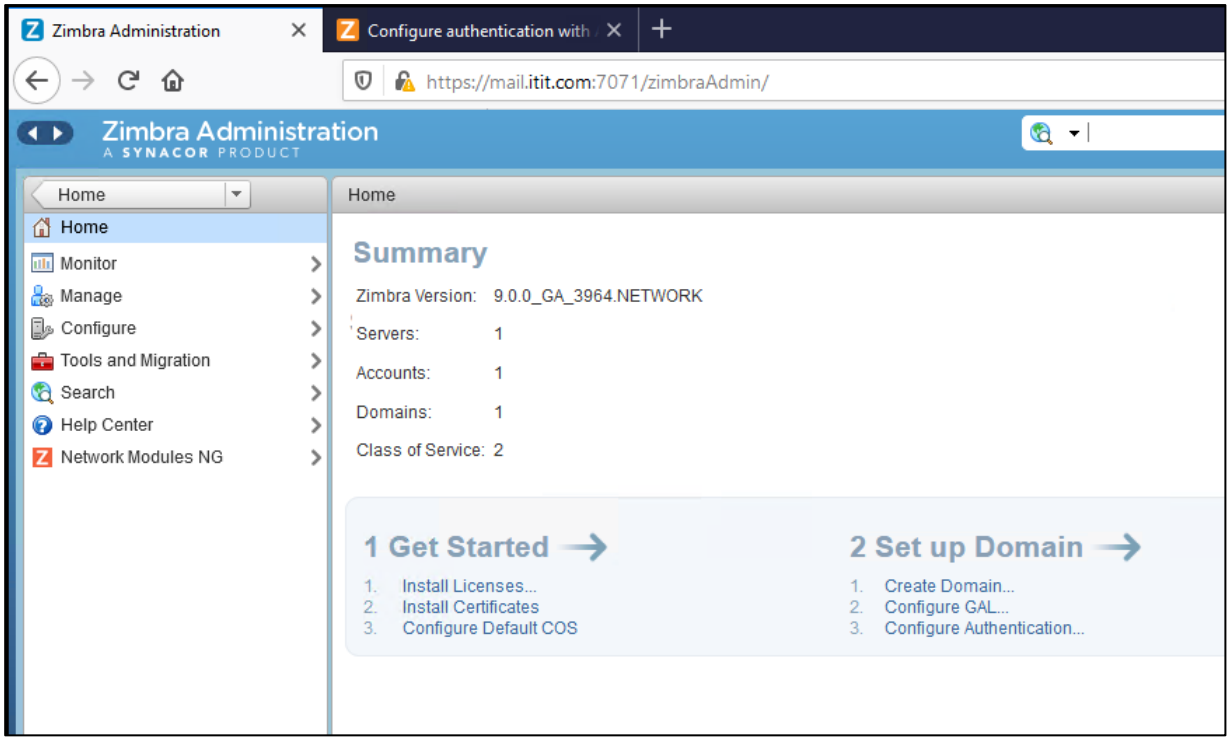


Figure 8.173

21. From the side menu select Configure, then select Domain, and click on the gear icon on the right and select New to add a new domain.

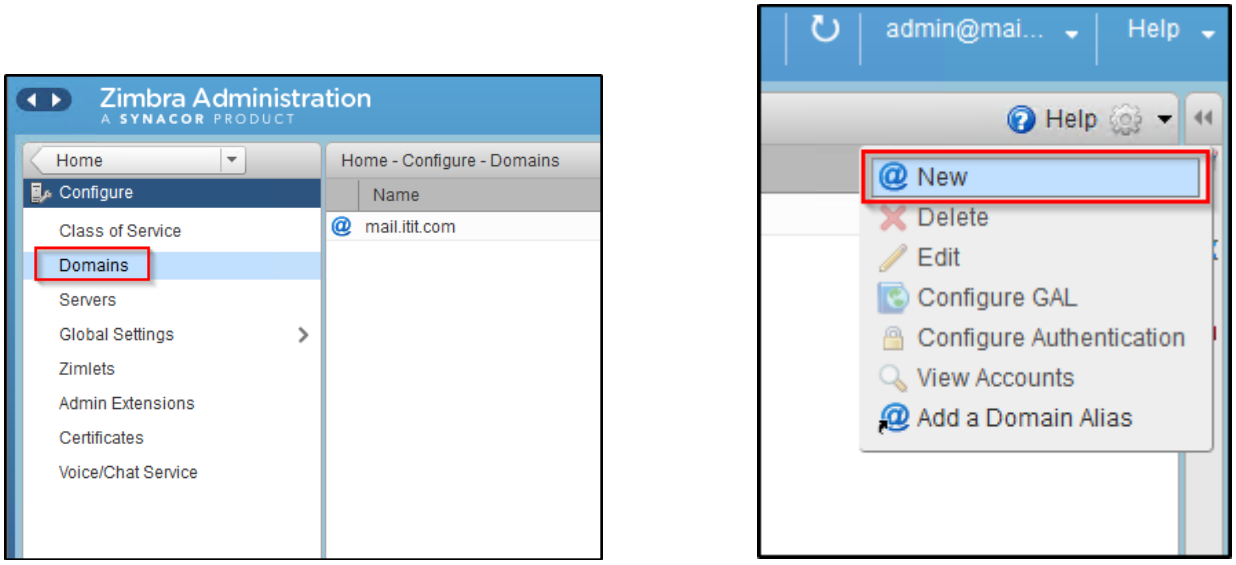


Figure 8.174

22. Enter the information on the domain and provide details about the domain controller as shown below.

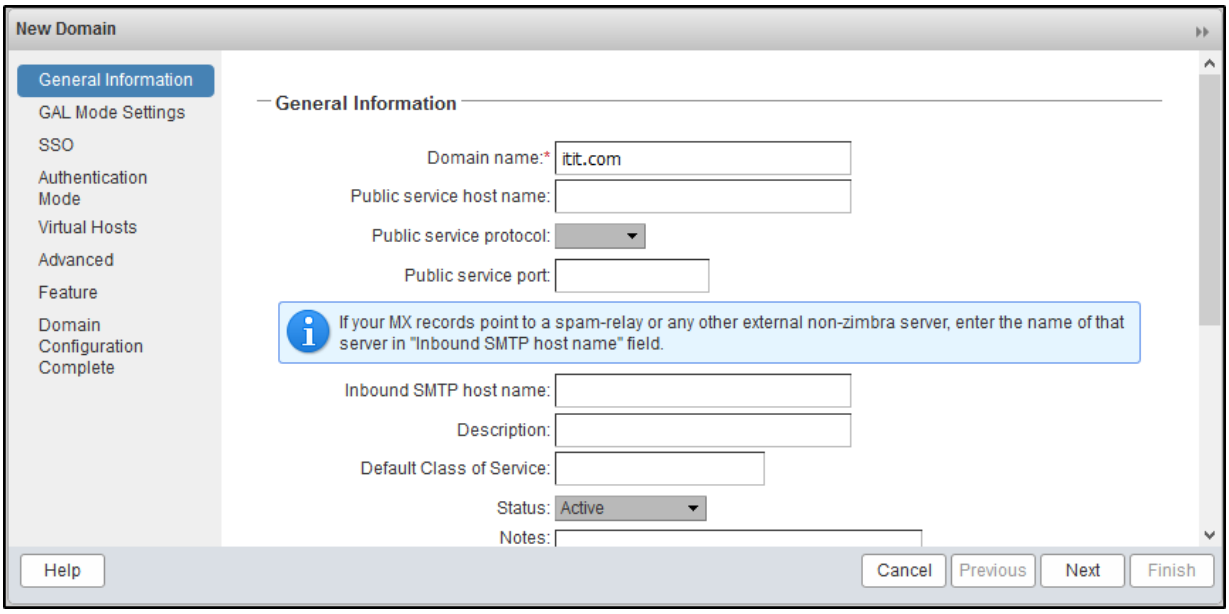


Figure 8.175

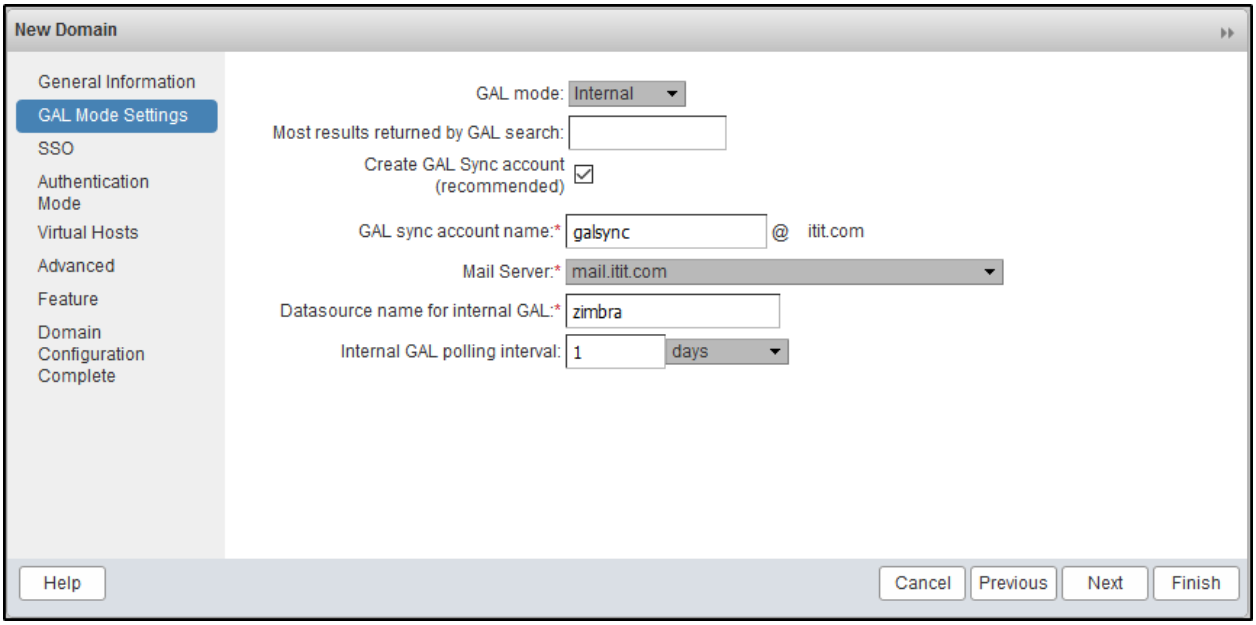


Figure 8.176

Set authentication mechanism as External Active Directory and provide the fully qualified domain name or the IP address of the Active Directory domain controller.

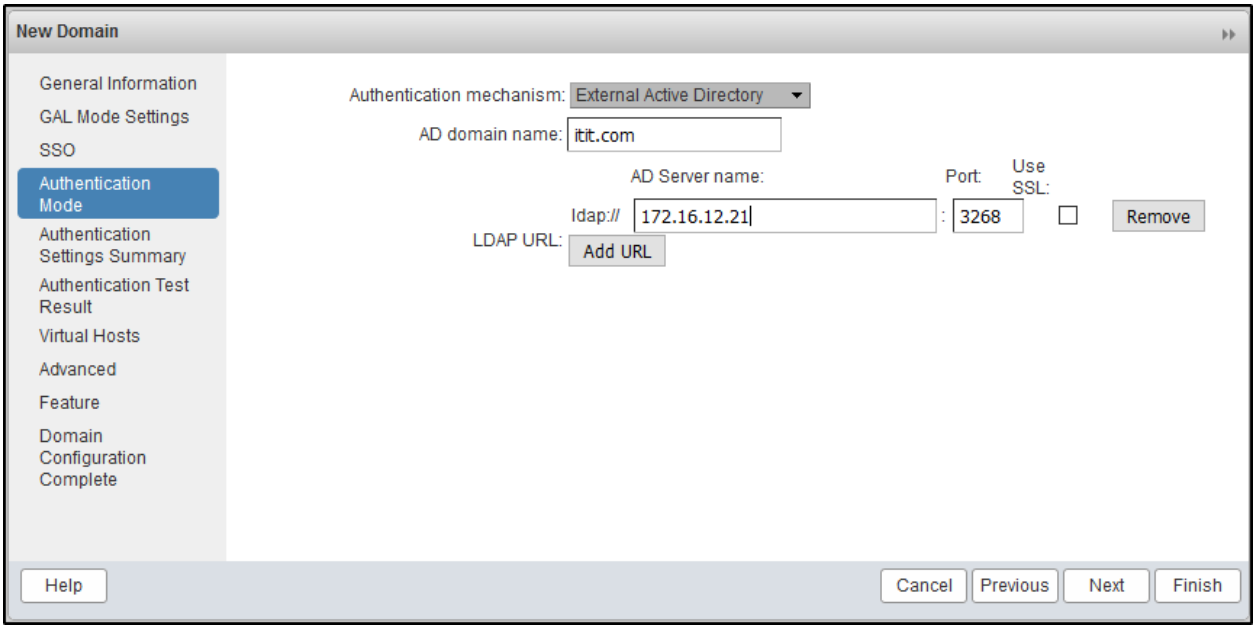


Figure 8.177

Provide the username and password of a domain account to test authentication.

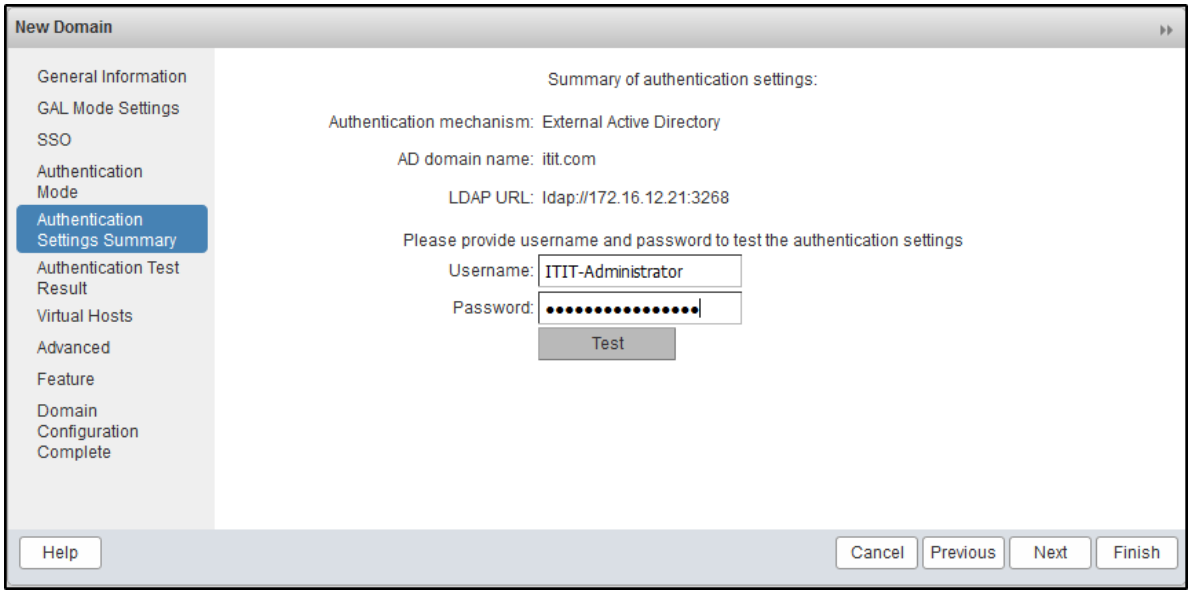


Figure 8.178

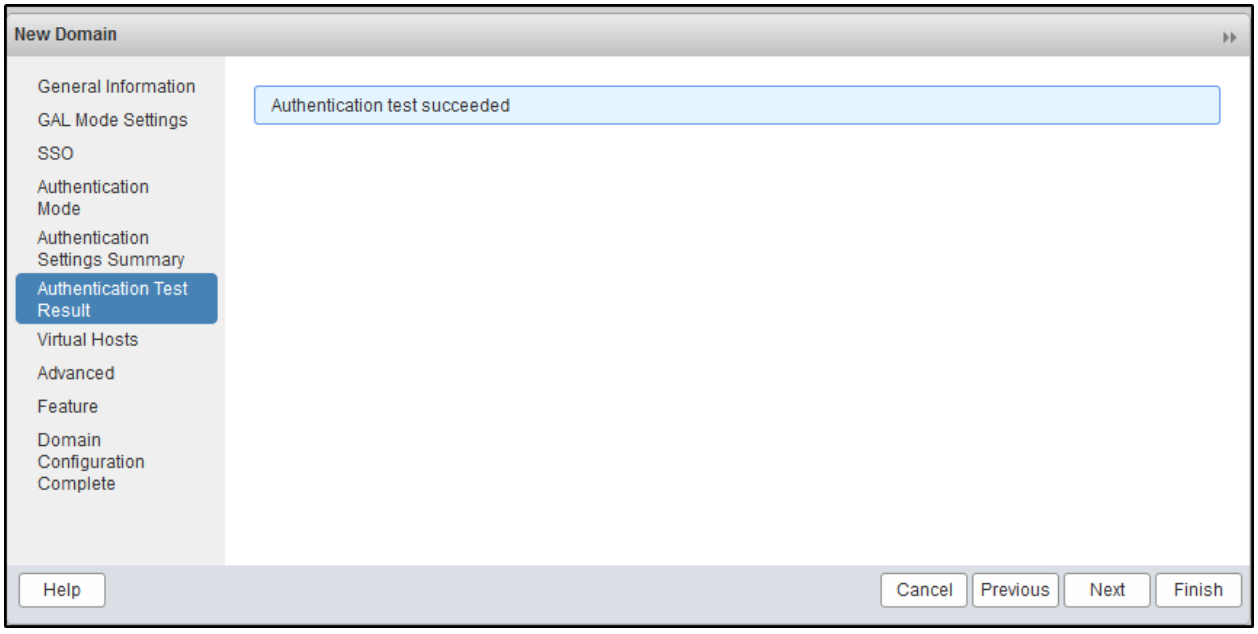


Figure 8.179

The next two steps can be skipped and the domain adding process can be finished by clicking the Finish button.

At the domains menu the Active Directory domain will now be visible.

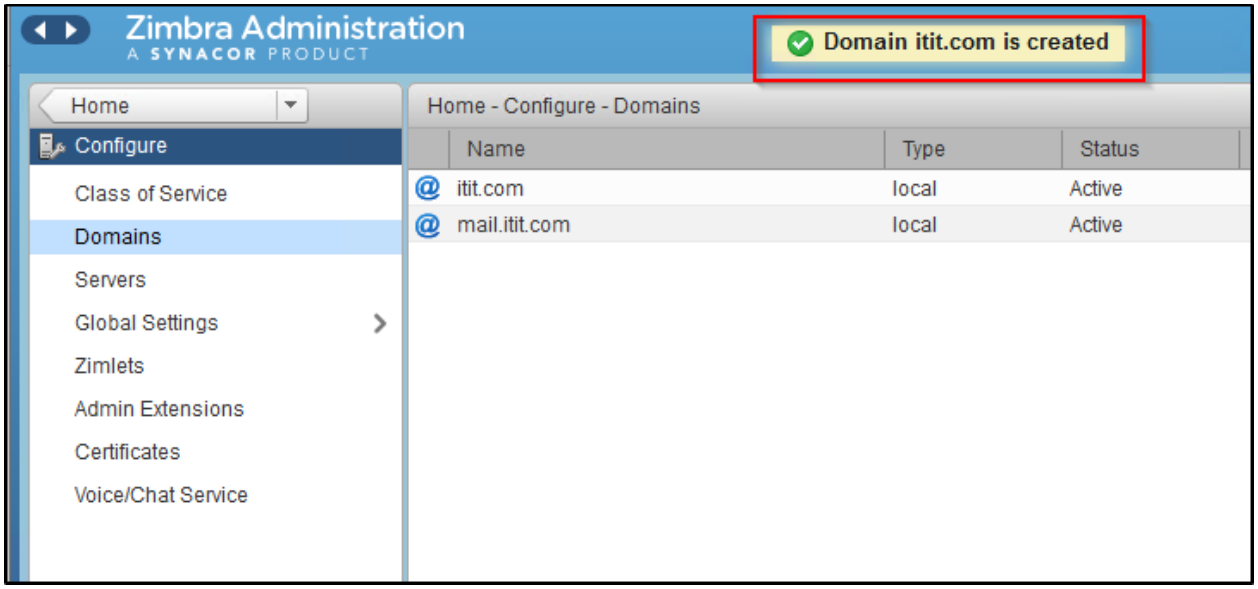


Figure 8.180

To test the authentication of users in Active Directory for Zimbra services, two test users were created as follows.

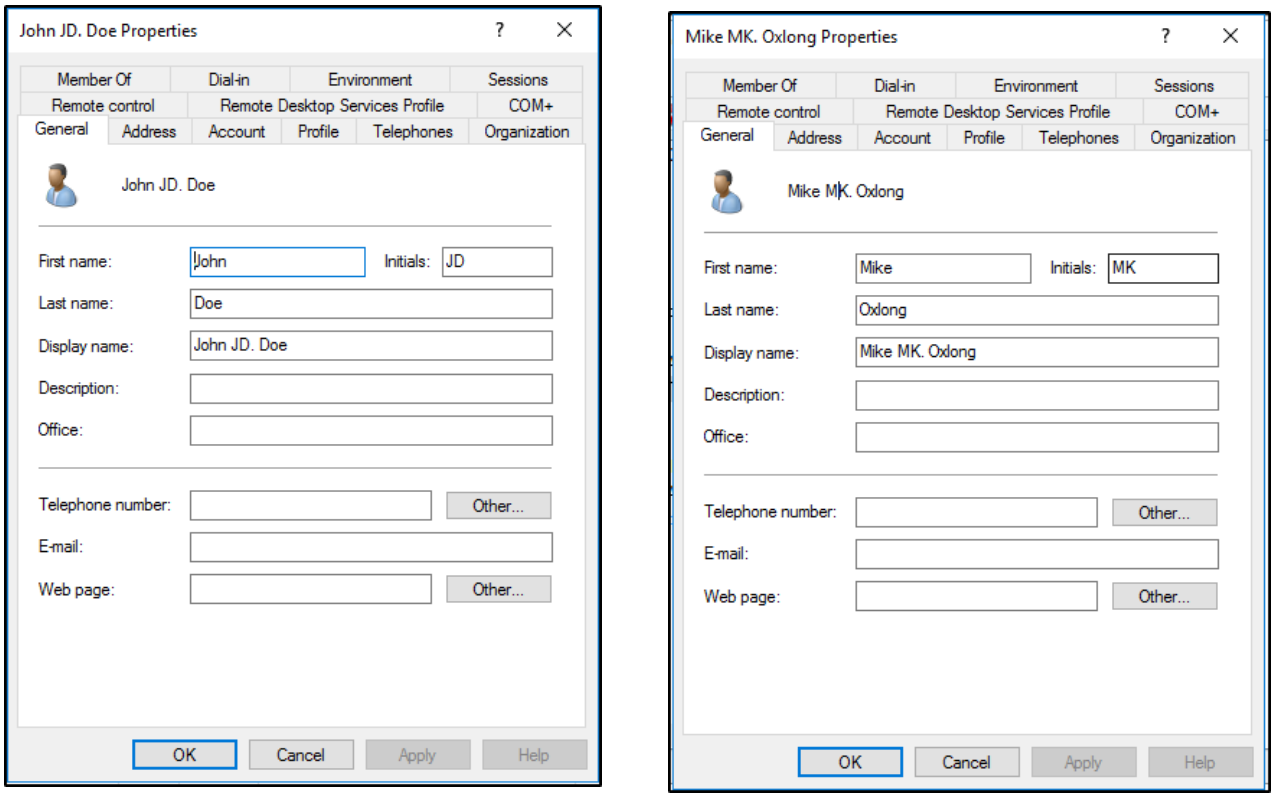


Figure 8.181

The user profiles were added in the Zimbra mail server as follows,

23. From the side menu at the administrator panel select Manage, and then select Accounts. Click on the gear icon on the right and click New.

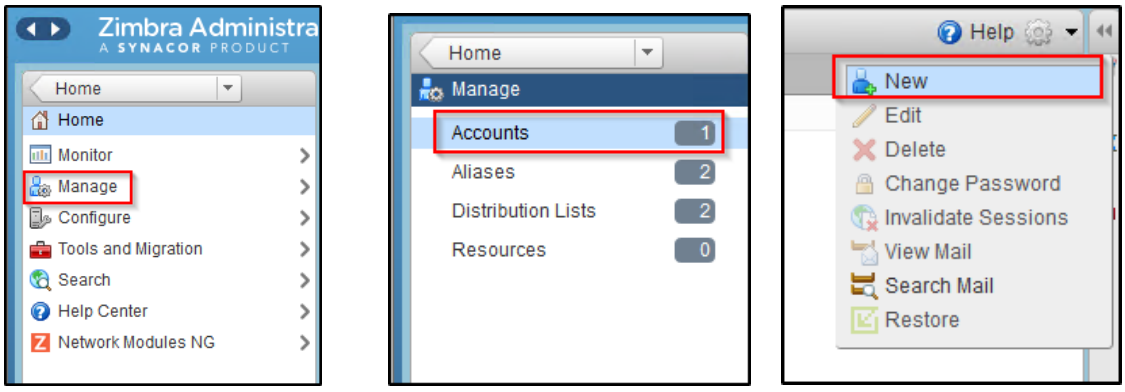


Figure 8.182

24. Add user details as follows and click Finish.

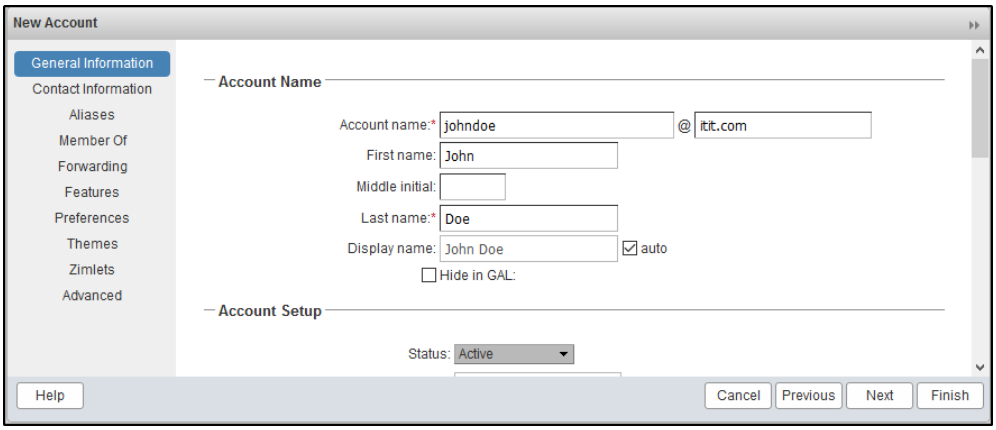


Figure 8.183

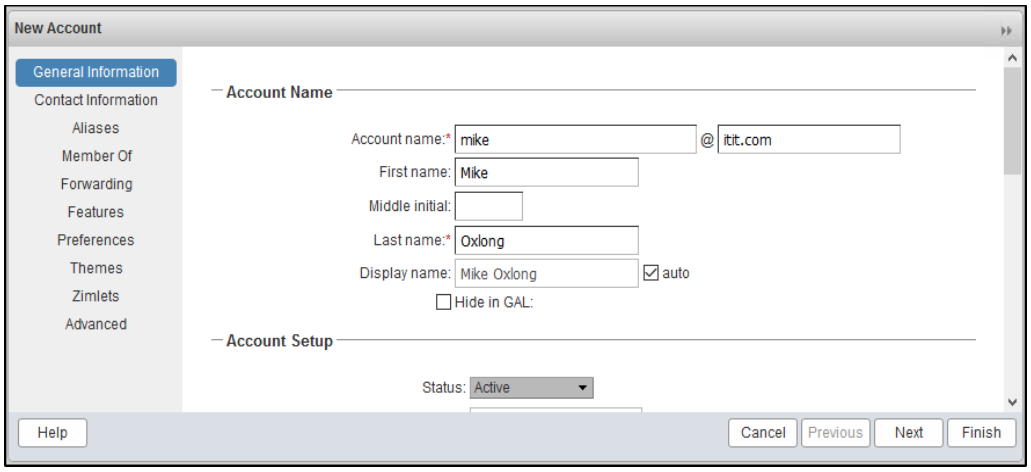


Figure 8.184

25. In the below diagram the user John Doe has logged into his Zimbra mailbox and is sending a mail to the user Mike.

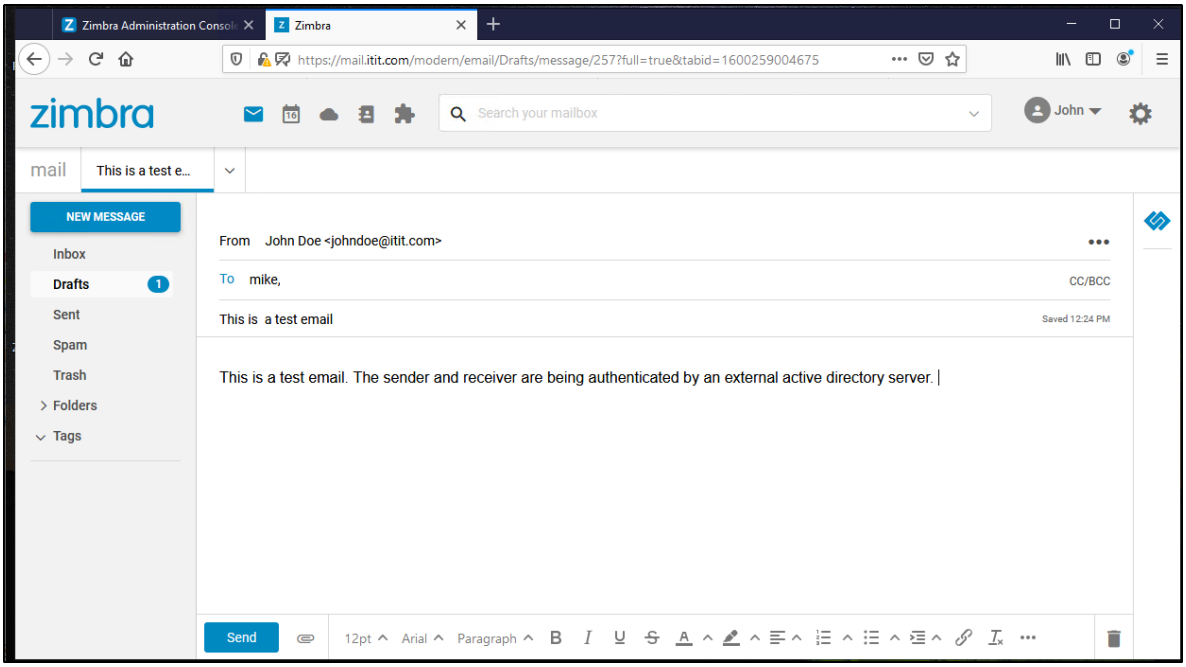


Figure 8.185

The mail was sent successfully to the user Mike.

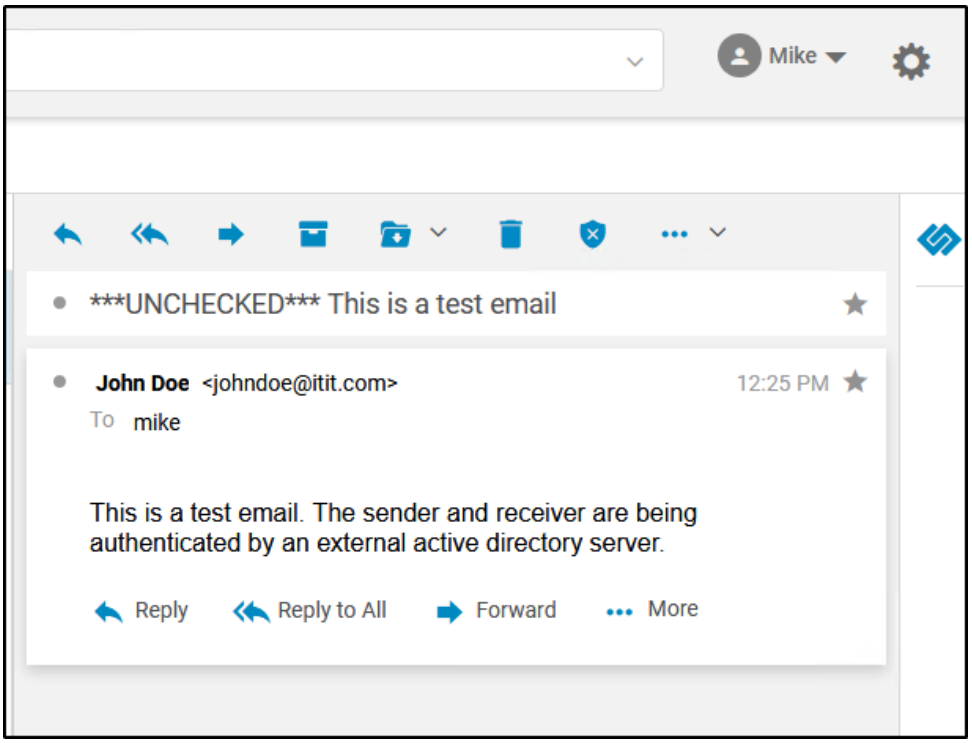


Figure 8.186

Implementing an internal File Server

01. Install the required Samba packages using the YUM package manager. Press Y to confirm the download and installation of the packages when prompted

```
[itit-administrator@itit-file-server ~]$ sudo yum install samba samba-winbind
samba-client samba-winbind-clients
Last metadata expiration check: 0:07:13 ago on Wed 16 Sep 2020 02:15:24 PM UTC
.
Dependencies resolved.
================================================================================
 Package                  Arch       Version                Repository  Size
================================================================================
Installing:
 samba                    x86_64     4.11.2-13.el8          BaseOS     766 k
 samba-client             x86_64     4.11.2-13.el8          BaseOS     658 k
 samba-winbind            x86_64     4.11.2-13.el8          BaseOS     575 k
 samba-winbind-clients    x86_64     4.11.2-13.el8          BaseOS     150 k
Installing dependencies:
 perl-Carp                noarch     1.42-396.el8           BaseOS      30 k
 perl-Errno               x86_64     1.28-416.el8           BaseOS      76 k
 perl-Exporter            noarch     5.72-396.el8           BaseOS      34 k
 perl-File-Path           noarch     2.15-2.el8             BaseOS      38 k
 perl-IO                  x86_64     1.38-416.el8           BaseOS     141 k
 perl-PathTools           x86_64     3.74-1.el8             BaseOS      90 k
 perl-Scalar-List-Utils   x86_64     3:1.49-2.el8           BaseOS      68 k
 perl-Socket              x86_64     4:2.027-3.el8          BaseOS      59 k
 perl-Text-Tabs+Wrap      noarch     2013.0523-395.el8      BaseOS      24 k
 perl-Unicode-Normalize   x86_64     1.25-396.el8           BaseOS      82 k
```

Figure 8.187

02. Start the smb and nmb services. Ensure that both services are active and running.

```
[itit-administrator@itit-file ~]$ sudo systemctl start smb nmb
[itit-administrator@itit-file ~]$ sudo systemctl status smb
● smb.service - Samba SMB Daemon
   Loaded: loaded (/usr/lib/systemd/system/smb.service; disabled; vendor >
   Active: active (running) since Wed 2020-09-16 15:12:20 UTC; 6s ago
     Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
 Main PID: 28735 (smbd)
   Status: "smbd: ready to serve connections..."
    Tasks: 4 (limit: 21780)
   Memory: 9.8M
   CGroup: /system.slice/smb.service
           ├─28735 /usr/sbin/smbd --foreground --no-process-group
           ├─28738 /usr/sbin/smbd --foreground --no-process-group
           ├─28739 /usr/sbin/smbd --foreground --no-process-group
           └─28743 /usr/sbin/smbd --foreground --no-process-group
```

Figure 8.188

```
[itit-administrator@itit-file ~]$ sudo systemctl status nmb
● nmb.service - Samba NMB Daemon
   Loaded: loaded (/usr/lib/systemd/system/nmb.service; disabled; vendor >
   Active: active (running) since Wed 2020-09-16 15:12:19 UTC; 43s ago
     Docs: man:nmbd(8)
           man:samba(7)
           man:smb.conf(5)
 Main PID: 28736 (nmbd)
   Status: "nmbd: ready to serve connections..."
    Tasks: 1 (limit: 21780)
   Memory: 2.6M
   CGroup: /system.slice/nmb.service
           └─28736 /usr/sbin/nmbd --foreground --no-process-group
```

Figure 8.189

03. Edit the Samba configuration file at /etc/samba/smb.conf and add the below entries. The ITIT-Shared-Drive is the network drive that will be shared across the domain.

```
[global]
        kerberos method = secrets and keytab
        realm = ITIT.COM
        security = ADS
        template shell = /bin/bash
        winbind enum groups = Yes
        winbind enum users = Yes
        winbind separator = +
        workgroup = ITIT
        idmap config * : rangesize = 1000000
        idmap config * : range = 1000000-19999999
        idmap config * : backend = autorid
```

Figure 8.190

```
[ITIT-Shared-Drive]
        comment = ITIT Network Drive
        path = /home/itit-administrator/ITIT-Shared-Drive
        valid users = @FTPUsers
        browsable = yes
        public = yes
        writable = yes
        read only = no
        create mask = 0777
        directory mask = 0777
        hide unreadable = yes
```

Figure 8.191

04. Enable the Samba Winbind daemon and restart the smb and nmb services.

```
[itit-administrator@itit-file ~]$ sudo systemctl enable winbind --now
[itit-administrator@itit-file ~]$ sudo systemctl status winbind.service
● winbind.service - Samba Winbind Daemon
   Loaded: loaded (/usr/lib/systemd/system/winbind.service; enabled; vend>
   Active: active (running) since Wed 2020-09-16 15:09:42 UTC; 12s ago
     Docs: man:winbindd(8)
           man:samba(7)
           man:smb.conf(5)
 Main PID: 28670 (winbindd)
   Status: "winbindd: ready to serve connections..."
    Tasks: 2 (limit: 21780)
   Memory: 4.8M
   CGroup: /system.slice/winbind.service
           ├─28670 /usr/sbin/winbindd --foreground --no-process-group
           └─28672 /usr/sbin/winbindd --foreground --no-process-group
```

Figure 8.192

```
[itit-administrator@itit-file ~]$ sudo systemctl restart smb nmb
[itit-administrator@itit-file ~]$ sudo systemctl status smb nmb.service
● smb.service - Samba SMB Daemon
   Loaded: loaded (/usr/lib/systemd/system/smb.service; disabled; vendor >
   Active: active (running) since Wed 2020-09-16 15:22:26 UTC; 21s ago
     Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
 Main PID: 28924 (smbd)
   Status: "smbd: ready to serve connections..."
    Tasks: 4 (limit: 21780)
   Memory: 7.5M
   CGroup: /system.slice/smb.service
           ├─28924 /usr/sbin/smbd --foreground --no-process-group
           ├─28929 /usr/sbin/smbd --foreground --no-process-group
           ├─28930 /usr/sbin/smbd --foreground --no-process-group
           └─28931 /usr/sbin/smbd --foreground --no-process-group
```

Figure 8.193

```
● nmb.service - Samba NMB Daemon
   Loaded: loaded (/usr/lib/systemd/system/nmb.service; disabled; vendor >
   Active: active (running) since Wed 2020-09-16 15:22:26 UTC; 21s ago
     Docs: man:nmbd(8)
           man:samba(7)
           man:smb.conf(5)
 Main PID: 28927 (nmbd)
   Status: "nmbd: ready to serve connections..."
    Tasks: 1 (limit: 21780)
   Memory: 2.6M
   CGroup: /system.slice/nmb.service
           └─28927 /usr/sbin/nmbd --foreground --no-process-group
```

Figure 8.194

05. Join the Samba server to the domain. Ensure that the domain controller has an A record pointing to the Samba sever to prevent failure.

```
[itit-administrator@itit-file ~]$ sudo net ads join -U ITIT-Administrator
itit.com
Enter ITIT-Administrator's password:
Using short domain name -- ITIT
Joined 'ITIT-FILE' to dns domain 'itit.com'
[itit-administrator@itit-file ~]$
```

Figure 8.195

The server should now be visible in the domain controller.

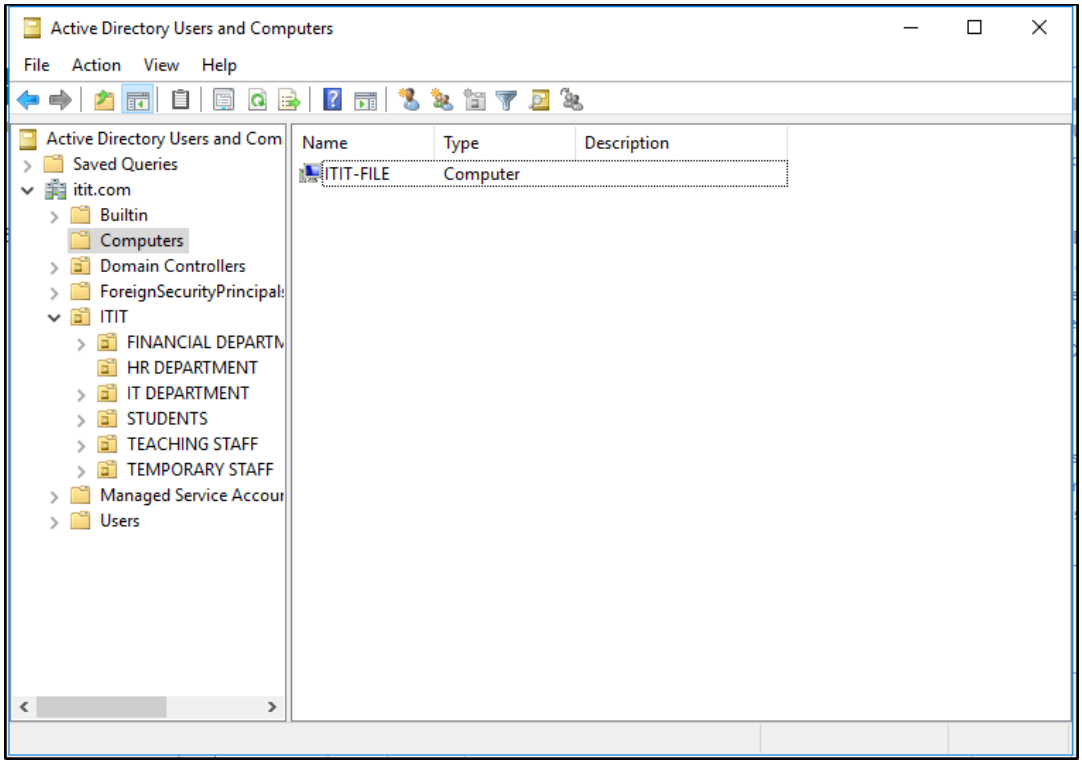


Figure 8.196

In this setup the shared drive was configured to be only accessible to the members of the Active Directory group FTPUsers. As proof of concept two dummy users were created, John Doe and Mike Oxlong, the first was added to the group FTPUsers while the latter wasn't. The user John Doe should be enable to

access the network drive successfully and use it for sharing files while the user Mike should be denied access.

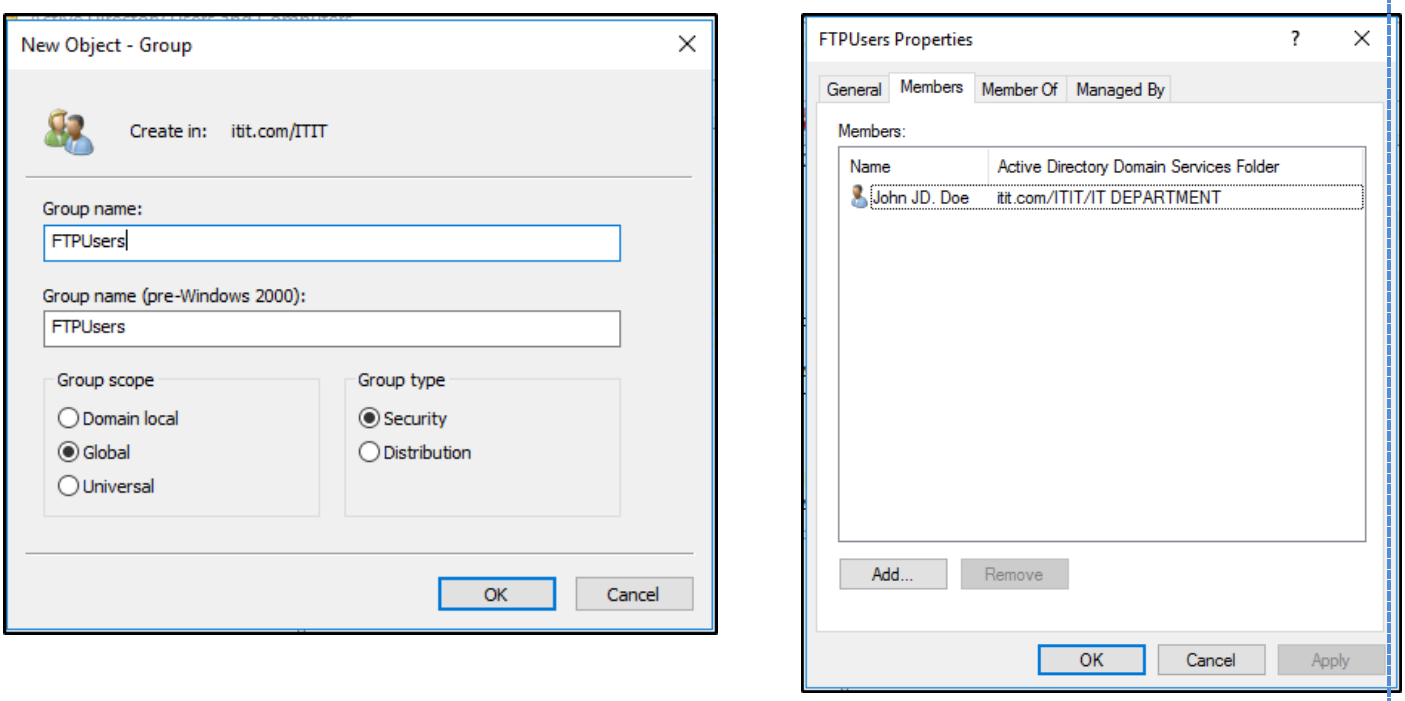


Figure 8.197

06. Map the network drive from a computer that has joined into the domain.

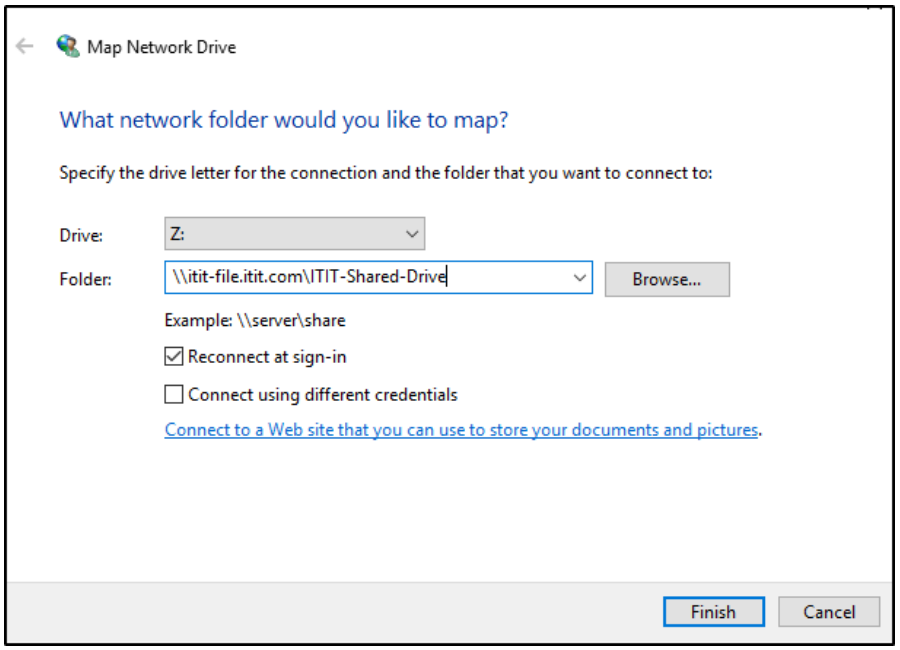


Figure 8.198

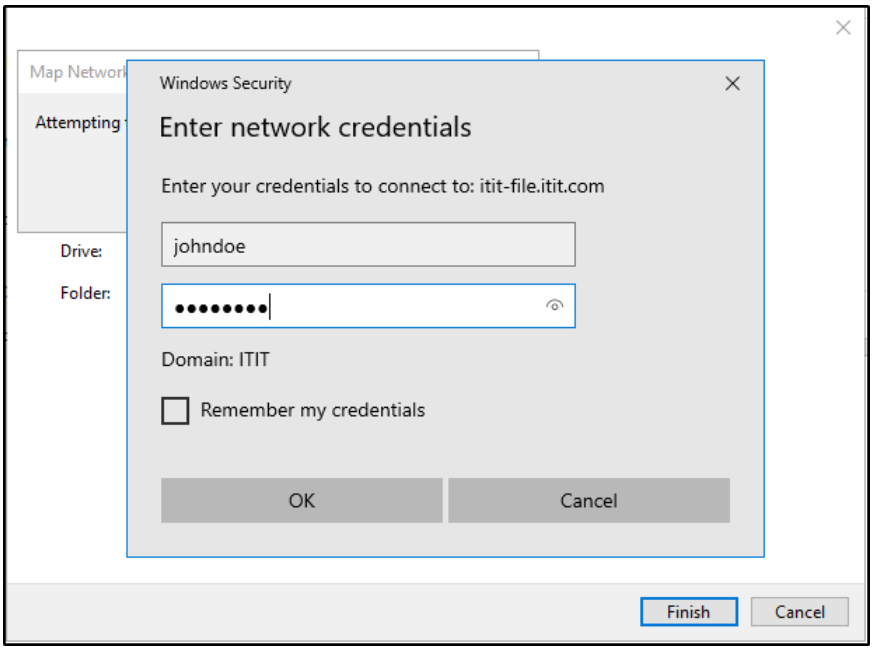


Figure 8.199

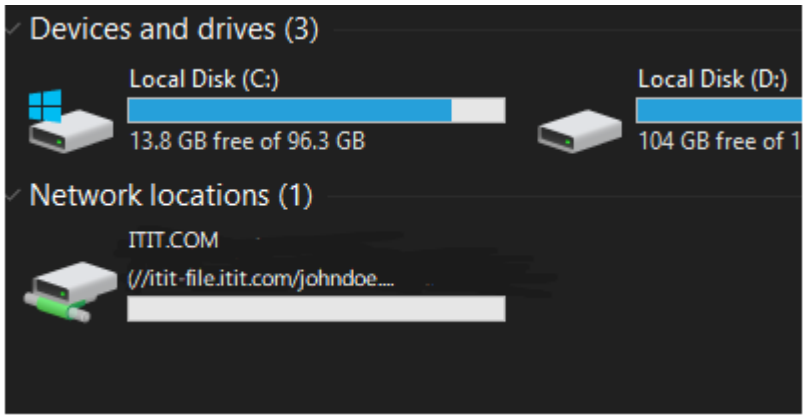


Figure 8.200

The user Mike is denied access as configured.

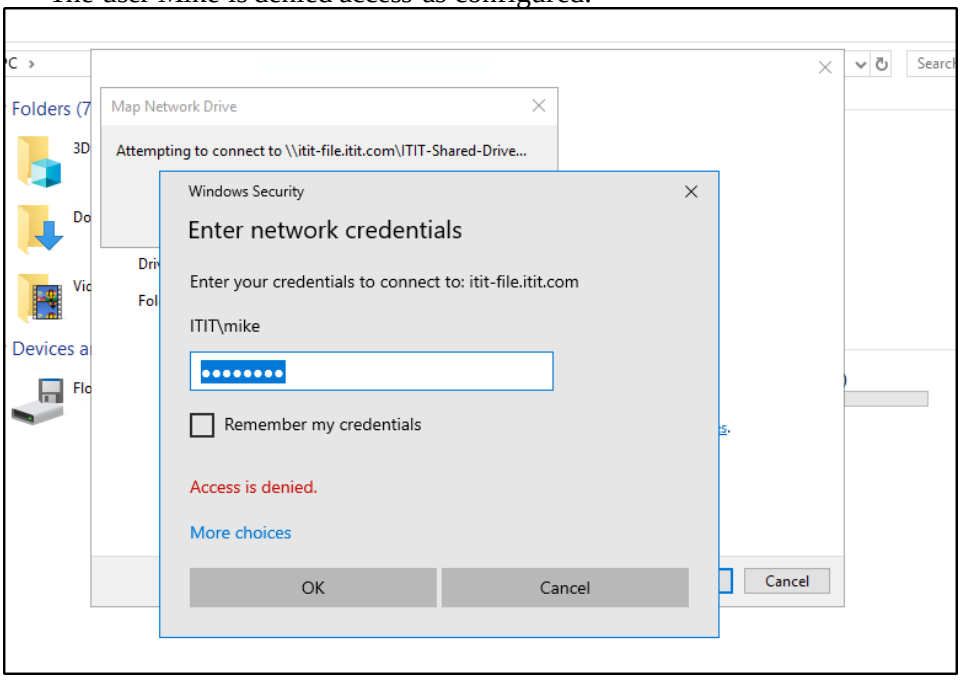


Figure 8.201

Implementing a Highly Available Web Server Cluster

A highly available web server cluster was implemented within the demilitarized zone by using two Apache servers and by using Keepalived. It provides a framework of health checking on multiple layers for server failover, and VRRPv2 to provide redundancy.
In this setup two web sites were required to be served to external hosts in a single Apache instance. This was easily addressed by using name-based virtual-hosting. In name-based virtual hosting, the server relies on the client to report the hostname as part of the HTTP headers. Using this technique, many different hosts can share the same IP address. The web server relies on the DNS records to map the hostnames to the respective sites.

01. The system information of the web server is as follows.

```
[itit-admin@itit-web01 ~]$ cat /etc/os-release
NAME="CentOS Linux"
VERSION="8 (Core)"
ID="centos"
ID_LIKE="rhel fedora"
VERSION_ID="8"
PLATFORM_ID="platform:el8"
PRETTY_NAME="CentOS Linux 8 (Core)"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:centos:centos:8"
HOME_URL="https://www.centos.org/"
BUG_REPORT_URL="https://bugs.centos.org/"

CENTOS_MANTISBT_PROJECT="CentOS-8"
CENTOS_MANTISBT_PROJECT_VERSION="8"
REDHAT_SUPPORT_PRODUCT="centos"
REDHAT_SUPPORT_PRODUCT_VERSION="8"

[itit-admin@itit-web01 ~]$ hostname
itit-web01.itit.com
[itit-admin@itit-web01 ~]$
```

Figure 8.203

02. Install the Apache httpd package using the YUM package manager.

```
[itit-admin@itit-web01 ~]$ sudo yum install httpd
Last metadata expiration check: 3:29:44 ago on Thu 17 Sep 2020 11:10:23 AM +0530.
Dependencies resolved.
================================================================================
 Package              Arch       Version                                Repository     Size
================================================================================
Installing:
 httpd                x86_64     2.4.37-21.module_el8.2.0+382+15b0afa8  AppStream     1.7 M
Installing dependencies:
 apr                  x86_64     1.6.3-9.el8                            AppStream     125 k
 apr-util             x86_64     1.6.1-6.el8                            AppStream     105 k
 centos-logos-httpd   noarch     80.5-2.el8                             BaseOS         24 k
 httpd-filesystem     noarch     2.4.37-21.module_el8.2.0+382+15b0afa8  AppStream      36 k
 httpd-tools          x86_64     2.4.37-21.module_el8.2.0+382+15b0afa8  AppStream     103 k
 mod_http2            x86_64     1.11.3-3.module_el8.2.0+307+4d18d695   AppStream     157 k
Installing weak dependencies:
 apr-util-bdb         x86_64     1.6.1-6.el8                            AppStream      25 k
 apr-util-openssl     x86_64     1.6.1-6.el8                            AppStream      27 k

Transaction Summary
================================================================================
Install  9 Packages

Total download size: 2.3 M
Installed size: 6.0 M
Is this ok [y/N]:
```

Figure 8.204

03. Enable and start the httpd.service and check status to confirm it is active and running.

```
[itit-admin@itit-web01 ~]$ systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)

Sep 17 14:36:57 itit-web01.itit.com systemd[1]: Starting The Apache HTTP Server...
Sep 17 14:36:58 itit-web01.itit.com systemd[1]: Started The Apache HTTP Server.
Sep 17 14:36:58 itit-web01.itit.com httpd[15448]: Server configured, listening on: port 80
Sep 17 14:37:33 itit-web01.itit.com systemd[1]: Stopping The Apache HTTP Server...
Sep 17 14:37:34 itit-web01.itit.com systemd[1]: Stopped The Apache HTTP Server.
[itit-admin@itit-web01 ~]$ sudo systemctl enable httpd.service
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/
httpd.service.
[itit-admin@itit-web01 ~]$ sudo systemctl start httpd.service
[itit-admin@itit-web01 ~]$ sudo systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2020-09-17 14:43:06 +0530; 6s ago
     Docs: man:httpd.service(8)
 Main PID: 26046 (httpd)
   Status: "Started, listening on: port 80"
    Tasks: 213 (limit: 4839)
   Memory: 24.9M
   CGroup: /system.slice/httpd.service
           ├─26046 /usr/sbin/httpd -DFOREGROUND
           ├─26047 /usr/sbin/httpd -DFOREGROUND
           ├─26048 /usr/sbin/httpd -DFOREGROUND
           ├─26049 /usr/sbin/httpd -DFOREGROUND
           └─26050 /usr/sbin/httpd -DFOREGROUND

Sep 17 14:43:06 itit-web01.itit.com systemd[1]: Starting The Apache HTTP Server...
Sep 17 14:43:06 itit-web01.itit.com systemd[1]: Started The Apache HTTP Server.
Sep 17 14:43:07 itit-web01.itit.com httpd[26046]: Server configured, listening on: port 80
[itit-admin@itit-web01 ~]$
```

Figure 8.205

04. Configure the firewall to allow Apache web traffic.

```
[itit-admin@itit-web01 ~]$ sudo firewall-cmd --permanent --add-service=http
success
[itit-admin@itit-web01 ~]$ sudo firewall-cmd --permanent --add-port=80/tcp
success
[itit-admin@itit-web01 ~]$ sudo firewall-cmd --reload
success
[itit-admin@itit-web01 ~]$ sudo firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: ens33
  sources:
  services: cockpit dhcpv6-client http ssh
  ports: 80/tcp
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Figure 8.206

05. Access the Apache default page by entering the IP address of the web server to confirm that Apache is running.

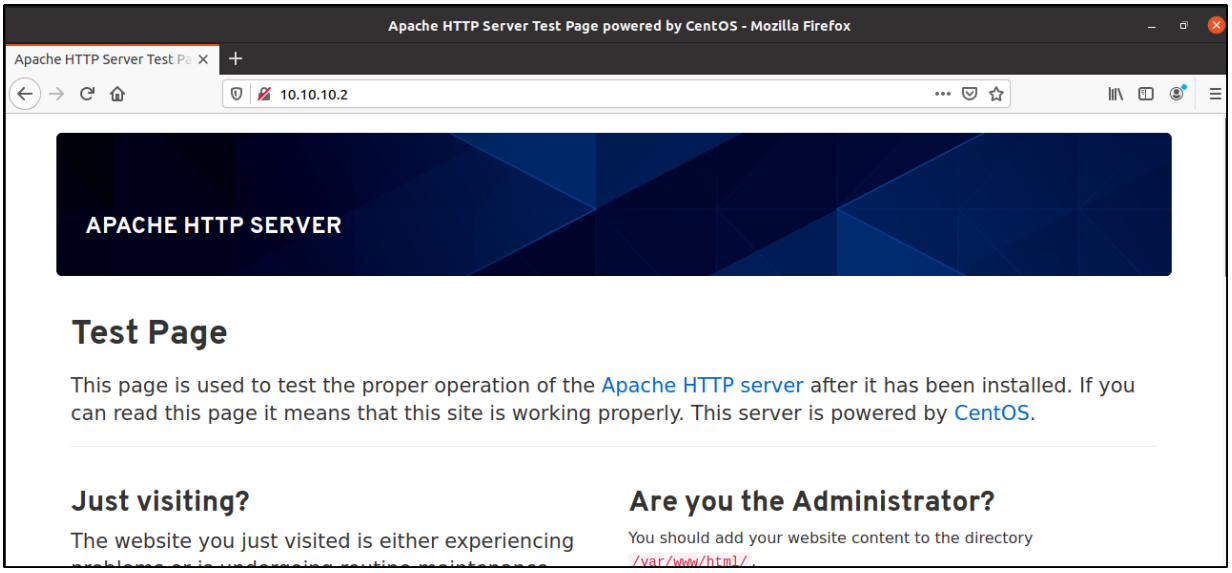


Figure 8.207

06. Create directory structures to store the necessary files for the websites. In this implementation two websites were created –www.itit.com and www.itit-lms.com, so two directories were created under /var/www/ directory. The created directories' ownership was assigned to the apache user and the /var/www directory structure was provided with default permissions.

```
[itit-admin@itit-web01 ~]$ sudo mkdir -p /var/www/itit.com/html
[itit-admin@itit-web01 ~]$ sudo mkdir -p /var/www/itit.com/logs
[itit-admin@itit-web01 ~]$ sudo mkdir -p /var/www/itit-lms.com/html
[itit-admin@itit-web01 ~]$ sudo mkdir -p /var/www/itit-lms.com/logs
[itit-admin@itit-web01 ~]$ sudo chown -R apache:apache /var/www/itit.com/
[itit-admin@itit-web01 ~]$ sudo chown -R apache:apache /var/www/itit-lms.com/
[itit-admin@itit-web01 ~]$ sudo chmod 755 /var/www/
[itit-admin@itit-web01 ~]$ _
```

Figure 8.208

07. Add the website files into the /html/ sub-directory of both the newly created directories.

08. Create separate directories to store the virtual hosts. The *sites-available* directory is used to store the virtual host information while the *sites-enabled* directory tells Apache which virtul hosts are ready to serve.

```
[itit-admin@itit-web01 ~]$ sudo mkdir -p /etc/httpd/sites-available
[itit-admin@itit-web01 ~]$ sudo mkdir -p /etc/httpd/sites-enable
[itit-admin@itit-web01 ~]$ _
```

Figure 8.209

09. Edit the Apache configuration file (httpd.conf) located at /etc/httpd/conf/ to inform Apache to look for virtual hosts in the *sites-enabled* directory. This is done by declaring an optional directory at the bottom of the file.

```
# Supplemental configuration
#
# Load config files in the "/etc/httpd/conf.d" directory, if any.
IncludeOptional conf.d/*.conf
IncludeOptional sites-enabled/*.conf_
```

Figure 8.210

10. Create virtual host configuration files in the *sites-available* directory. Create separate virtual host configuration files for each virtual host (itit.com and itit-lms.com).

```
[itit-admin@itit-web01 ~]$ sudo touch /etc/httpd/sites-available/itit.com.conf
[itit-admin@itit-web01 ~]$ sudo nano /etc/httpd/sites-available/itit.com.conf_
```

Figure 8.211

```
<VirtualHost *:80>

        ServerName www.itit.com
        ServerAlias itit.com
        DocumentRoot /var/www/itit.com/html/index.html
        ErrorLog /var/www/itit.com/logs/error.log
        CustomLog /var/www/itit.com/logs/access.log combined
</VirtualHost>_
```

Figure 8.212

```
[itit-admin@itit-web01 ~]$ sudo touch /etc/httpd/sites-available/itit-lms.com.conf
[itit-admin@itit-web01 ~]$ sudo nano /etc/httpd/sites-available/itit-lms.com.conf_
```

Figure 8.213

```
<VirtualHost *:80>

        ServerName www.itit-lms.com
        ServerAlias itit-lms.com
        DocumentRoot /var/www/itit-lms.com/html/index.html
        ErrorLog /var/www/itit-lms.com/logs/error.log
        CustomLog /var/www/itit-lms.com/logs/access.log

</VirtualHost>_
```

Figure 8.214

11. Enable the virtual hosts by creating symbolic links to their configuration files at *sites-available* directory in *sites-enabled* directory

```
[itit-admin@itit-web01 ~]$ sudo ln -s /etc/httpd/sites-available/itit.com.conf /etc/httpd/sites-enab
le/itit.com.conf
[itit-admin@itit-web01 ~]$ sudo ln -s /etc/httpd/sites-available/itit-lms.com.conf /etc/httpd/sites-
enable/itit-lms.com.conf
[itit-admin@itit-web01 ~]$
```

Figure 8.215

12. Adjust the SELinux permissions for the virtual hosts. Since a custom log directory was created, an error will be displayed when attempting to restart the httpd.service service. This is caused by SELinux denying Apache from writing into files (logs).

```
[itit-admin@itit-web01 ~]$ sudo ls -ldZ /var/www/itit.com/logs/
drwxr-xr-x. 2 apache apache unconfined_u:object_r:httpd_sys_content_t:s0 6 Sep 17 15:12 /var/www/iti
t.com/logs/
[itit-admin@itit-web01 ~]$ sudo semanage fcontext -a -t httpd_log_t "/var/www/itit.com/logs(/.*)?"
[itit-admin@itit-web01 ~]$ sudo restorecon -R -v /var/www/itit.com/logs
Relabeled /var/www/itit.com/logs from unconfined_u:object_r:httpd_sys_content_t:s0 to unconfined_u:o
bject_r:httpd_log_t:s0
[itit-admin@itit-web01 ~]$ ls -ldZ /var/www/itit-lms.com/logs/
drwxr-xr-x. 2 apache apache unconfined_u:object_r:httpd_sys_content_t:s0 6 Sep 17 15:12 /var/www/iti
t-lms.com/logs/
[itit-admin@itit-web01 ~]$ sudo semanage fcontext -a -t httpd_log_t "/var/www/itit-lms.com/logs(/.*)
?"
[itit-admin@itit-web01 ~]$ sudo restorecon -R -v /var/www/itit-lms.com/
Relabeled /var/www/itit-lms.com/logs from unconfined_u:object_r:httpd_sys_content_t:s0 to unconfined
_u:object_r:httpd_log_t:s0
[itit-admin@itit-web01 ~]$
```

Figure 8.216

13. Disable the Apache default page by commenting out the file at /etc/httpd/conf.d/welcome.conf

```
 GNU nano 2.9.8                    /etc/httpd/conf.d/welcome.conf

#
# This configuration file enables the default "Welcome" page if there
# is no default index page present for the root URL.  To disable the
# Welcome page, comment out all the lines below.
#
# NOTE: if this file is removed, it will be restored on upgrades.
#
#<LocationMatch "^/+$">
#  Options -Indexes
#  ErrorDocument 403 /noindex/index.html
#</LocationMatch>

#Alias /noindex /usr/share/httpd/noindex

#<Directory /usr/share/httpd/noindex>
#  Options MultiViews
#  DirectoryIndex index.html
#
#  AddLanguage en-US .en-US
#  AddLanguage es-ES .es-ES
#  AddLanguage zh-CN .zh-CN
#  AddLanguage zh-HK .zh-HK
#  AddLanguage zh-TW .zh-TW

#  LanguagePriority en
#  ForceLanguagePriority Fallback

#  AllowOverride None
#  Require all granted
#</Directory>
```

Figure 8.217

14. Restart the httpd.service service and check the contents of the websites' log directory. The log files should be created which holds proof that the virtual host configuration files were without error.

```
[itit-admin@itit-web01 ~]$ ls -l /var/www/itit.com/log/
total 0
-rw-r--r--. 1 root root 0 Sep 17 21:07 access.log
-rw-r--r--. 1 root root 0 Sep 17 21:07 error.log
[itit-admin@itit-web01 ~]$
```

Figure 8.218

```
[itit-admin@itit-web01 html]$ ls -l /var/www/itit-lms.com/log/
total 0
-rw-r--r--. 1 root root 0 Sep 17 21:19 access.log
-rw-r--r--. 1 root root 0 Sep 17 21:19 error.log
[itit-admin@itit-web01 html]$
```

Figure 8.219

15. Setup DNS records to point the www.itit.com and www.itit-lms.com to the web servers' IP addresses and check the websites using a browser.

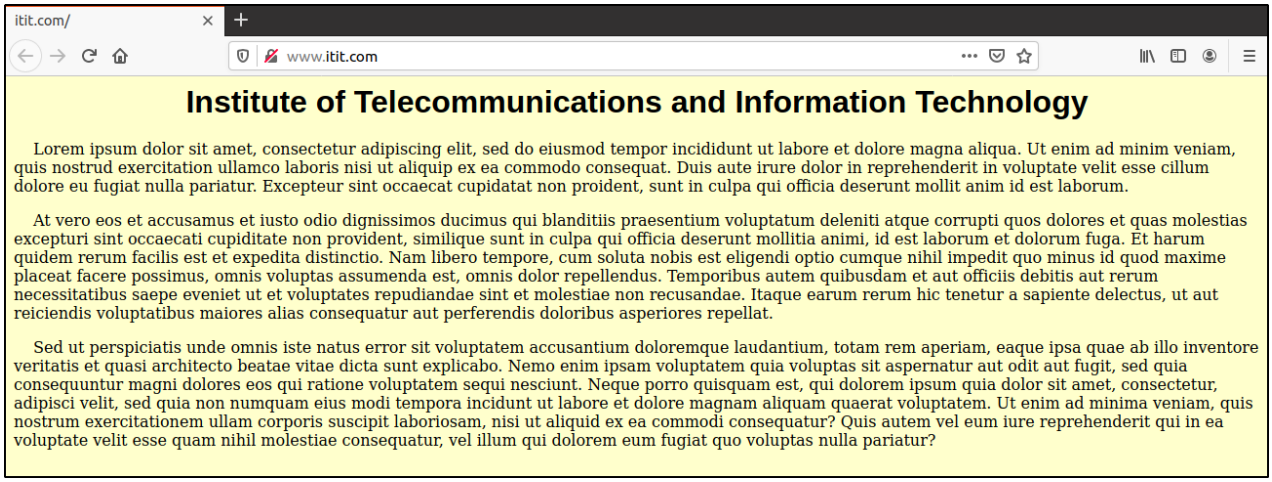


Figure 8.220

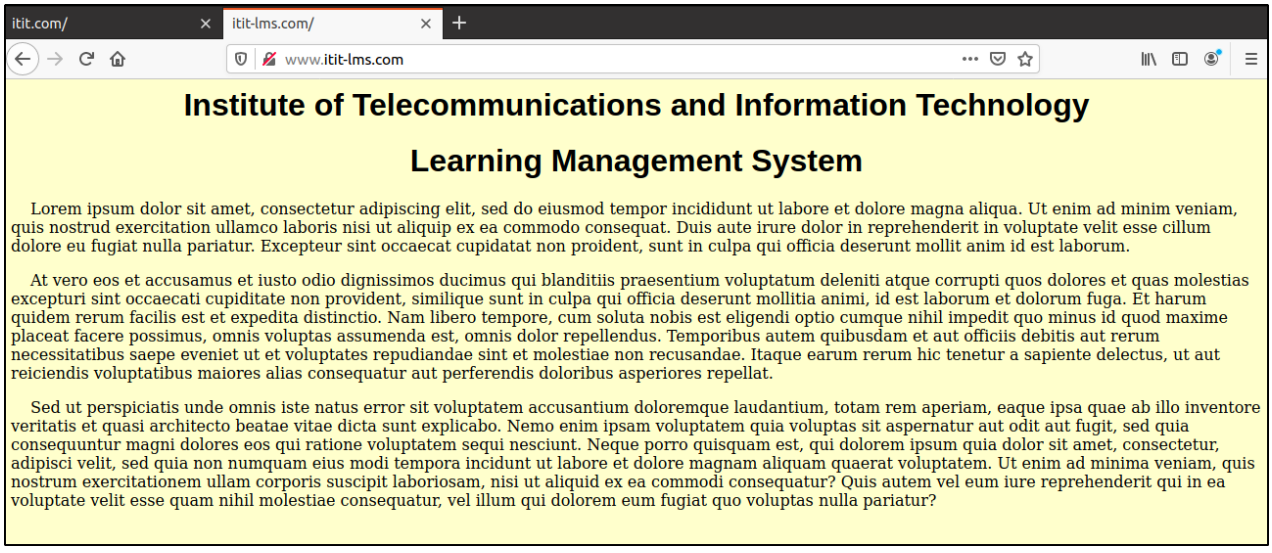


Figure 8.221

Configure the second web server (itit-web02.itit.com) as above before setting up high availability. Both web servers must be configured identically to avoid clashes in the content. The installation of Keepalived to provide the high availability features are documented below and just like the above configuration of the web server, configure both hosts in the same manner except where differences has to be made.

01. Install the Keepalived package using the YUM package manager.

```
[itit-admin@itit-web01 opt]$ yum install keepalived
Error: This command has to be run under the root user.
[itit-admin@itit-web01 opt]$ sudo yum install keepalived
Last metadata expiration check: 1:57:26 ago on Thu 17 Sep 2020 08:47:51 PM +0530.
Dependencies resolved.
================================================================================
 Package                 Arch       Version                          Repository     Size
================================================================================
Installing:
 keepalived              x86_64     2.0.10-10.el8                    AppStream     466 k
Installing dependencies:
 lm_sensors-libs         x86_64     3.4.0-21.20180522git70f7e08.el8  BaseOS         59 k
 mariadb-connector-c     x86_64     3.0.7-1.el8                      AppStream     148 k
 mariadb-connector-c-config noarch  3.0.7-1.el8                      AppStream      13 k
 net-snmp-agent-libs     x86_64     1:5.8-14.el8_2.1                 AppStream     747 k
 net-snmp-libs           x86_64     1:5.8-14.el8_2.1                 BaseOS        821 k
 perl-Carp               noarch     1.42-396.el8                     BaseOS         30 k
 perl-Exporter           noarch     5.72-396.el8                     BaseOS         34 k
 perl-libs               x86_64     4:5.26.3-416.el8                 BaseOS        1.6 M

Transaction Summary
================================================================================
Install  9 Packages

Total download size: 3.8 M
Installed size: 13 M
Is this ok [y/N]: y
```

Figure 8.222

02. Install kernel-headers and kernel-devel packages.

```
[itit-admin@itit-web01 opt]$ sudo yum install kernel-headers kernel-devel
Last metadata expiration check: 2:01:23 ago on Thu 17 Sep 2020 08:47:51 PM +0530.
Dependencies resolved.
================================================================================
 Package                 Architecture   Version                      Repository     Size
================================================================================
Installing:
 kernel-devel            x86_64         4.18.0-193.14.2.el8_2        BaseOS         15 M
 kernel-headers          x86_64         4.18.0-193.14.2.el8_2        BaseOS        4.0 M
Installing dependencies:
 perl-Errno              x86_64         1.28-416.el8                 BaseOS         76 k
 perl-File-Path          noarch         2.15-2.el8                   BaseOS         38 k
 perl-IO                 x86_64         1.38-416.el8                 BaseOS        141 k
 perl-PathTools          x86_64         3.74-1.el8                   BaseOS         90 k
 perl-Scalar-List-Utils  x86_64         3:1.49-2.el8                 BaseOS         68 k
 perl-Socket             x86_64         4:2.027-3.el8                BaseOS         59 k
 perl-Text-Tabs+Wrap     noarch         2013.0523-395.el8            BaseOS         24 k
 perl-Unicode-Normalize  x86_64         1.25-396.el8                 BaseOS         82 k
 perl-constant           noarch         1.33-396.el8                 BaseOS         25 k
 perl-interpreter        x86_64         4:5.26.3-416.el8             BaseOS        6.3 M
 perl-macros             x86_64         4:5.26.3-416.el8             BaseOS         72 k
 perl-parent             noarch         1:0.237-1.el8                BaseOS         20 k
 perl-threads            x86_64         1:2.21-2.el8                 BaseOS         61 k
 perl-threads-shared     x86_64         1.58-2.el8                   BaseOS         48 k

Transaction Summary
================================================================================
Install  16 Packages

Total download size: 26 M
Installed size: 67 M
Is this ok [y/N]: y_
```

Figure 8.223

03. Edit the keepalived.conf configuration file located at /etc/keepalived directory as follow. The priority 255 is set for the active/master server (itit-web01.itit.com). The virtual IP address is set to 10.10.10.4.

```
  GNU nano 2.9.8                                  /et

! Configuration file for keepalived

vrrp_instance VI_1 {
    state MASTER
    interface ens34
    virtual_router_id 51
    priority 255
    authentication {
        auth_type PASS
        auth_pass admin@itit.com1234566
    }
    virtual_ipaddress {
        10.10.10.4/28 dev ens34
    }
}
```

Figure 8.224

The keepalived.conf file for the passive/slave server (itit-web02.itit.com) is as follows. The priority is set to 254. Make sure that both servers shared the same authentication parameters and virtual IP addresses

```
  GNU nano 2.9.8                                  /et

! Configuration File for keepalived

vrrp_instance VI_1 {
    state BACKUP
    interface ens34
    virtual_router_id 51
    priority 254_
    authentication {
        auth_type PASS
        auth_pass admin@itit.com1234566
    }
    virtual_ipaddress {
        10.10.10.4/28 dev ens34
    }
}
```

Figure 8.225

04. Start and enable the keepalived.service service

```
[itit-admin@itit-web01 keepalived]$ sudo systemctl start keepalived
[itit-admin@itit-web01 keepalived]$ sudo systemctl enable keepalived
Created symlink /etc/systemd/system/multi-user.target.wants/keepalived.service → /usr/lib/systemd/sy
stem/keepalived.service.
[itit-admin@itit-web01 keepalived]$ sudo systemctl status keepalived
● keepalived.service - LVS and VRRP High Availability Monitor
   Loaded: loaded (/usr/lib/systemd/system/keepalived.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2020-09-17 23:38:04 +0530; 11s ago
 Main PID: 19693 (keepalived)
    Tasks: 2 (limit: 4839)
   Memory: 2.0M
   CGroup: /system.slice/keepalived.service
           ├─19693 /usr/sbin/keepalived -D
           └─19694 /usr/sbin/keepalived -D
```

Figure 8.226

05. Check the /var/log/messages log file and observe gratuitious ARP messages being sent to update the MAC address tables of layer 3 devices about the new virtual IP address and associated interfaces.

```
[itit-admin@itit-web01 keepalived]$ sudo tail -f /var/log/messages
Sep 17 23:38:08 itit-web01 Keepalived_vrrp[19694]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:38:08 itit-web01 Keepalived_vrrp[19694]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:38:08 itit-web01 Keepalived_vrrp[19694]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:38:09 itit-web01 systemd[1]: Reloading.
Sep 17 23:38:13 itit-web01 Keepalived_vrrp[19694]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:38:13 itit-web01 Keepalived_vrrp[19694]: (VI_1) Sending/queueing gratuitous ARPs on ens34
for 10.10.10.4
Sep 17 23:38:13 itit-web01 Keepalived_vrrp[19694]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:38:13 itit-web01 Keepalived_vrrp[19694]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:38:13 itit-web01 Keepalived_vrrp[19694]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:38:13 itit-web01 Keepalived_vrrp[19694]: Sending gratuitous ARP on ens34 for 10.10.10.4
```

```
[itit-admin@itit-web02 opt]$ sudo tail -f /var/log/messages
Sep 17 23:40:01 itit-web02 Keepalived_vrrp[45405]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:40:01 itit-web02 Keepalived_vrrp[45405]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:40:01 itit-web02 Keepalived_vrrp[45405]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:40:06 itit-web02 Keepalived_vrrp[45405]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:40:06 itit-web02 Keepalived_vrrp[45405]: (VI_1) Sending/queueing gratuitous ARPs on ens34
for 10.10.10.4
Sep 17 23:40:06 itit-web02 Keepalived_vrrp[45405]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:40:06 itit-web02 Keepalived_vrrp[45405]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:40:06 itit-web02 Keepalived_vrrp[45405]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:40:06 itit-web02 Keepalived_vrrp[45405]: Sending gratuitous ARP on ens34 for 10.10.10.4
Sep 17 23:40:07 itit-web02 systemd[1]: Reloading.
```

Figure 8.226

From the above outputs it is clear that both the hosts are sending gratuitous ARP messages to update devices about their new virtual IP addresses.

06. Check the interface binded with the virtual IP address to confirm if the virtual IP address has been assigned properly.

```
[itit-admin@itit-web01 keepalived]$ ip addr show ens34
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:23:b7:2c brd ff:ff:ff:ff:ff:ff
    inet 10.10.10.2/28 brd 10.10.10.15 scope global ens34
       valid_lft forever preferred_lft forever
    inet 10.10.10.4/28 scope global secondary ens34
       valid_lft forever preferred_lft forever
[itit-admin@itit-web01 keepalived]$
```

Figure 8.226

Both hosts should have the virtual IP address bound to the respective interface.

```
[itit-admin@itit-web02 opt]$ ip addr show ens34
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:c3:97:b4 brd ff:ff:ff:ff:ff:ff
    inet 10.10.10.3/28 brd 10.10.10.15 scope global ens34
       valid_lft forever preferred_lft forever
    inet 10.10.10.4/28 scope global secondary ens34
       valid_lft forever preferred_lft forever
[itit-admin@itit-web02 opt]$
```

Figure 8.227

07. The transfer of VRRP advertisements between the two hosts can be also seen using a tcpdump packet capture.

```
[itit-admin@itit-web01 ~]$ sudo tcpdump -i ens34
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens34, link-type EN10MB (Ethernet), capture size 262144 bytes
09:25:55.571303 IP 10.10.10.3 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, authtype s
imple, intvl 1s, length 20
09:25:56.455626 IP itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype sim
ple, intvl 1s, length 20
09:25:56.574528 IP 10.10.10.3 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, authtype s
imple, intvl 1s, length 20
09:25:57.456360 IP itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype sim
ple, intvl 1s, length 20
09:25:57.576692 IP 10.10.10.3 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, authtype s
imple, intvl 1s, length 20
09:25:58.459291 IP itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype sim
ple, intvl 1s, length 20
09:25:58.577773 IP 10.10.10.3 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, authtype s
imple, intvl 1s, length 20
09:25:59.460792 IP itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype sim
ple, intvl 1s, length 20
09:25:59.578551 IP 10.10.10.3 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, authtype s
imple, intvl 1s, length 20
^C
9 packets captured
9 packets received by filter
0 packets dropped by kernel
```

Figure 8.228

```
[itit-admin@itit-web02 ~]$ sudo tcpdump -i ens34
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens34, link-type EN10MB (Ethernet), capture size 262144 bytes
09:23:20.127769 IP 10.10.10.2 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype s
imple, intvl 1s, length 20
09:23:20.205549 IP itit-web02.itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, a
uthtype simple, intvl 1s, length 20
09:23:21.131229 IP 10.10.10.2 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype s
imple, intvl 1s, length 20
09:23:21.208634 IP itit-web02.itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, a
uthtype simple, intvl 1s, length 20
09:23:22.135233 IP 10.10.10.2 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype s
imple, intvl 1s, length 20
09:23:22.212296 IP itit-web02.itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, a
uthtype simple, intvl 1s, length 20
09:23:23.138451 IP 10.10.10.2 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype s
imple, intvl 1s, length 20
09:23:23.214775 IP itit-web02.itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, a
uthtype simple, intvl 1s, length 20
09:23:24.140419 IP 10.10.10.2 > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 255, authtype s
imple, intvl 1s, length 20
09:23:24.217812 IP itit-web02.itit.com > vrrp.mcast.net: VRRPv2, Advertisement, vrid 51, prio 254, a
uthtype simple, intvl 1s, length 20
^C
10 packets captured
10 packets received by filter
0 packets dropped by kernel
```

Figure 8.229

08. To check which server is providing the website the curl command can be used. The output shows that both the web sites are served via the virtual IP address.

Figure 8.230

```
itit-user@ubuntu:~$ curl -v http://www.itit.com
*   Trying 10.10.10.4:80...
* TCP_NODELAY set
* Connected to www.itit.com (10.10.10.4) port 80 (#0)
> GET / HTTP/1.1
> Host: www.itit.com
> User-Agent: curl/7.68.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Thu, 17 Sep 2020 18:29:18 GMT
< Server: Apache/2.4.37 (centos)
< Last-Modified: Thu, 17 Sep 2020 15:18:10 GMT
< ETag: "a00-5af83e45f32ec"
< Accept-Ranges: bytes
< Content-Length: 2560
< Content-Type: text/html; charset=UTF-8
<
```

```
itit-user@ubuntu:~$ curl -v http://www.itit-lms.com
*   Trying 10.10.10.4:80...
* TCP_NODELAY set
* Connected to www.itit-lms.com (10.10.10.4) port 80 (#0)
> GET / HTTP/1.1
> Host: www.itit-lms.com
> User-Agent: curl/7.68.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Date: Thu, 17 Sep 2020 18:31:09 GMT
< Server: Apache/2.4.37 (centos)
< Last-Modified: Thu, 17 Sep 2020 15:42:17 GMT
< ETag: "a24-5af843a9bc8d4"
< Accept-Ranges: bytes
< Content-Length: 2596
< Content-Type: text/html; charset=UTF-8
<
```

Figure 8.231

09. In case of failure of the active server (itit-web01.itit.com 10.10.10.2) the passive server will step up as the active server and continue providing via the virtual IP address. The end user will not be aware of this change. Below, the interface of the active server is bought down but it can be seen that the web site will continue to work seamlessly at the virtual IP address.

```
[itit-admin@itit-web01 keepalived]$ sudo ifconfig ens34 down
[sudo] password for itit-admin:
[itit-admin@itit-web01 keepalived]$ _
```

Figure 8.231

The web host at 10.10.10.2 (itit-web01.itit.com) is down

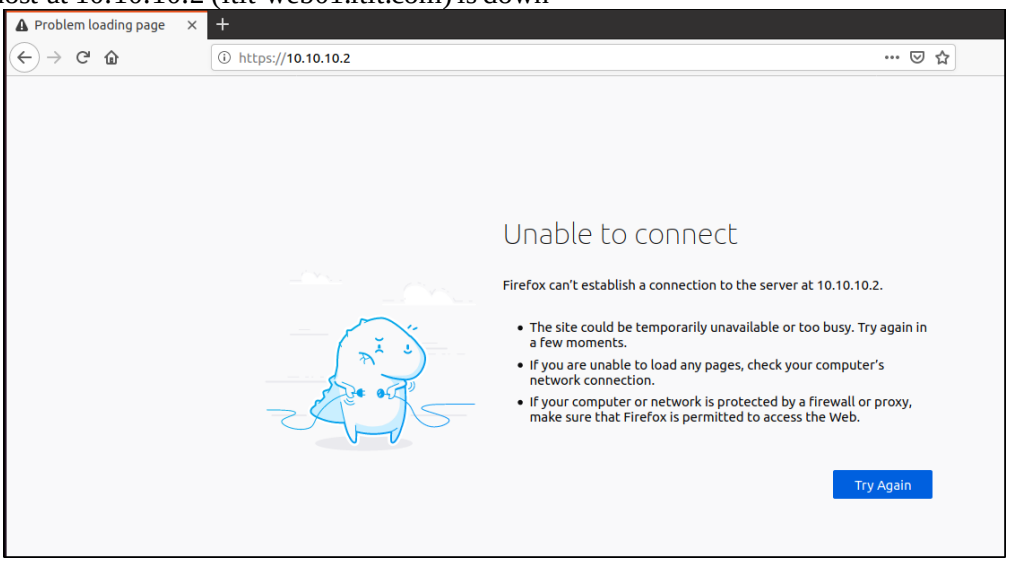


Figure 8.232

The virtual IP address remains up eben though the active server has failed.

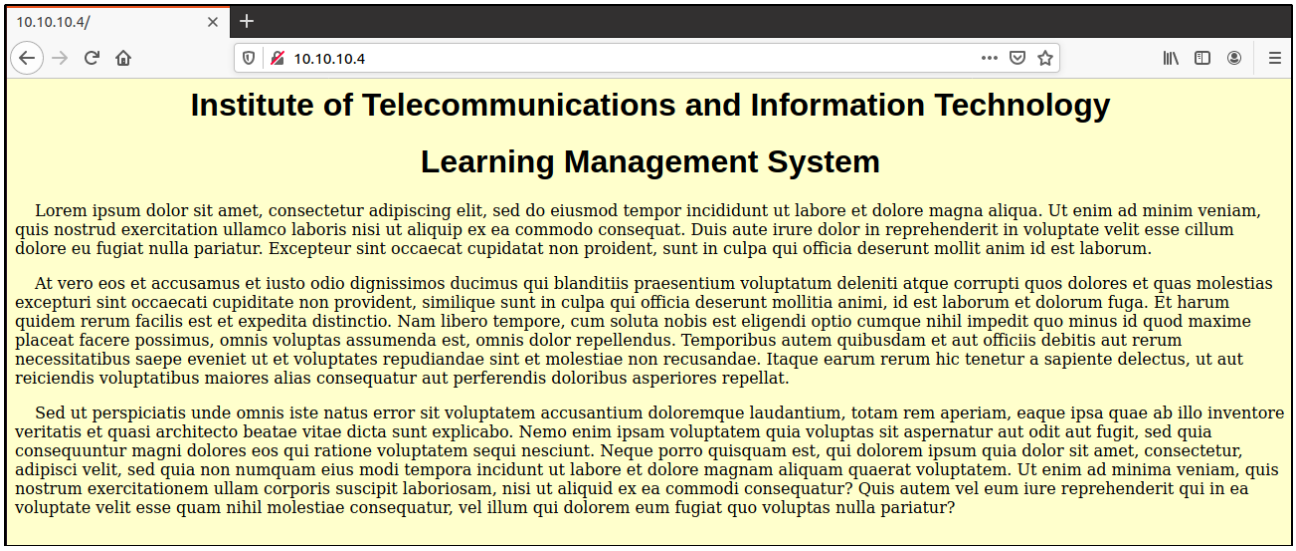


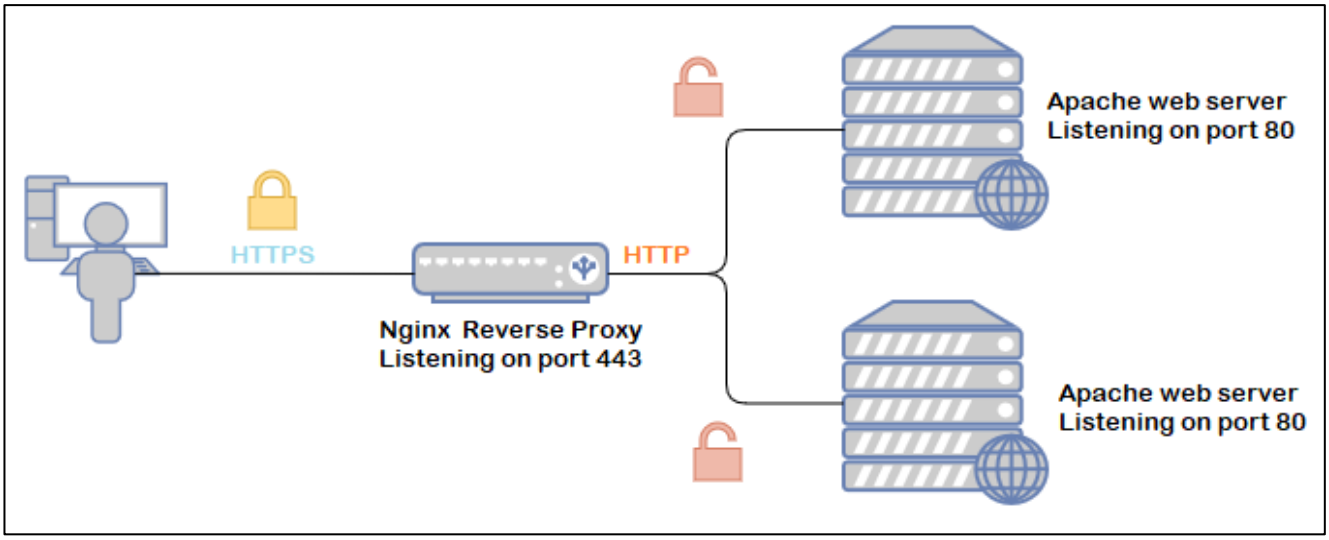
Figure 8.232

Implementing a Highly Available SSL Reverse Proxy

The above configured web servers provide the user with web services using the insecure HTTP protocol. When multiple backend web servers are present in a cluster, reverse proxies can be installed to provide encryption and SSL acceleration between the client and the web server. This is known as SSL offloading. SSL offloading can greatly increase the performance of your secure Web servers, thus increasing customer satisfaction, and can greatly enhance the effectiveness of intrusion detection systems, virus detection systems, etc.

In this implementation, SSL offloading takes the form of SSL termination, the reverse proxy decrypts the SSL traffic from the client and sends it to the backend web servers in an unencrypted state and re-encrypts the data when sending it out to the client. This increases performance at the server level.

Figure 8.233



To provide SSL offloading and reverse proxy services to the back end Apache web cluster an Nginx reverse proxy server cluster with high availability will be implemented. Similar to the backend web servers, the two Nginx reverse proxy servers will be installed with the Keepalived daemon to provide redundancy.

01. Install the epel-release package using the YUM package manager.

```
[itit-admin@itit-revproxy-01 nginx]$ sudo yum install epel-release
Last metadata expiration check: 2:30:46 ago on Sat 19 Sep 2020 09:56:53 AM +0530.
Dependencies resolved.
================================================================================
 Package                Architecture        Version              Repository          Size
================================================================================
Installing:
 epel-release           noarch              8-8.el8              extras              23 k

Transaction Summary
================================================================================
Install  1 Package

Total download size: 23 k
Installed size: 32 k
Is this ok [y/N]: y
```

Figure 8.234

02. Install nginx using the YUM package manager. The epel-release package needs to be installed first before downloading nginx using the YUM package manager.

```
[itit-admin@itit-revproxy-01 ~]$ sudo yum install nginx
Last metadata expiration check: 0:36:05 ago on Sat 19 Sep 2020 01:04:18 AM +0530.
Dependencies resolved.
================================================================================
 Package                  Arch     Version                                Repository   Size
================================================================================
Installing:
 nginx                    x86_64   1:1.14.1-9.module_el8.0.0+184+e34fea82  AppStream   570 k
Installing dependencies:
 gd                       x86_64   2.2.5-6.el8                            AppStream   144 k
 jbigkit-libs             x86_64   2.1-14.el8                             AppStream    55 k
 libXpm                   x86_64   3.5.12-8.el8                           AppStream    58 k
 libjpeg-turbo            x86_64   1.5.3-10.el8                           AppStream   156 k
 libtiff                  x86_64   4.0.9-17.el8                           AppStream   188 k
 libwebp                  x86_64   1.0.0-1.el8                            AppStream   273 k
 nginx-all-modules        noarch   1:1.14.1-9.module_el8.0.0+184+e34fea82  AppStream    23 k
 nginx-filesystem         noarch   1:1.14.1-9.module_el8.0.0+184+e34fea82  AppStream    24 k
 nginx-mod-http-image-filter x86_64 1:1.14.1-9.module_el8.0.0+184+e34fea82 AppStream    35 k
 nginx-mod-http-perl      x86_64   1:1.14.1-9.module_el8.0.0+184+e34fea82  AppStream    45 k
 nginx-mod-http-xslt-filter x86_64 1:1.14.1-9.module_el8.0.0+184+e34fea82  AppStream    33 k
 nginx-mod-mail           x86_64   1:1.14.1-9.module_el8.0.0+184+e34fea82  AppStream    64 k
 nginx-mod-stream         x86_64   1:1.14.1-9.module_el8.0.0+184+e34fea82  AppStream    85 k
 perl-Carp                noarch   1.42-396.el8                           BaseOS       30 k
 perl-Errno               x86_64   1.28-416.el8                           BaseOS       76 k
 perl-Exporter            noarch   5.72-396.el8                           BaseOS       34 k
```

Figure 8.235

03. Install the keepalived package to create the high availability cluster for the reverse proxy.

```
[itit-admin@itit-revproxy-01 ~]$ sudo yum install keepalived
Last metadata expiration check: 2:50:11 ago on Sat 19 Sep 2020 04:27:56 PM +0530.
Dependencies resolved.
================================================================================
 Package                  Arch     Version                                Repository   Size
================================================================================
Installing:
 keepalived               x86_64   2.0.10-10.el8                          AppStream   466 k
Installing dependencies:
 lm_sensors-libs          x86_64   3.4.0-21.20180522git70f7e08.el8        BaseOS       59 k
 mariadb-connector-c      x86_64   3.0.7-1.el8                            AppStream   148 k
 mariadb-connector-c-config noarch 3.0.7-1.el8                            AppStream    13 k
 net-snmp-agent-libs      x86_64   1:5.8-14.el8_2.1                       AppStream   747 k
 net-snmp-libs            x86_64   1:5.8-14.el8_2.1                       BaseOS      821 k

Transaction Summary
================================================================================
Install  6 Packages

Total download size: 2.2 M
Installed size: 7.2 M
Is this ok [y/N]: y_
```

Figure 8.236

04. Install the kernel-headers and kernel-devel packages (dependencies of keepalived)

```
[itit-admin@itit-revproxy-01 ~]$ sudo yum install kernel-headers kernel-devel
Last metadata expiration check: 3:02:48 ago on Sat 19 Sep 2020 04:27:56 PM +0530.
Dependencies resolved.
================================================================================
 Package              Architecture     Version                        Repository      Size
================================================================================
Installing:
 kernel-devel         x86_64           4.18.0-193.19.1.el8_2          BaseOS          15 M
 kernel-headers       x86_64           4.18.0-193.19.1.el8_2          BaseOS         4.0 M

Transaction Summary
================================================================================
Install  2 Packages

Total download size: 19 M
Installed size: 51 M
Is this ok [y/N]: y
```

Figure 8.237

05. Add the below section to the nginx.conf file located at /etc/nginx/nginx.conf. This section shows the configuration to provide reverse proxy services to the www.itit.com site. Create a new block for the www.itit-lms.com site, changing only the *server_name* and *proxy_pass* values.

```
server {
    listen       443;
    ssl          on;
    server_name  www.itit.com itit.com;

    access_log      logs/ssl-access.log;
    error_log       logs/ssl-error.log;

    ssl_certificate         ssl/itit.com.crt;
    ssl_certificate_key     ssl/itit.com.key;

    ssl_protocols           SSLv5 TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers             HIGH:!aNULL:!ECPORT:!CAMELIA:!DES:!MD5:!PSK!:!RC4;
    ssl_prefer_server_ciphers on;

    location / {
            proxy_set_header        x-real-IP               $remote_addr;
            proxy_set_header        x-forwarded-for         $proxy_add_x_forwarded_for;
            proxy_set_header        host                    $host;
            proxy_pass              http://www.itit.com
    }

}
```

Figure 8.238

06. Restart the nginx service suing the *sudo systemctl restart nginx* command.

07. Start and enable the keepalived.service service

```
[itit-admin@itit-revproxy-01 ~]$ sudo systemctl start keepalived
[itit-admin@itit-revproxy-01 ~]$ sudo systemctl enable keepalived
Created symlink /etc/systemd/system/multi-user.target.wants/keepalived.service → /usr/lib/systemd/sy
stem/keepalived.service.
```

Figure 8.239

08. Edit the keepalived.conf file located at /etc/keepalived/ directory in each hosts to define the VRRP configurations to provide high availability. The first image is the configuration of the primary/master reverse proxy configured with a priority of 255 and a virtual IP address of 10.10.10.7

```
! Configuration file for keepalived

vrrp_instance VI_1 {
        state MASTER
        interface ens34
        virtual_router_id 51
        priority 255
        authentication {
                auth_type PASS
                auth_pass admin@itit.com1234566
        }
        virtual_ipaddress {
                10.10.10.7/28 dev ens34
        }
}
```

Figure 8.240

The keepalived configuration of the secondary/slave reverse proxy which has been configured with a priority of 254 and the same virtual IP address, 10.10.10.7.

```
! Configuration file for keepalived

vrrp_instance VI_1 {
        state BACKUP
        interface ens34
        virtual_router_id 51
        priority 254
        authentication {
                auth_type PASS
                auth_pass admin@itit.com1234566
        }
        virtual_ipaddress {
                10.10.10.7/28 dev ens34
        }
}_
```

Figure 8.241

09. Restart the keepalived service on both servers and confirm the communication between them

```
[itit-admin@itit-revproxy-01 ~]$ sudo systemctl restart keepalived
[sudo] password for itit-admin:
[itit-admin@itit-revproxy-01 ~]$ sudo systemctl status keepalived
● keepalived.service - LVS and VRRP High Availability Monitor
   Loaded: loaded (/usr/lib/systemd/system/keepalived.service; enabled; vendor preset: disabled)
   Active: active (running) since Sat 2020-09-19 19:58:44 +0530; 15s ago
  Process: 21087 ExecStart=/usr/sbin/keepalived $KEEPALIVED_OPTIONS (code=exited, status=0/SUCCESS)
 Main PID: 21088 (keepalived)
    Tasks: 2 (limit: 2637)
   Memory: 9.7M
   CGroup: /system.slice/keepalived.service
           ├─21088 /usr/sbin/keepalived -D
           └─21089 /usr/sbin/keepalived -D

Sep 19 19:58:46 itit-revproxy-01.itit.com Keepalived_vrrp[21089]: Sending gratuitous ARP on ens34 f>
Sep 19 19:58:46 itit-revproxy-01.itit.com Keepalived_vrrp[21089]: Sending gratuitous ARP on ens34 f>
Sep 19 19:58:46 itit-revproxy-01.itit.com Keepalived_vrrp[21089]: Sending gratuitous ARP on ens34 f>
Sep 19 19:58:46 itit-revproxy-01.itit.com Keepalived_vrrp[21089]: Sending gratuitous ARP on ens34 f>
```

Figure 8.242

```
[itit-admin@itit-revproxy-02 itit.com]$ sudo systemctl restart keepalived
[itit-admin@itit-revproxy-02 itit.com]$ systemctl status keepalived
● keepalived.service - LVS and VRRP High Availability Monitor
   Loaded: loaded (/usr/lib/systemd/system/keepalived.service; disabled; vendor preset: disabled)
   Active: active (running) since Sat 2020-09-19 20:27:21 +0530; 17s ago
  Process: 17969 ExecStart=/usr/sbin/keepalived $KEEPALIVED_OPTIONS (code=exited, status=0/SUCCESS)
 Main PID: 17970 (keepalived)
    Tasks: 2 (limit: 2637)
   Memory: 9.4M
   CGroup: /system.slice/keepalived.service
           ├─17970 /usr/sbin/keepalived -D
           └─17971 /usr/sbin/keepalived -D

Sep 19 20:27:24 itit-revproxy-02.itit.com Keepalived_vrrp[17971]: Sending gratuitous ARP on ens34 f>
Sep 19 20:27:24 itit-revproxy-02.itit.com Keepalived_vrrp[17971]: Sending gratuitous ARP on ens34 f>
```

Figure 8.243

The virtual IP address has been successfully bound to the interfaces of the two servers.

```
[itit-admin@itit-revproxy-01 ~]$ ip addr show ens34
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel
    link/ether 00:0c:29:be:46:57 brd ff:ff:ff:ff:ff:ff
    inet 10.10.10.5/28 brd 10.10.10.15 scope global ens34
       valid_lft forever preferred_lft forever
    inet 10.10.10.7/28 scope global secondary ens34
       valid_lft forever preferred_lft forever
```

Figure 8.244

```
[itit-admin@itit-revproxy-02 itit.com]$ ip addr show ens34
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel
    link/ether 00:0c:29:4e:0e:3e brd ff:ff:ff:ff:ff:ff
    inet 10.10.10.6/28 brd 10.10.10.15 scope global ens34
       valid_lft forever preferred_lft forever
    inet 10.10.10.7/28 scope global secondary ens34
       valid_lft forever preferred_lft forever
```

Figure 8.245

10. In both reverse proxies set the httpd_can_network_connect Boolean to on. This allows the server to relay the requests back and forth between the client and the backend web servers. Restart the nginx service one done.

```
[itit-admin@itit-revproxy-01 log]$ sudo getsebool -a | grep httpd_can_network_connect
httpd_can_network_connect --> off
httpd_can_network_connect_cobbler --> off
httpd_can_network_connect_db --> off
[itit-admin@itit-revproxy-01 log]$ sudo setsebool httpd_can_network_connect on -P
[itit-admin@itit-revproxy-01 log]$ sudo systemctl restart nginx
```

Figure 8.246

All the DNS records for the websites www.itit.com and www.itit-lms.com should now be pointed to the virtual IP address of the reverse proxy cluster (10.10.10.7). The reverse proxy will then forward the requests and responses to the backend web servers which will be only visible to it.

Figure 8.247

```
itit-user@ubuntu:~$ dig www.itit.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.itit.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48829
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;www.itit.com.                  IN      A

;; ANSWER SECTION:
www.itit.com.           0       IN      A       10.10.10.7
```

```
itit-user@ubuntu:~$ dig www.itit-lms.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.itit-lms.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38155
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;www.itit-lms.com.              IN      A

;; ANSWER SECTION:
www.itit-lms.com.       0       IN      A       10.10.10.7
```

Figure 8.248

The above outputs from the dig commands show that both websites are now pointed towards the virtual reverse proxy address.

11. By running a simple curl command against both websites it was shown that the web server is actually nginx and not Apache. Note that the https protocol is specified.

```
itit-user@ubuntu:~$ curl -Ik https://www.itit.com
HTTP/1.1 200 OK
Server: nginx/1.14.1
Date: Sat, 19 Sep 2020 16:17:50 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 2618
Connection: keep-alive
Last-Modified: Sat, 19 Sep 2020 15:32:31 GMT
ETag: "a3a-5afac536570b4"
Accept-Ranges: bytes

itit-user@ubuntu:~$ curl -Ik https://www.itit-lms.com
HTTP/1.1 200 OK
Server: nginx/1.14.1
Date: Sat, 19 Sep 2020 16:17:59 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 2596
Connection: keep-alive
Last-Modified: Thu, 17 Sep 2020 15:42:17 GMT
ETag: "a24-5af843a9bc8d4"
Accept-Ranges: bytes
```

Figure 8.249

This hiding of server information also provides a level of security and protection to the backend web servers. The reverse proxy also logs information about the users who visit the website via it in the log files specified in the nginx.conf file. In the below log section it is clear that the client IP address and the user agent is logged.

```
10.10.10.10 - - [19/Sep/2020:18:13:19 +0530] "GET / HTTP/1.1" 502 173 "-" "curl/7.68.0"
10.10.10.10 - - [19/Sep/2020:18:17:26 +0530] "GET / HTTP/1.1" 502 173 "-" "Mozilla/5.0 (X11; Ubuntu;
 Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0"
10.10.10.10 - - [19/Sep/2020:18:19:42 +0530] "GET / HTTP/1.1" 502 173 "-" "Mozilla/5.0 (X11; Ubuntu;
 Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0"
10.10.10.10 - - [19/Sep/2020:18:19:42 +0530] "GET /favicon.ico HTTP/1.1" 502 173 "-" "Mozilla/5.0 (X
11; Ubuntu; Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0"
10.10.10.10 - - [19/Sep/2020:18:21:16 +0530] "GET / HTTP/1.1" 502 173 "-" "Mozilla/5.0 (X11; Ubuntu;
 Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0"
10.10.10.10 - - [19/Sep/2020:18:21:17 +0530] "GET / HTTP/1.1" 502 173 "-" "Mozilla/5.0 (X11; Ubuntu;
 Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0"
10.10.10.10 - - [19/Sep/2020:18:35:41 +0530] "GET / HTTP/1.1" 502 173 "-" "Mozilla/5.0 (X11; Ubuntu;
 Linux x86_64; rv:80.0) Gecko/20100101 Firefox/80.0"
```

Figure 8.250

The web browser now displays that the website now utilizes https and that it is a secure connection in both web sites.

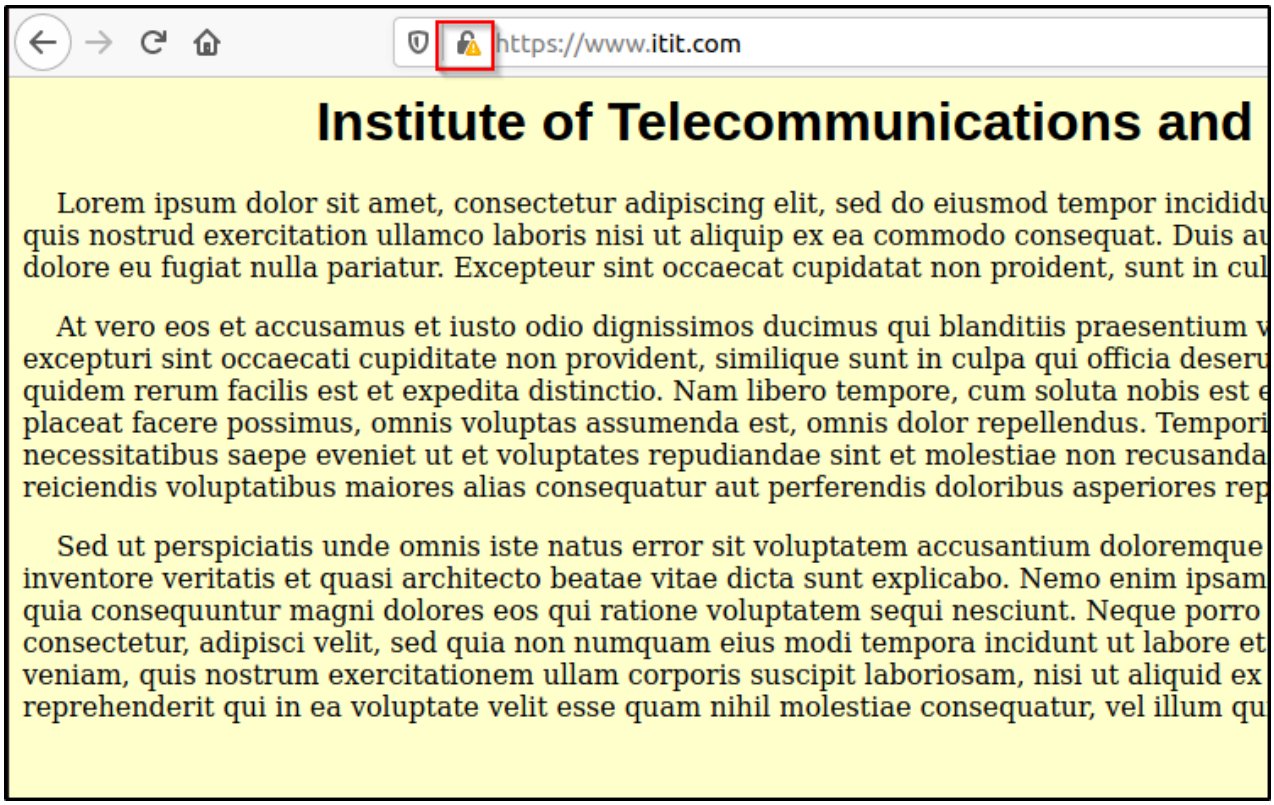


Figure 8.251

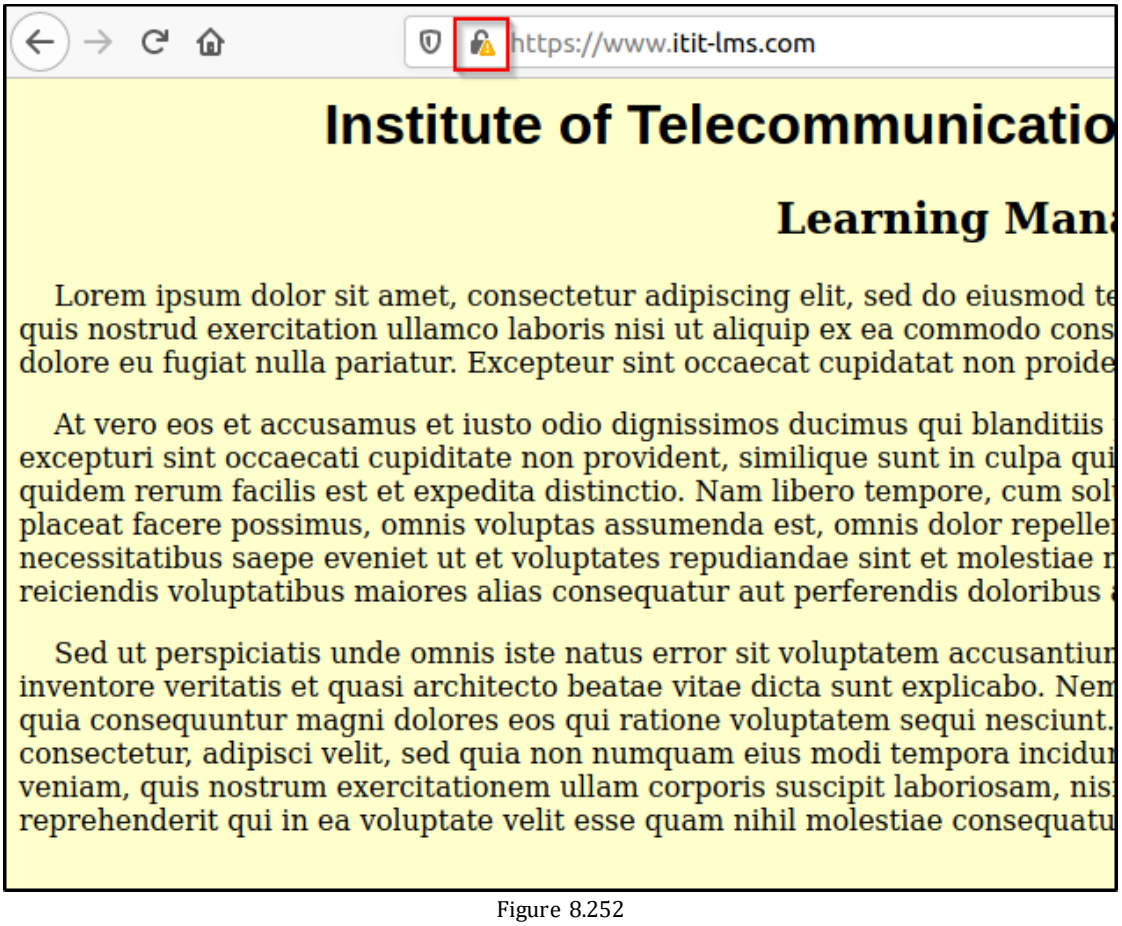


Figure 8.252

Implementing Azure Virtual networks

Step 1 – Go to resource groups to create a resource group for our Azure virtual network & for the VMs.

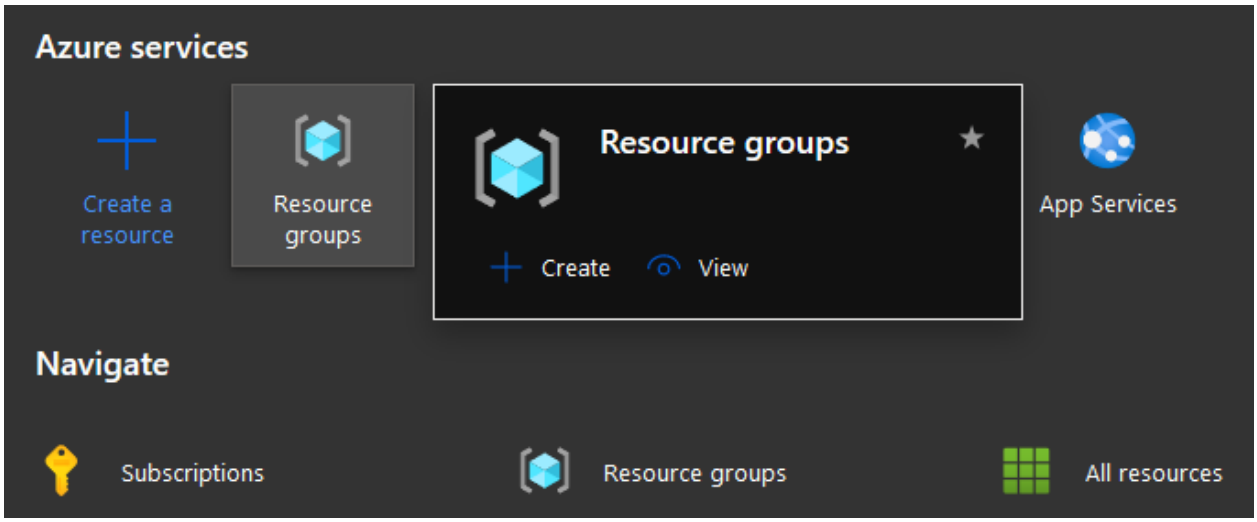


Figure 9.1

Step 2 – Select the subscription we purchased and create a resource group. After that review all the configurations we did and click create.

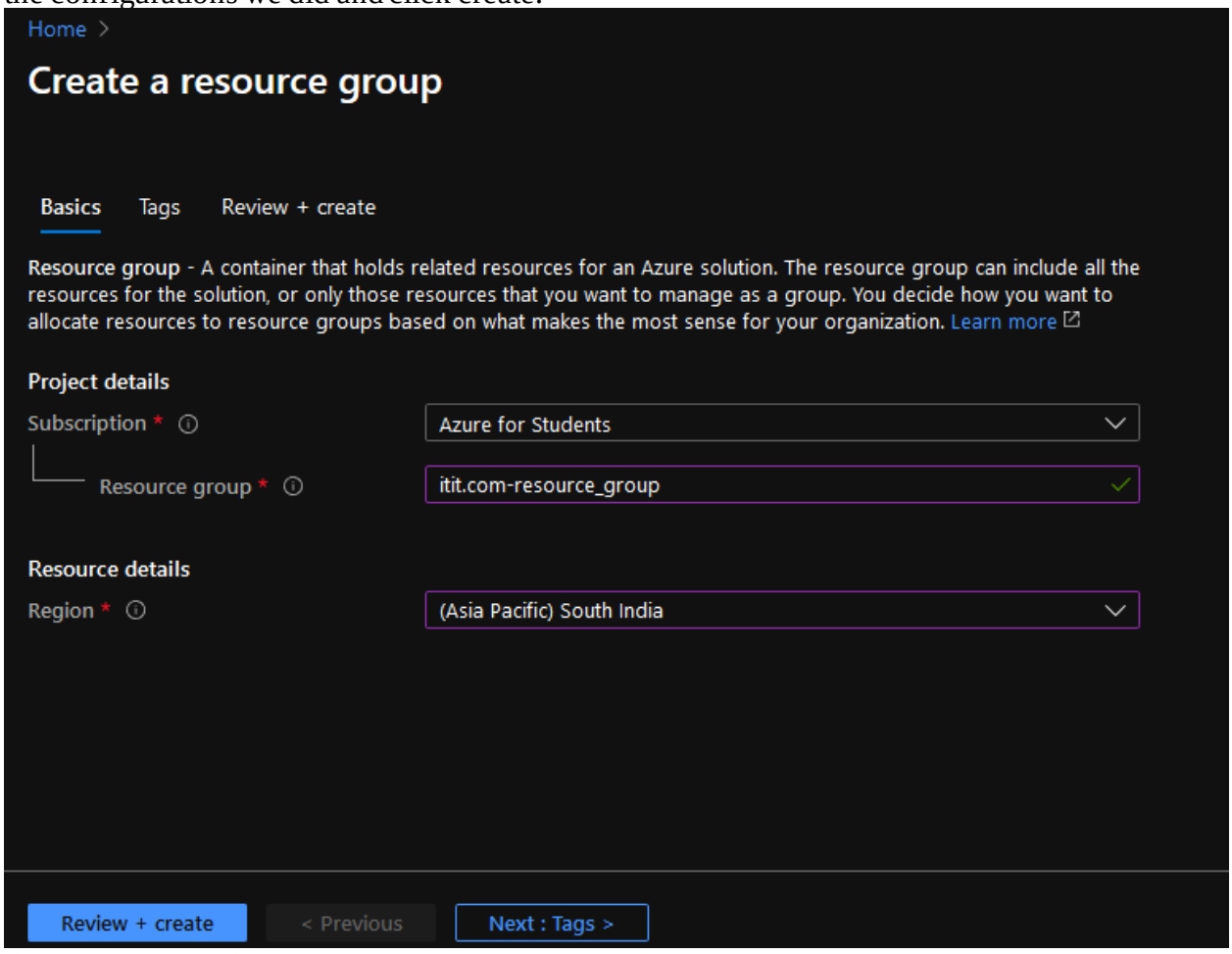


Figure 9.2

Step 3 – Next we will create a virtual network for our virtual machines in Azure to communicate with the on – premises devices.

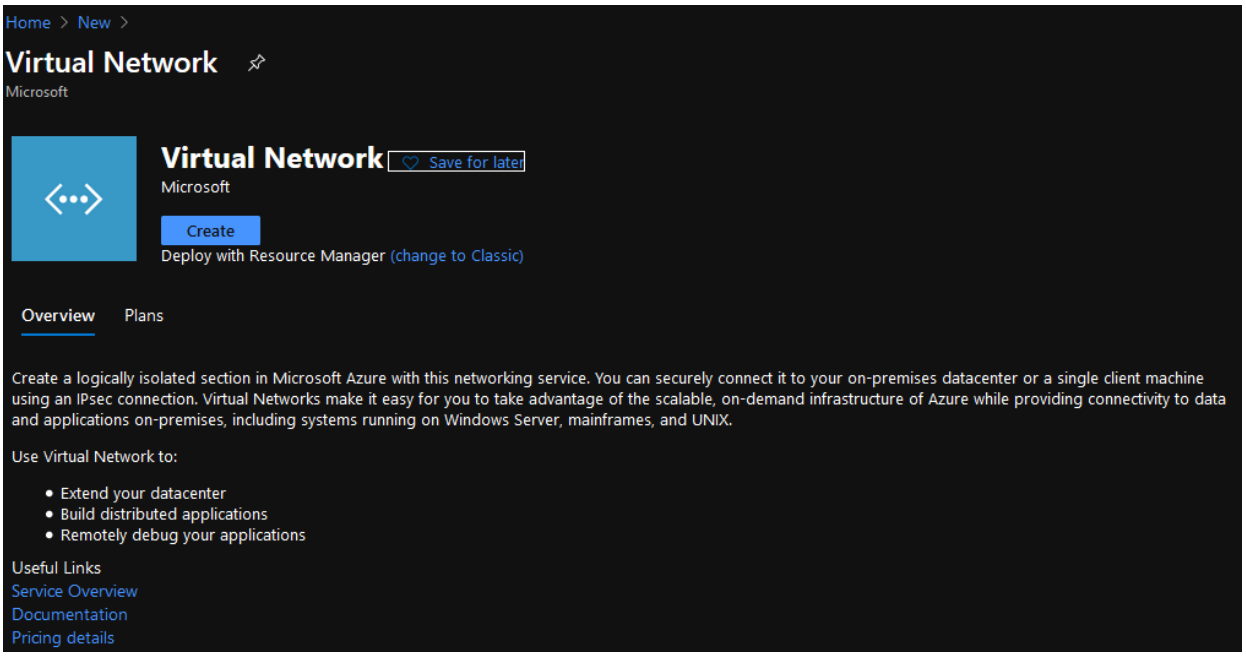


Figure 9.3

Step 4 – Select the subscription & resource group we created previously. Give a name for the instance.

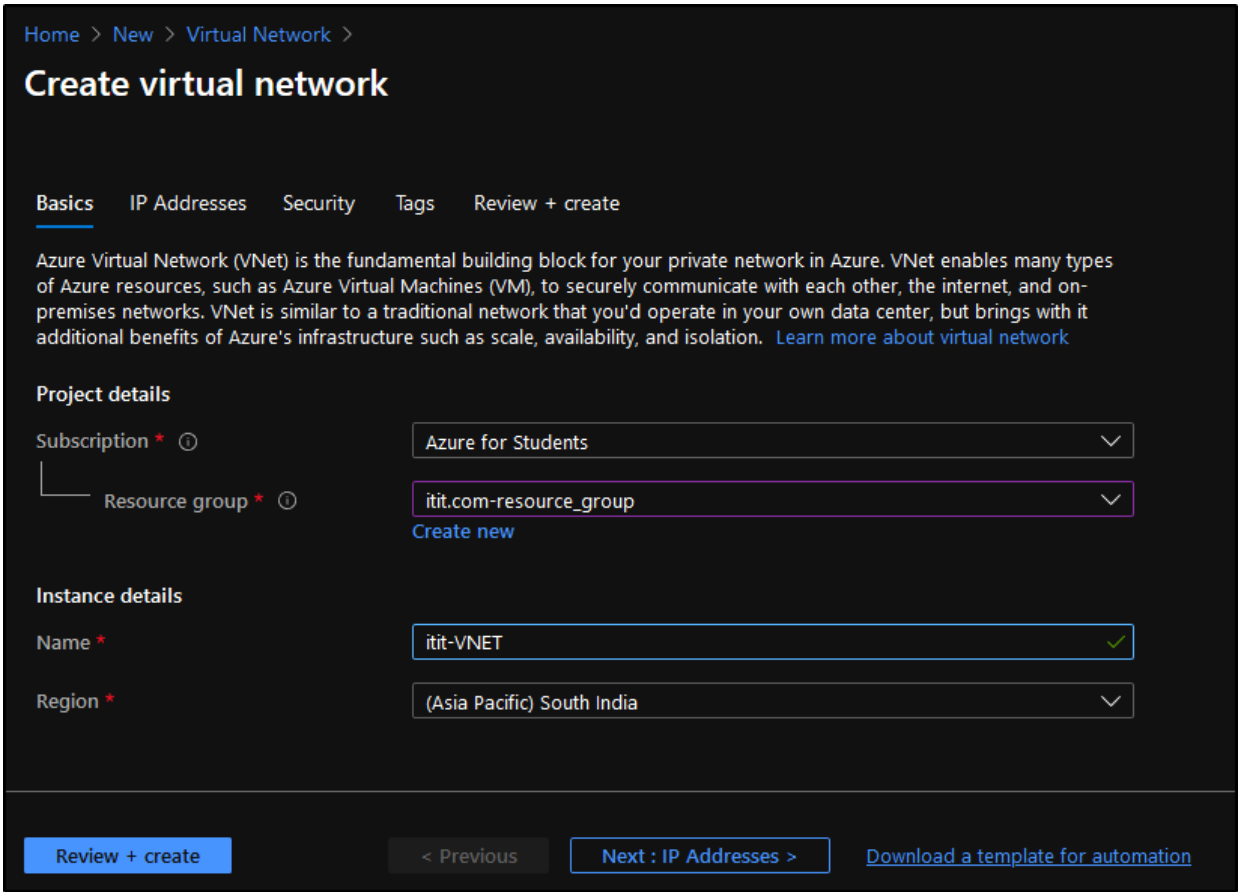


Figure 9.4

Step 5 – In the IP address space give a suitable IP addressing range which won't conflict with the on – premises network. From this IP address range we create more subnets within this virtual network. To create a new subnet click Add subnet and give a name and the range of the subnet we want to create.

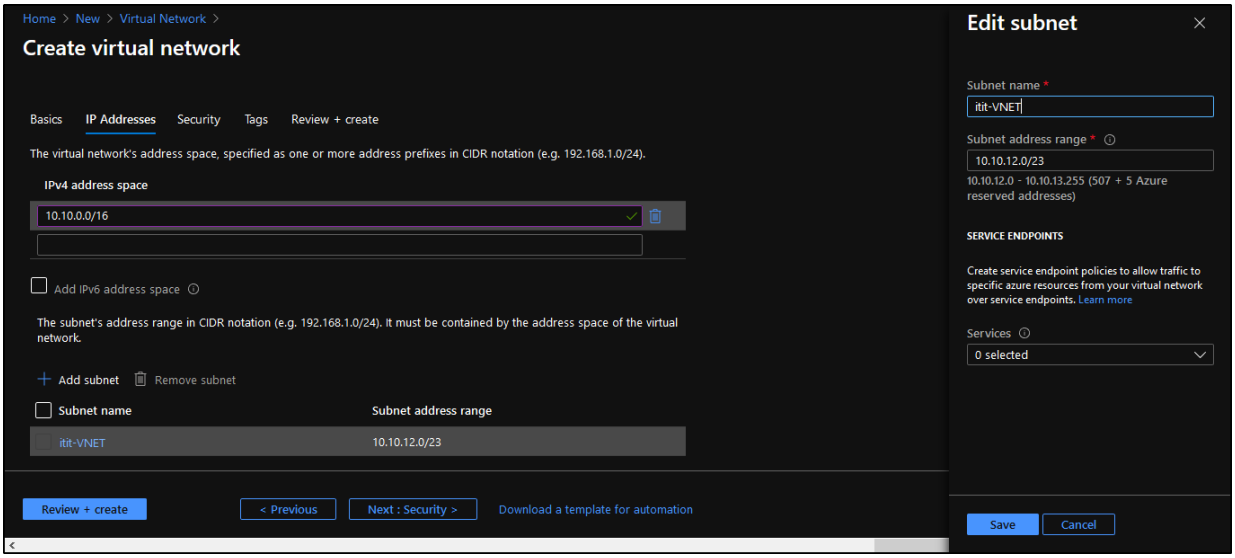


Figure 9.5

Step 6 – Review the configurations done and click create.

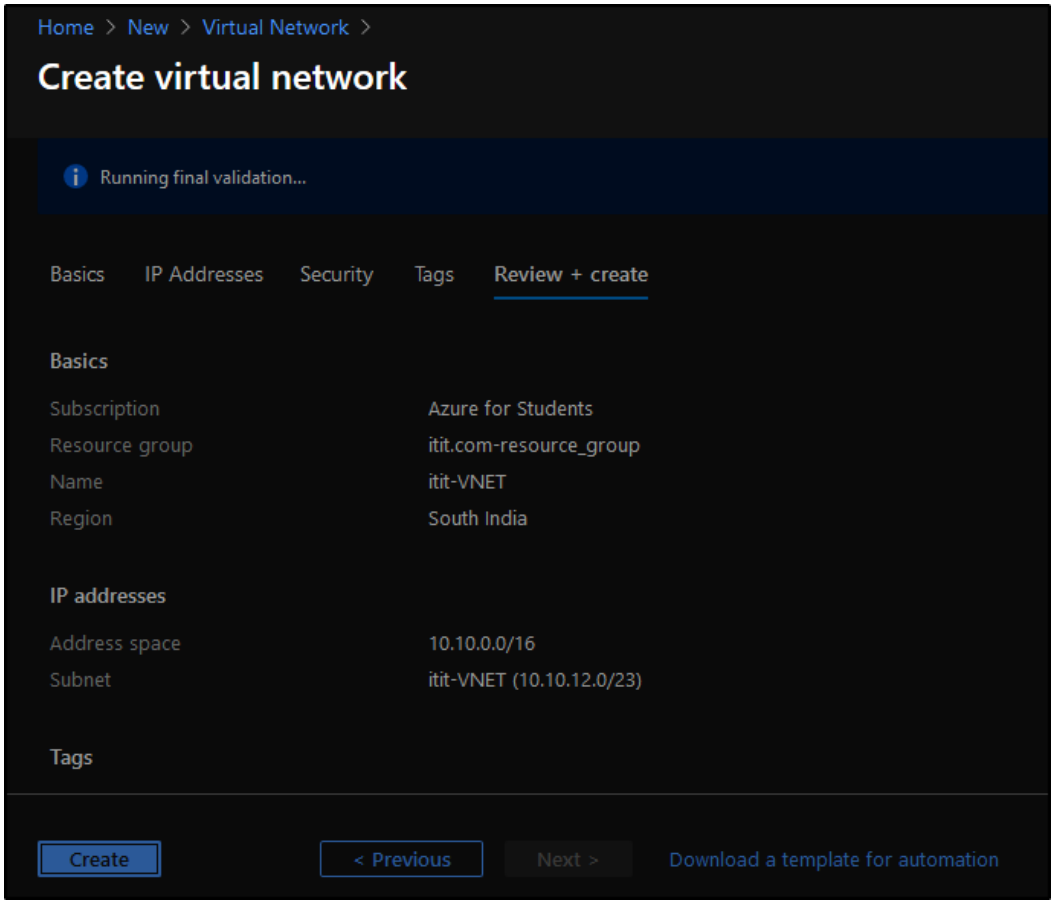


Figure 9.6

Step 7 – For the deployment it might take several minutes depending on the region you select before.

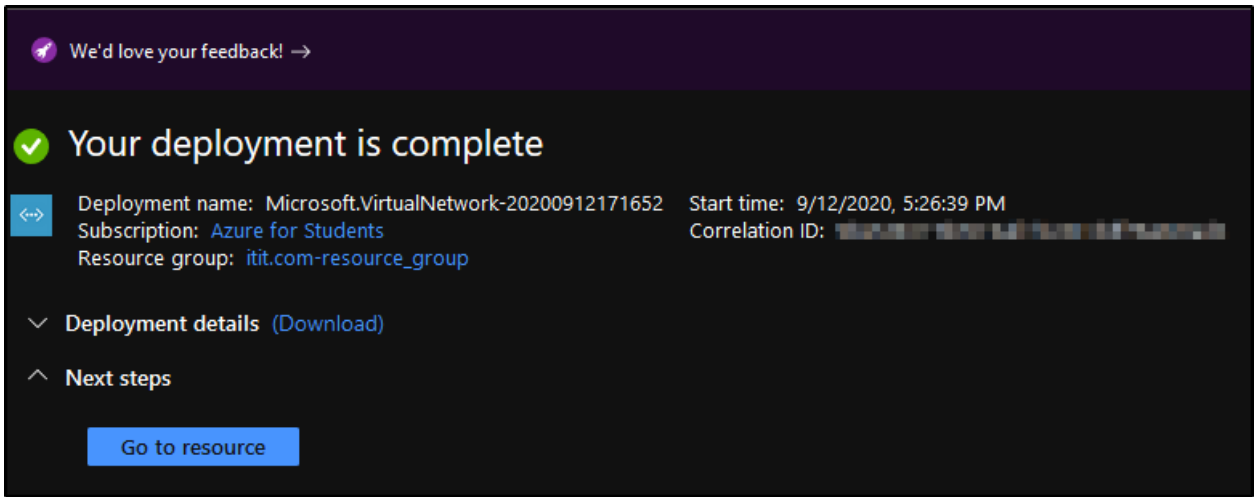


Figure 9.7

Step 8 – Next we will create a virtual network gateway. A virtual network gateway is composed of two or more VMs that are deployed to a specific subnet you create called the gateway subnet. To create this instance give an appropriate name and select the options as given.

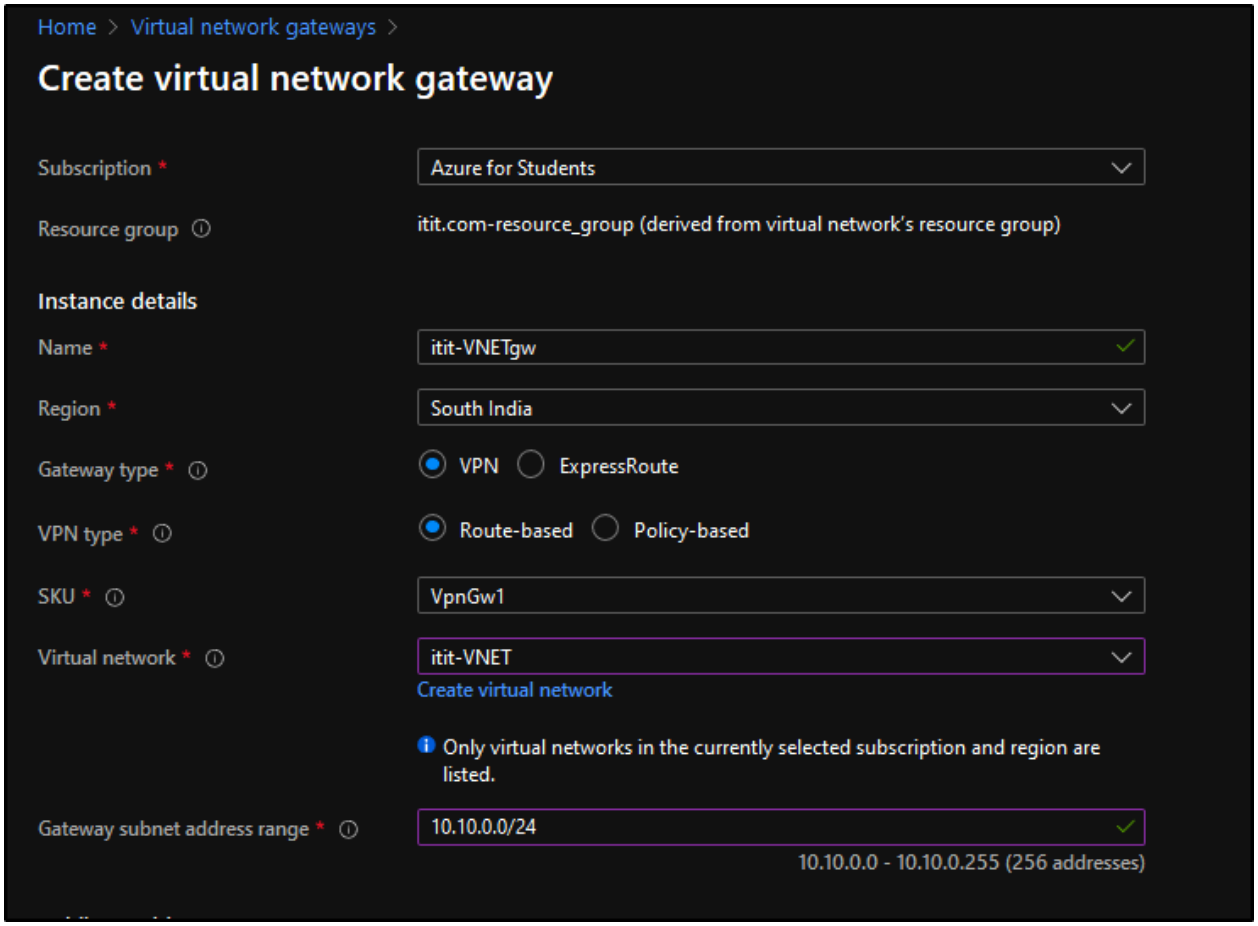


Figure 9.8

Step 9 – Create a new public IP address and give a suitable for the address name.

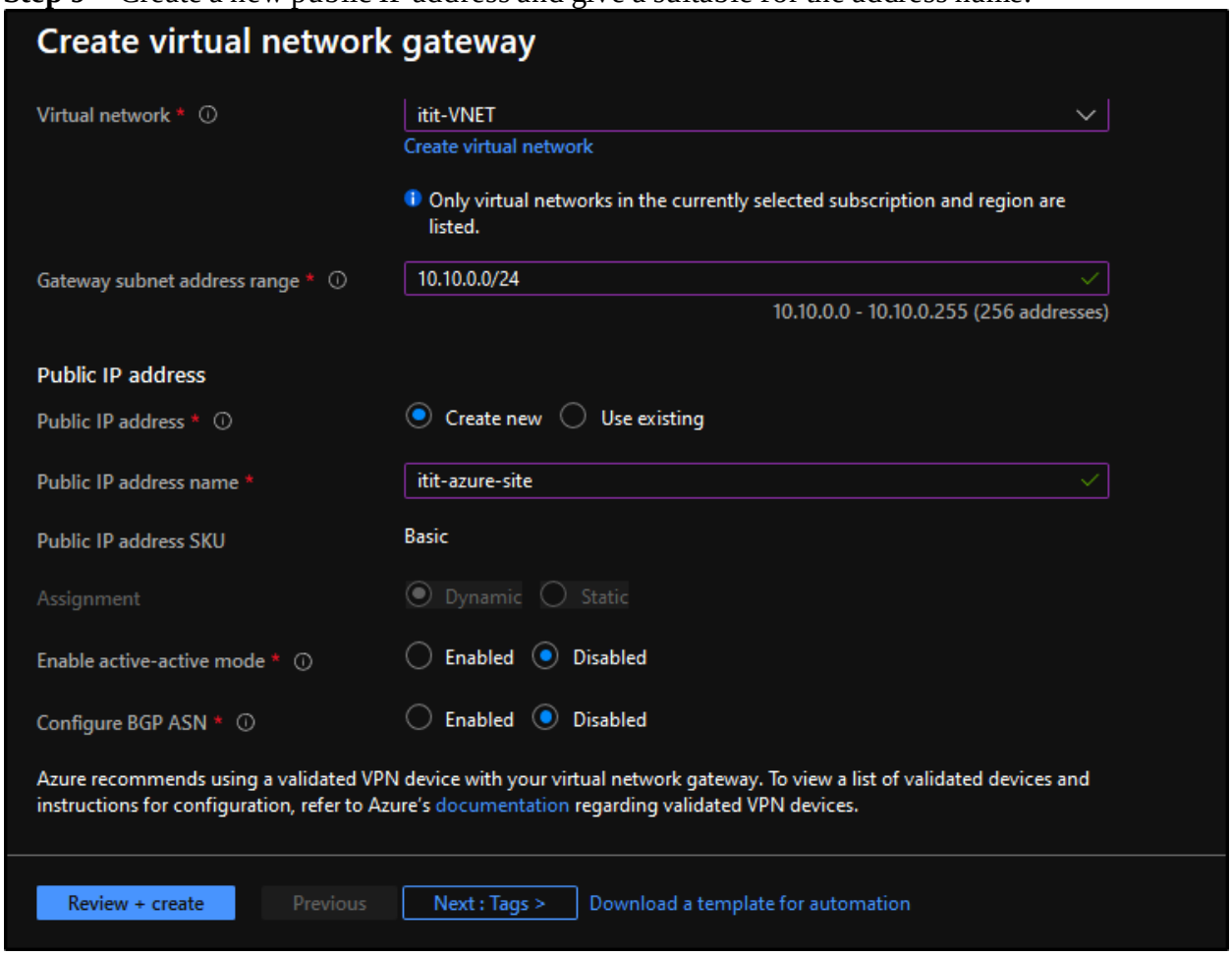


Figure 9.9

Step 10 – After finishing the process of creating a virtual network gateway it will do a validation.

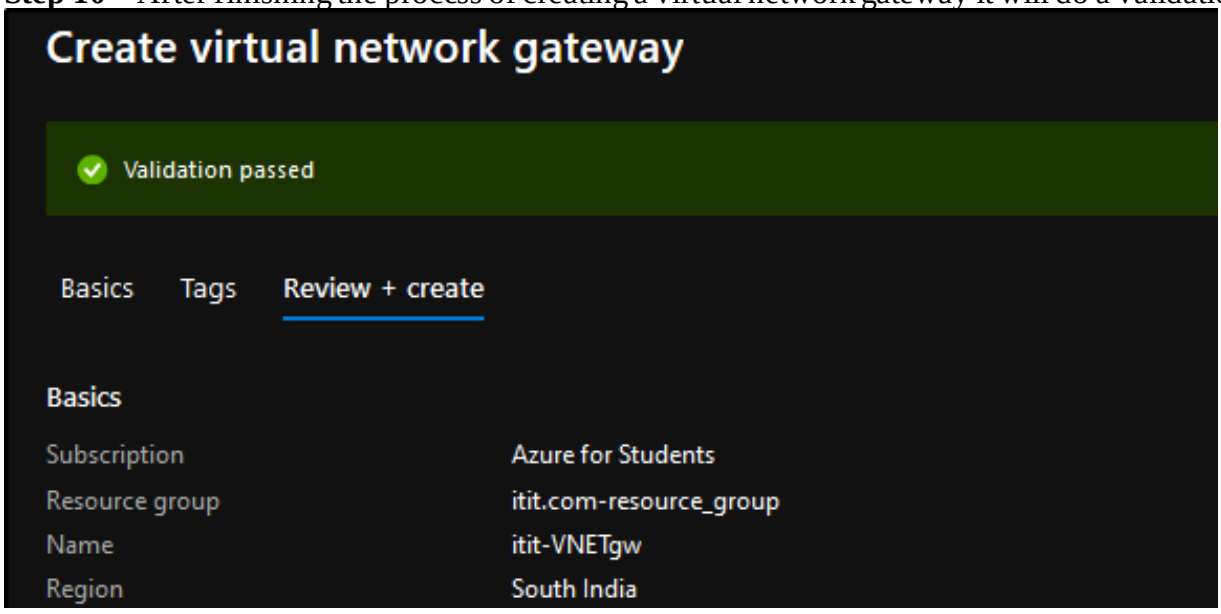


Figure 9.10

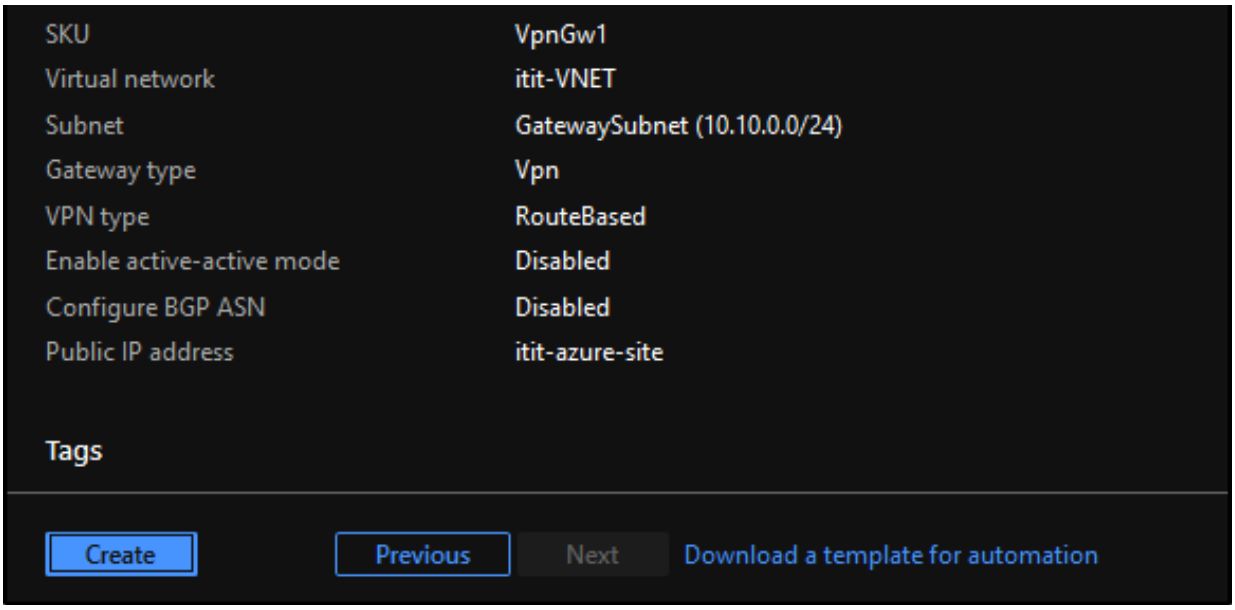


Figure 9.11

Step 11 – Creating a virtual network gateway can take up to 45 minutes to complete.

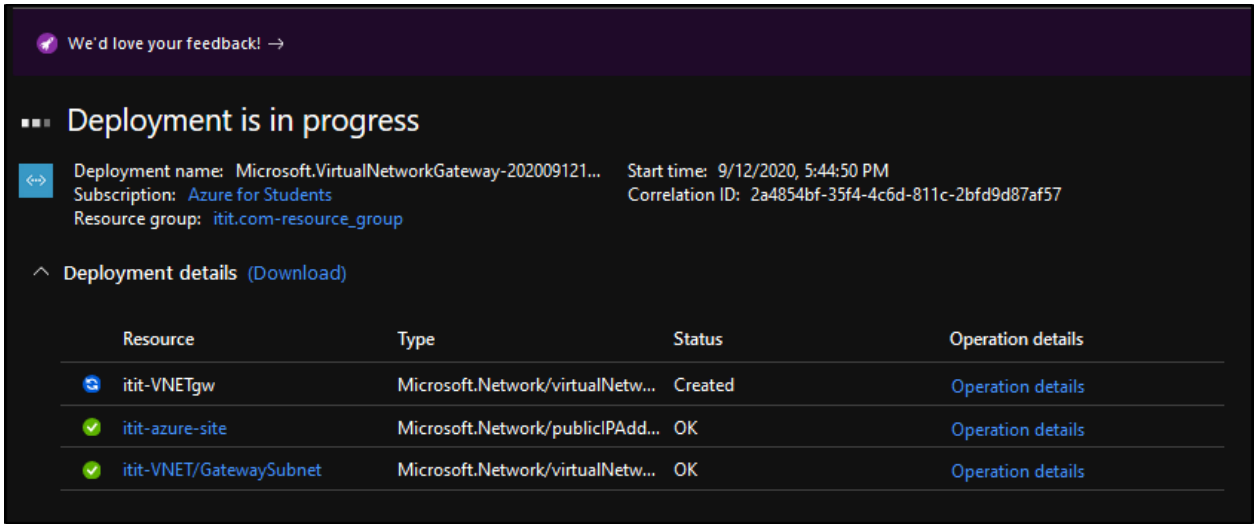


Figure 9.12

Step 12 – After several minute your deployment of the virtual network gateway will be completed.

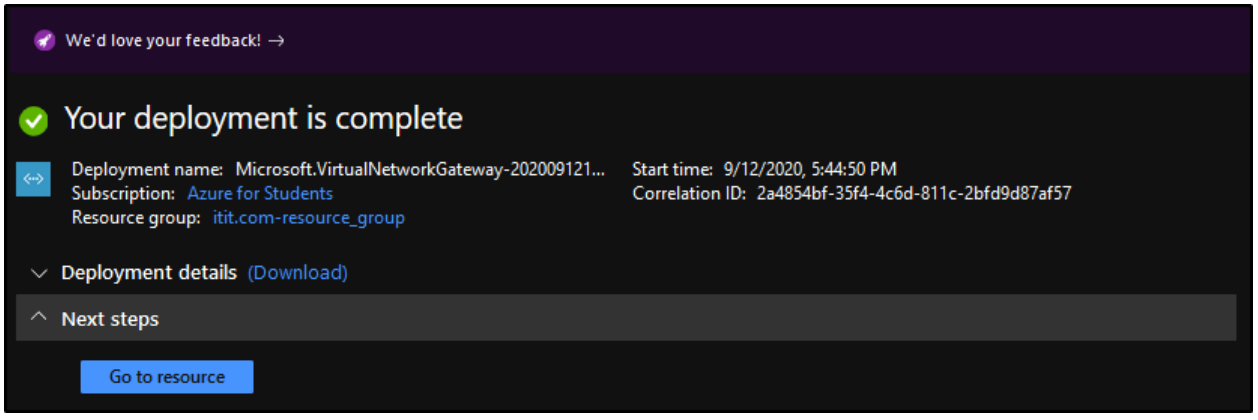


Figure 9.13

Step 13 – After that we will create a local network gateway. A local network gateway represents the hardware or software VPN device in your local network. Use this with a connection to set up a site-to-site VPN connection between an Azure virtual network and you're on – premises network.

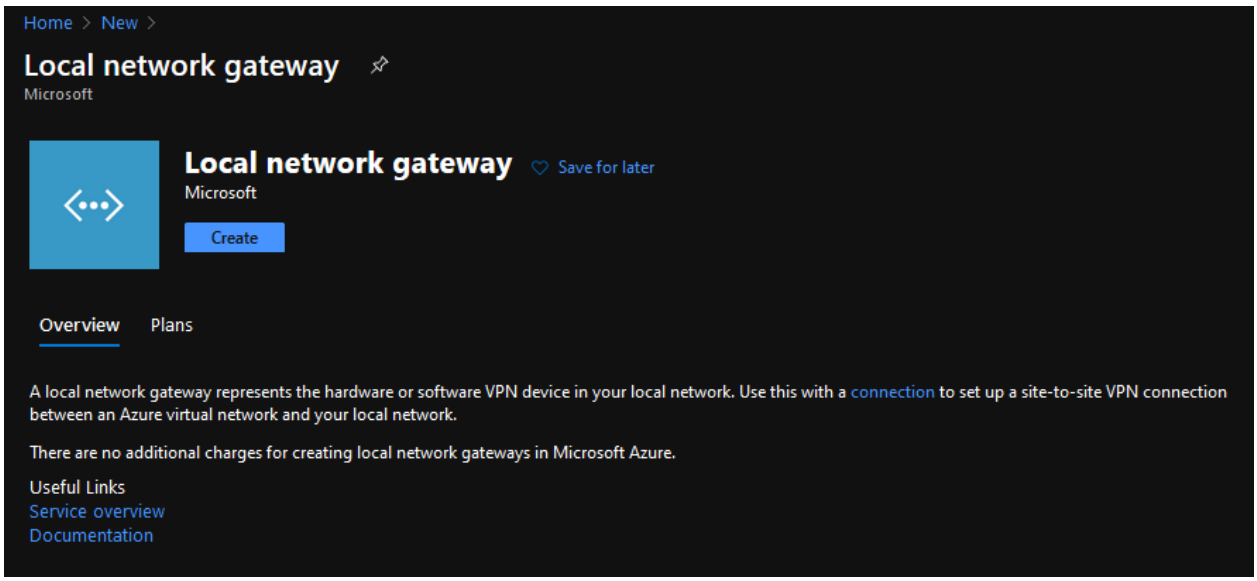


Figure 9.14

Step 14 – To create a local network gateway give an appropriate name and give the public IP addresses of the on – premises network. Add all the on – premises networks in the address space. After doing all the necessary configurations click create.

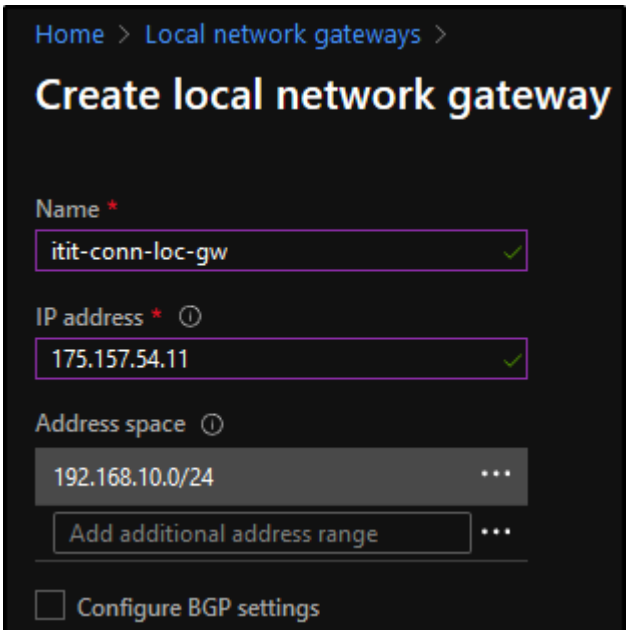


Figure 9.15

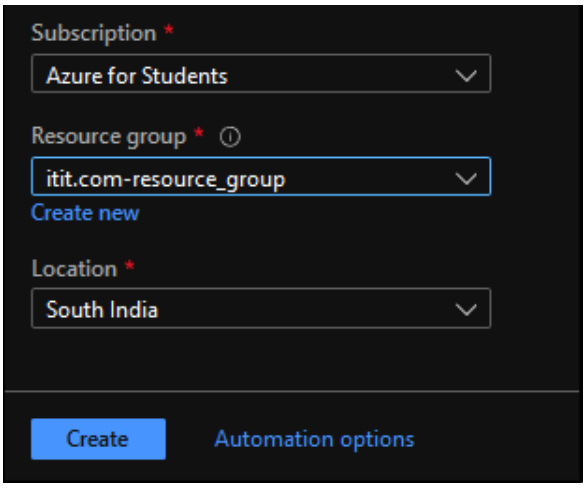


Figure 9.16

Step 15 – After that we will create the VPN connection between the Azure and the on – premises networks. Give a name for the connection, select site – to – site (IPsec) option, give a shared key (This will be used in configuring on – premises VPN too), select IKEv2 then and click OK.

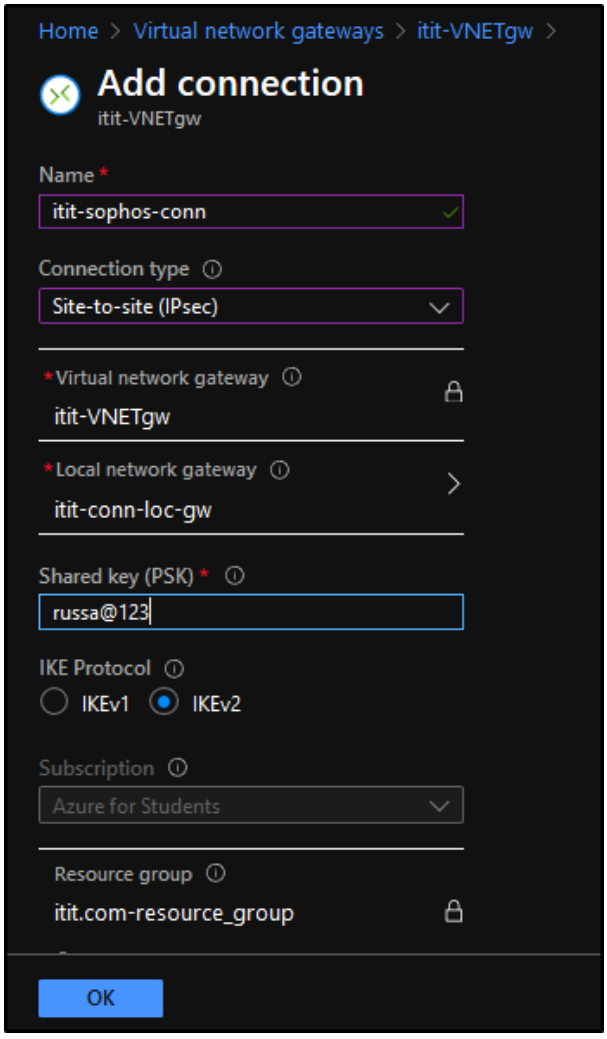


Figure 9.17

Step 16 – After creating the connection you can view it.

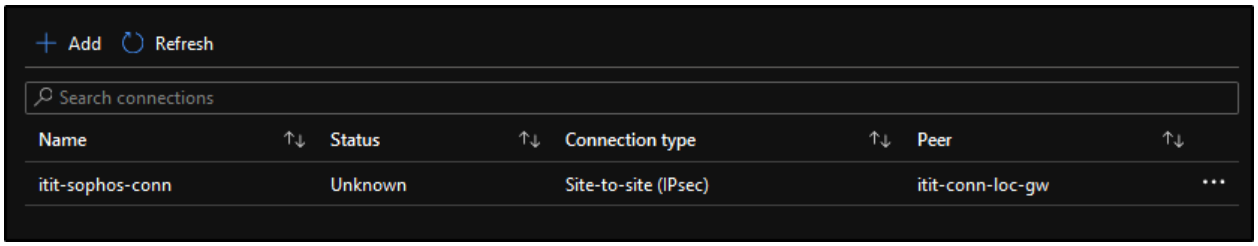


Figure 9.18

Configuring the Azure AD server

Step 1 – After installing the Windows Server change the hostname.

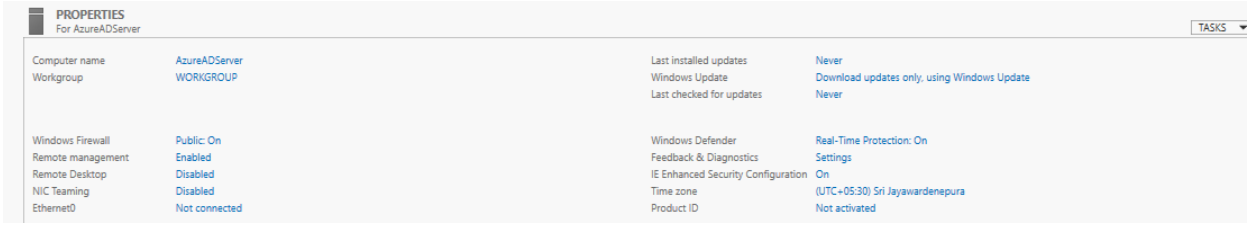


Figure 9.19

Step 2 – We can reach the on – premises network through the IPsec tunnel we created previously.

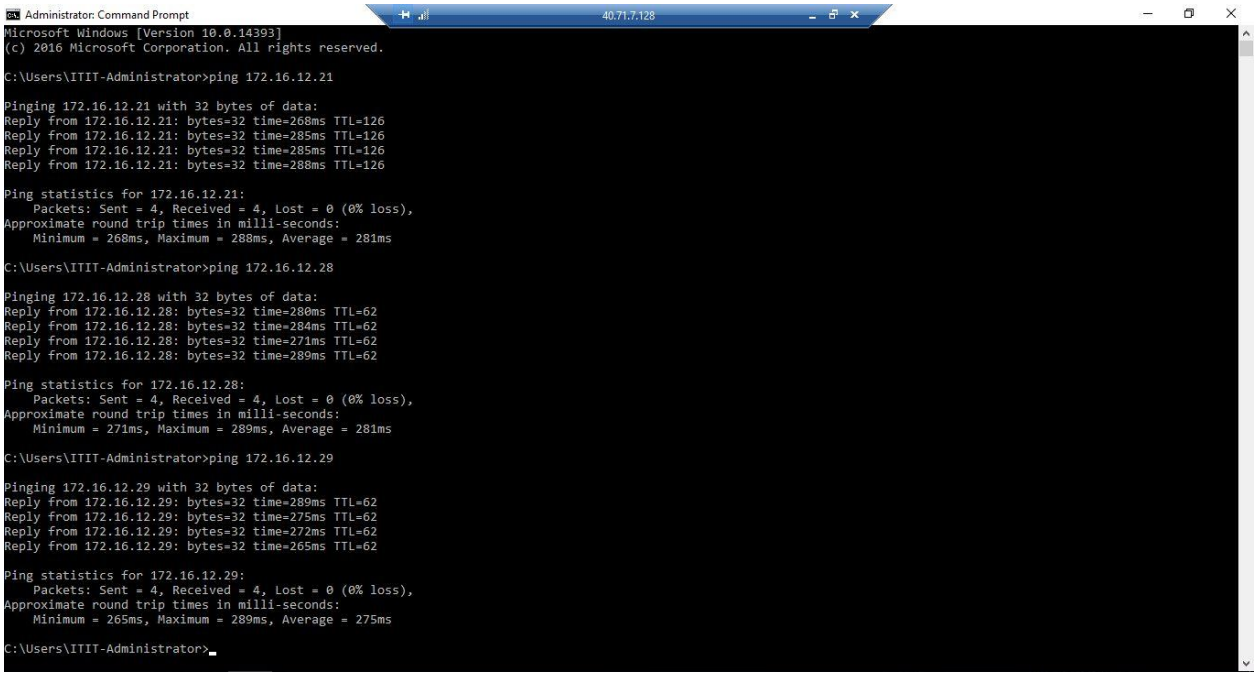


Figure 9.20

Step 3 – Trace route was successful from the Azure AD to the on – premises AD.

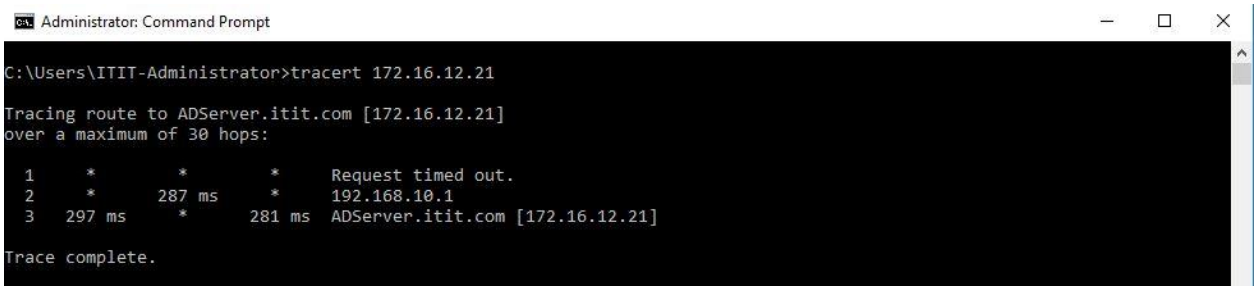


Figure 9.21

Step 4 – Now we will join the computer to the on – premises domain which is itit.com

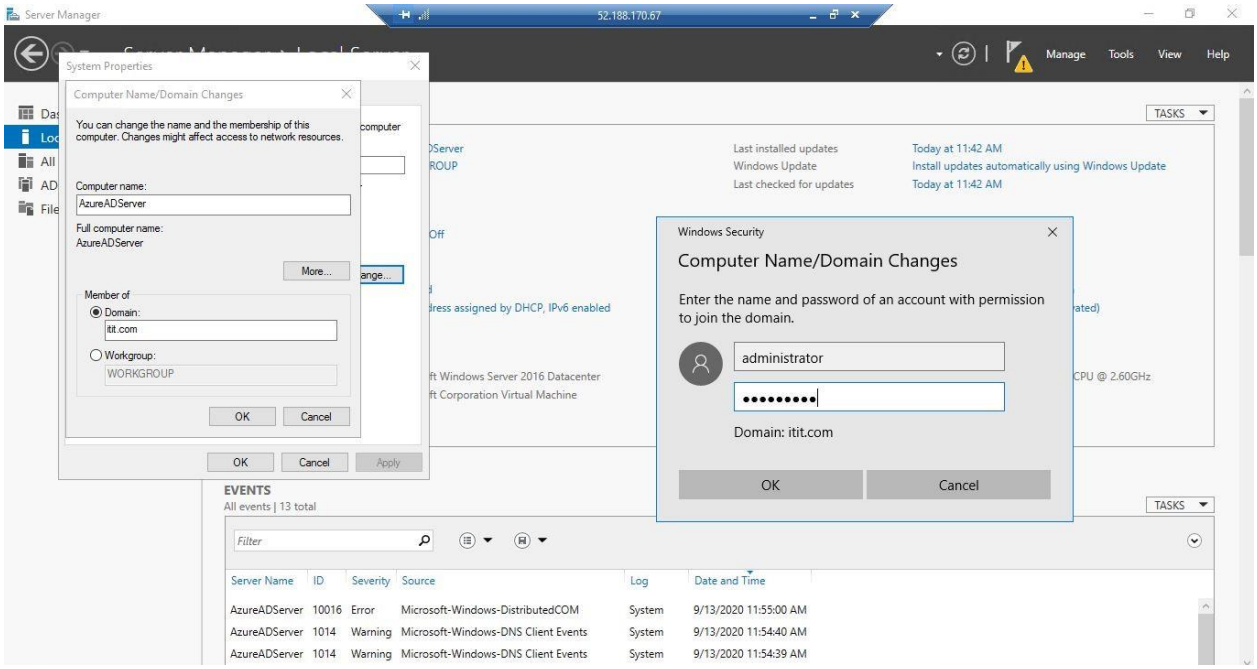


Figure 9.22

Step 5 – Azure AD successfully joined the domain.

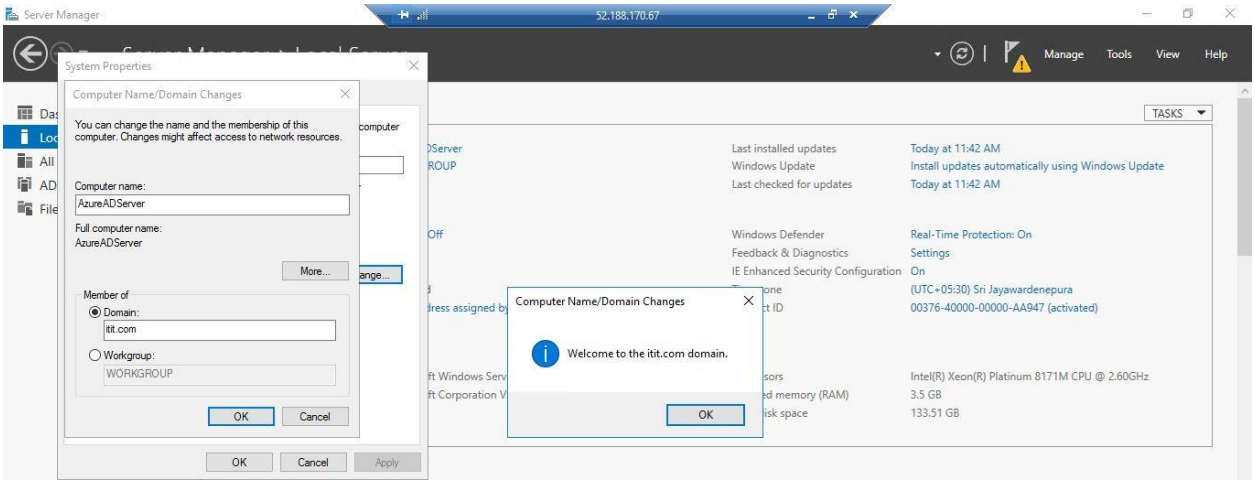


Figure 9.23

Step 6 – In the AD server now we can see the Azure AD Server.

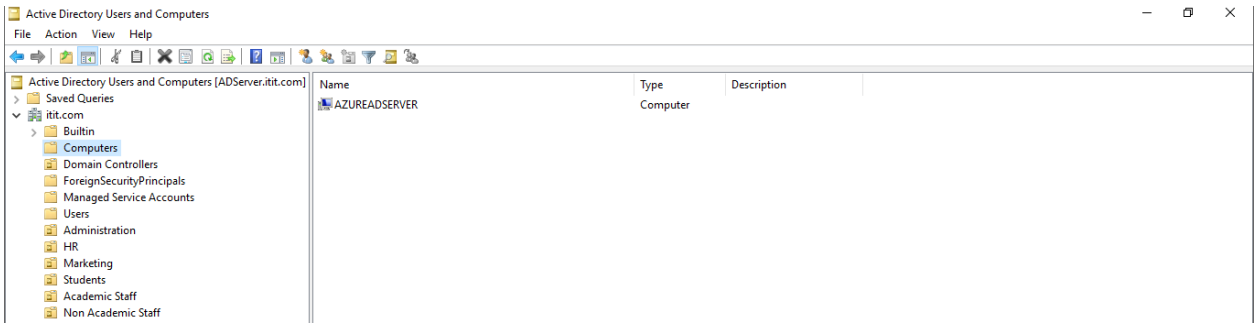


Figure 9.24

Step 7 – Now we will install the domain services in the Azure AD. Select add a domain controller to an existing domain option.

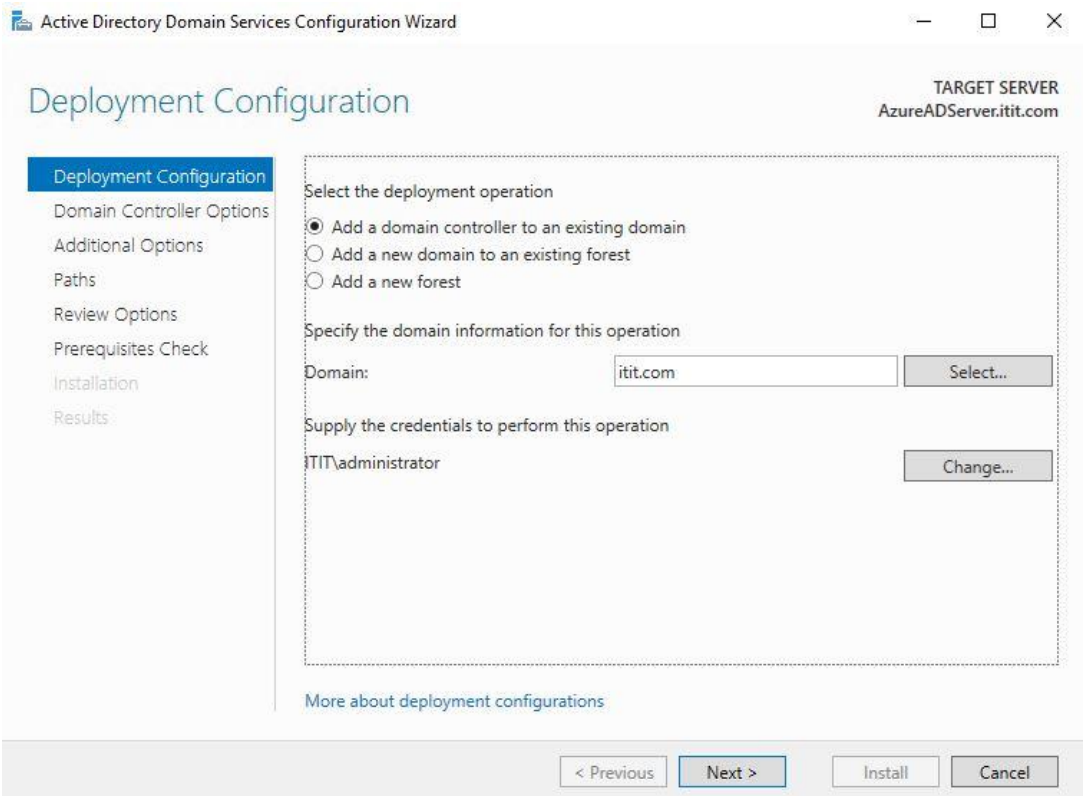


Figure 9.25

Step 8 – Type a suitable password and proceed to the next step.

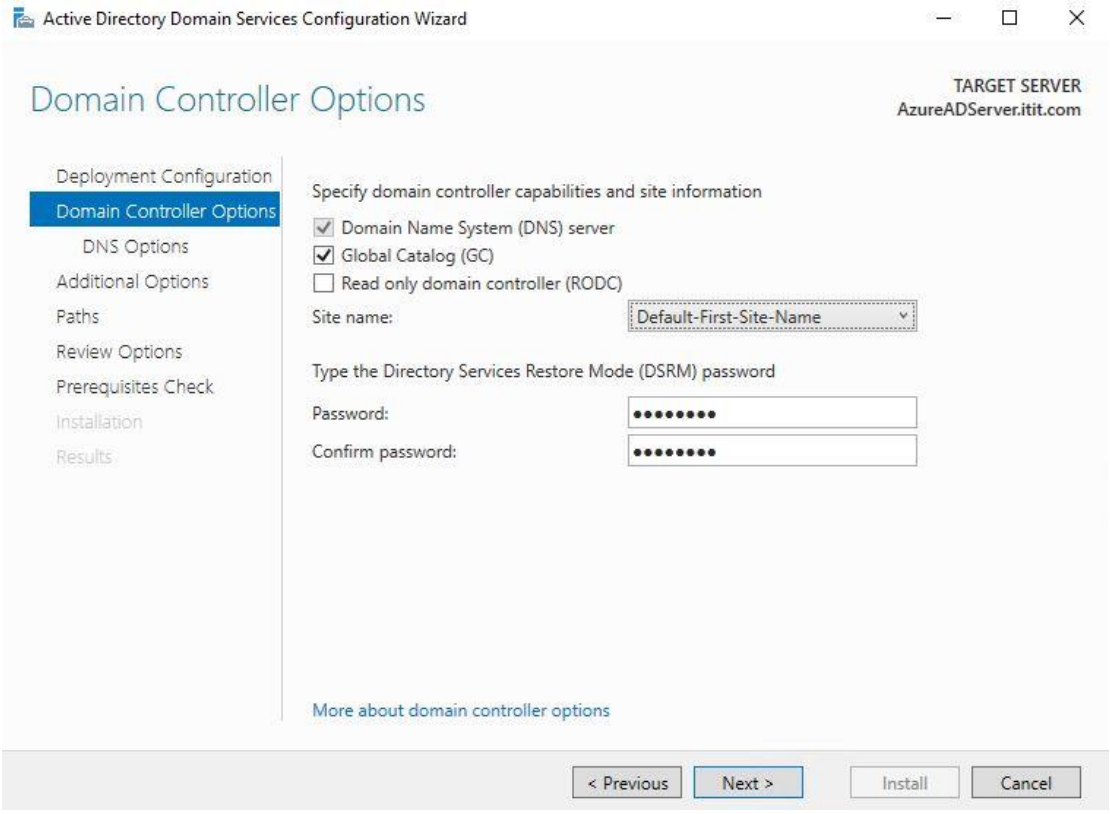


Figure 9.26

Step 9 – Select the on – premises AD server and click next.

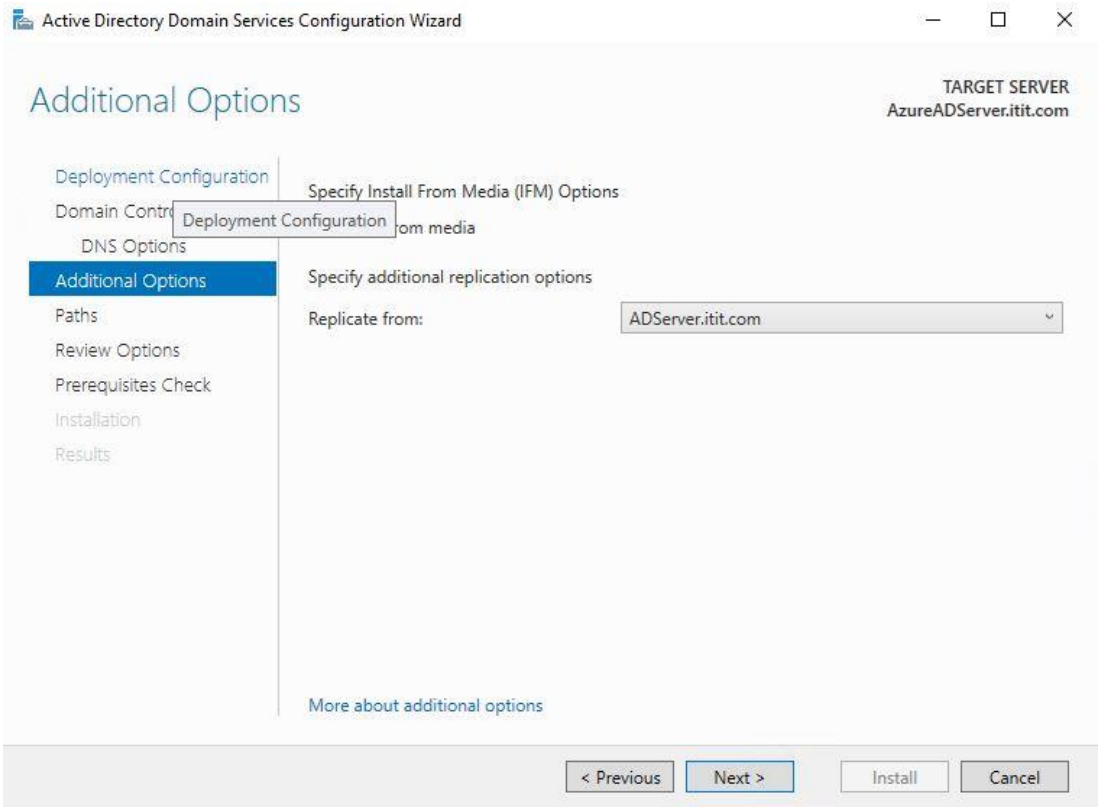


Figure 9.27

Step 10 – Review the configurations done and proceed to the next step.

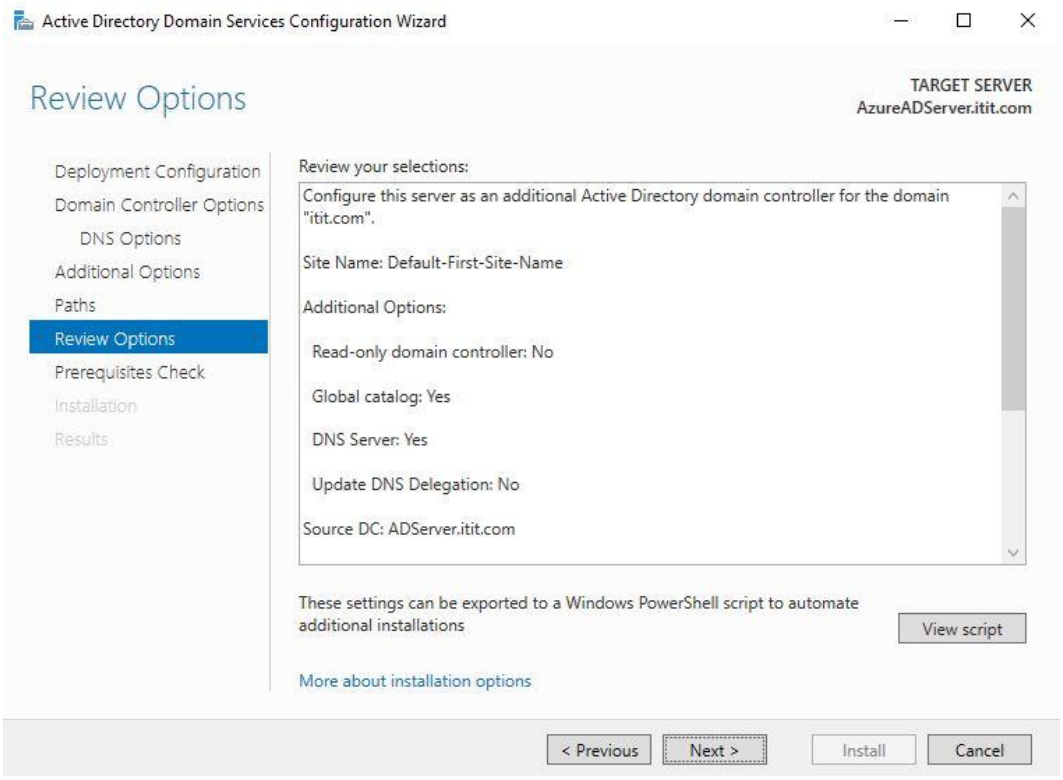


Figure 9.28

Step 11 – Click install.

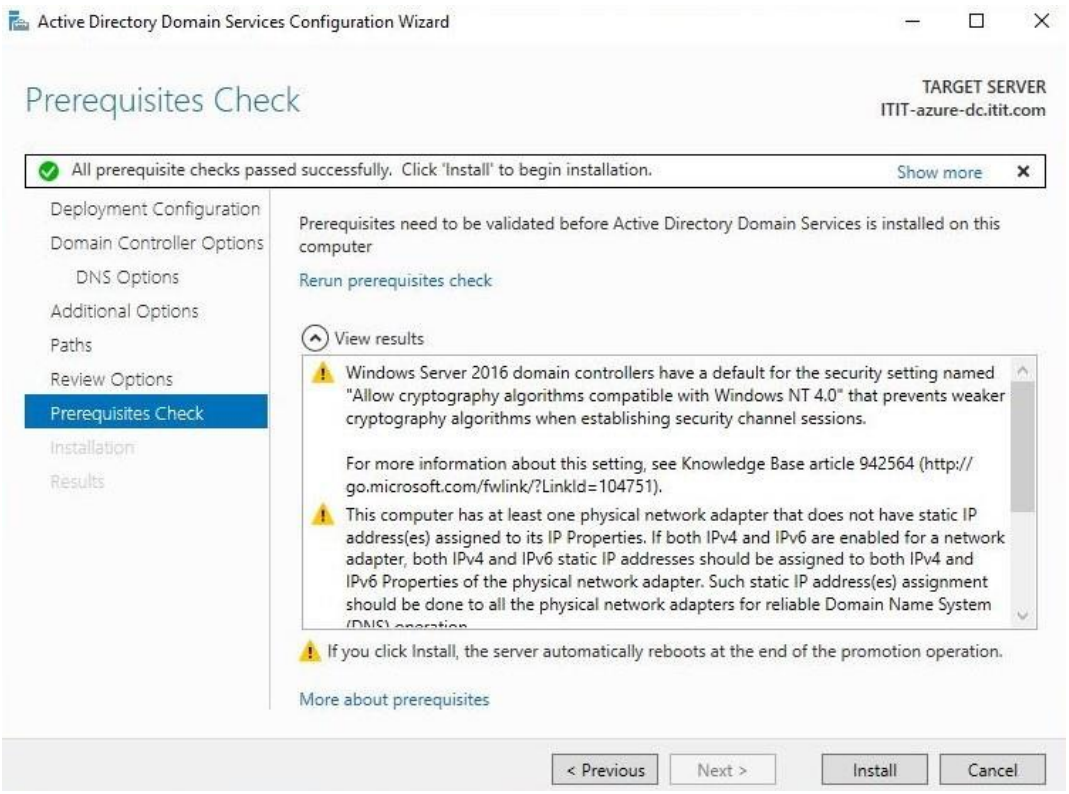


Figure 9.29

Step 12 – After a successful installation it will prompt to restart the computer.

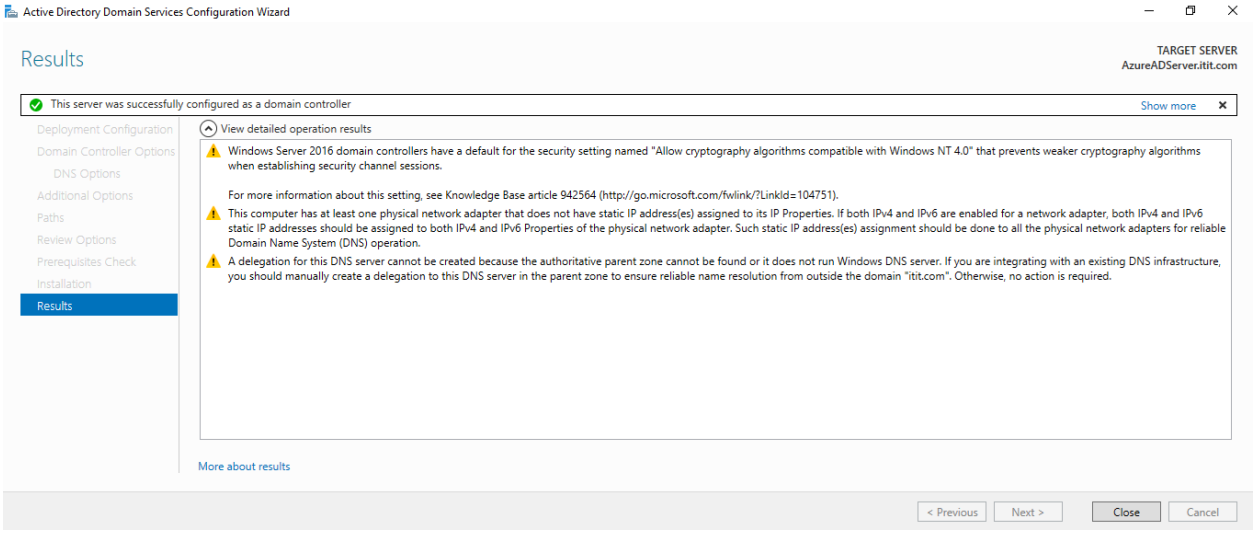


Figure 9.31

Step 13 – After joining the Azure AD to the on – premises domain it will successfully migrate all configurations to the Azure AD.

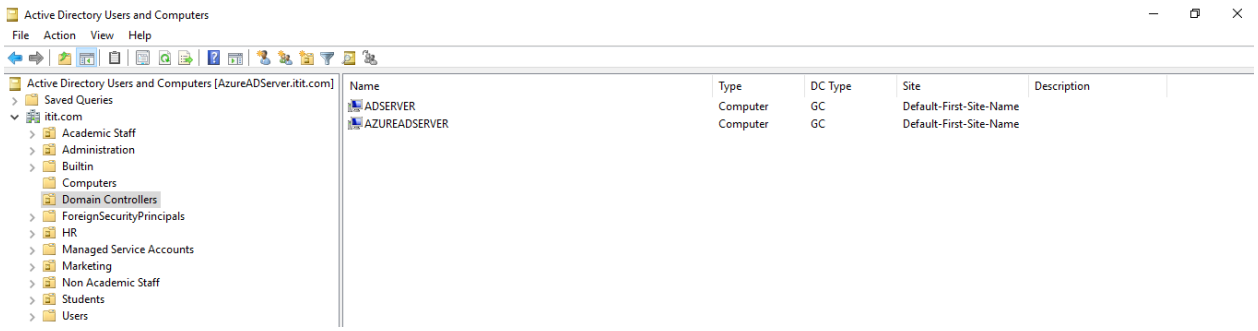


Figure 9.32

Step 14 – Now we create two users, one in the on – premises AD server and the other in the Azure AD server.

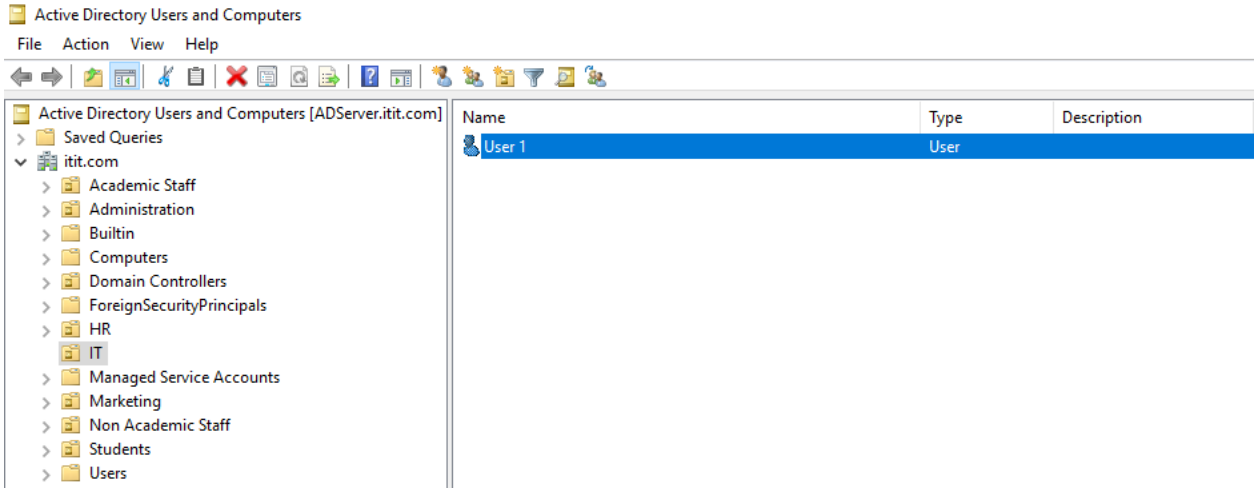


Figure 9.33

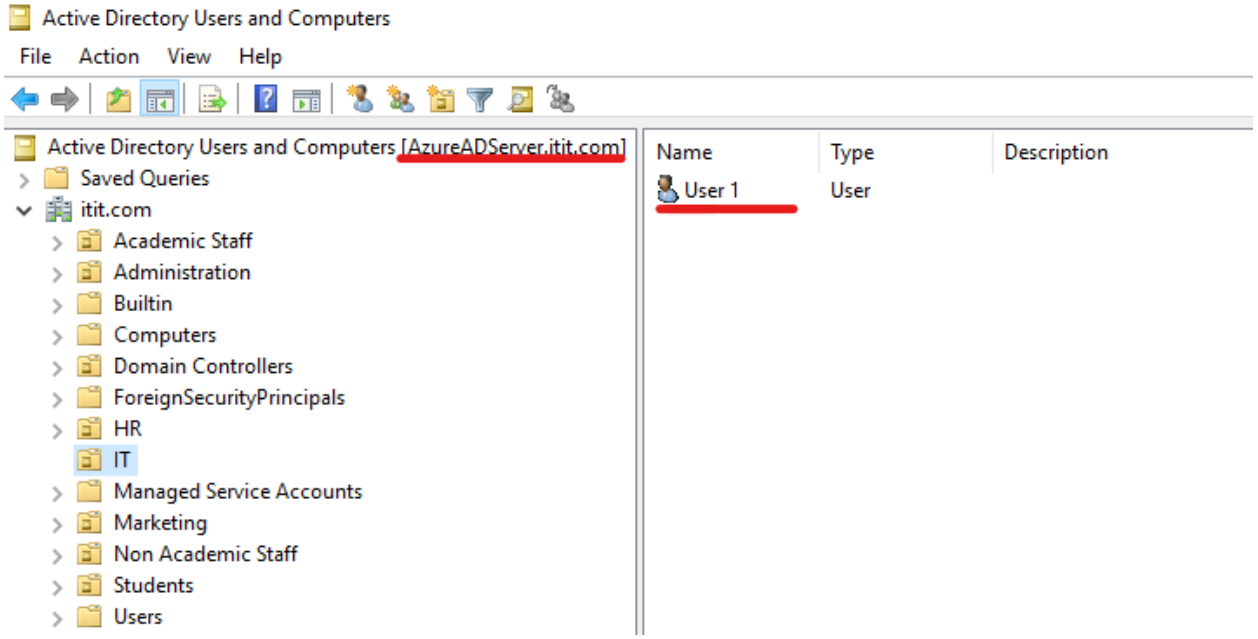


Figure 9.34

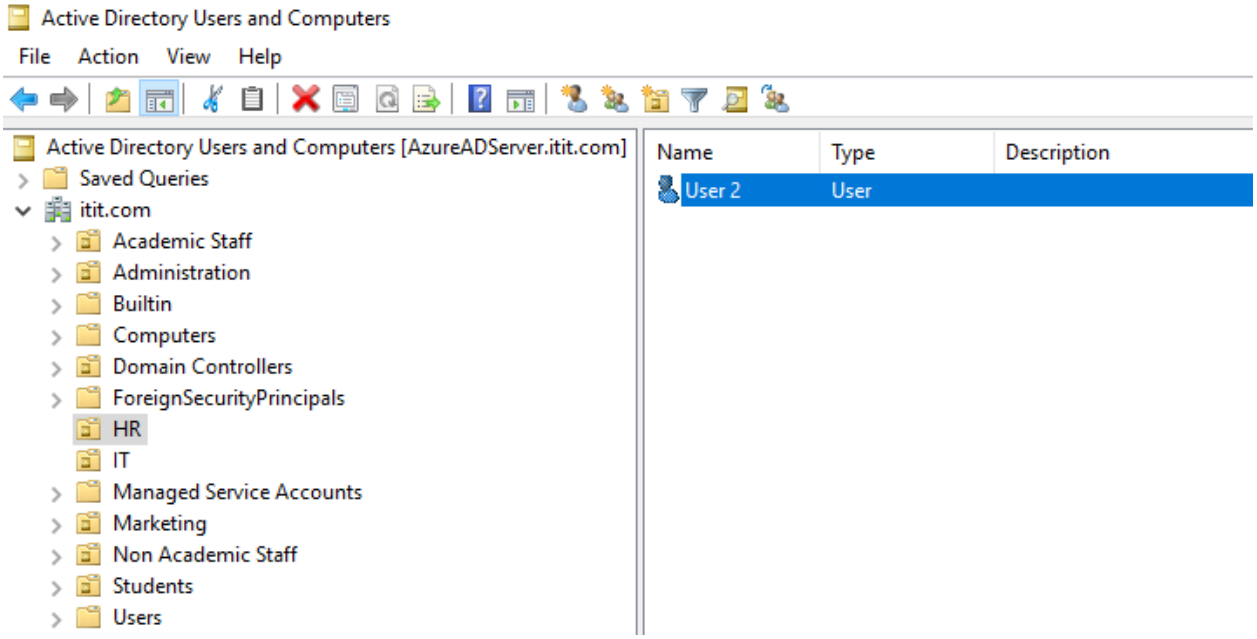


Figure 9.35

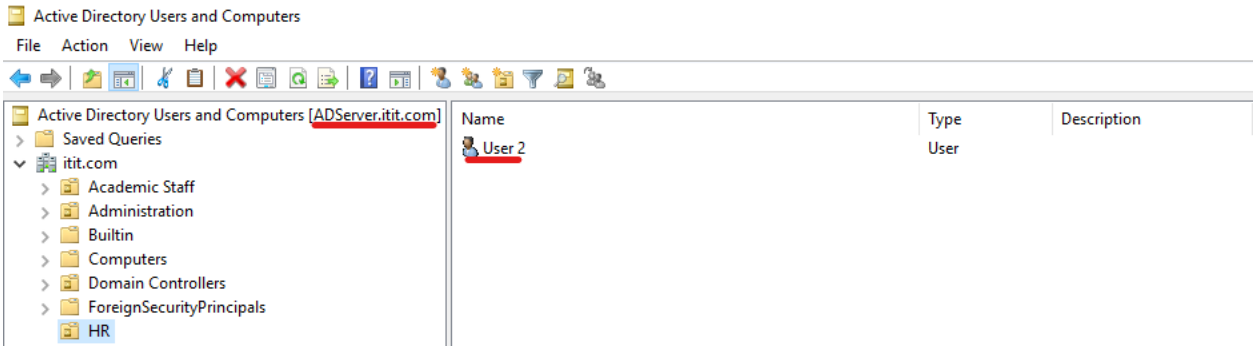


Figure 9.36

Step 15 – Now we will join a client PC to the domain.

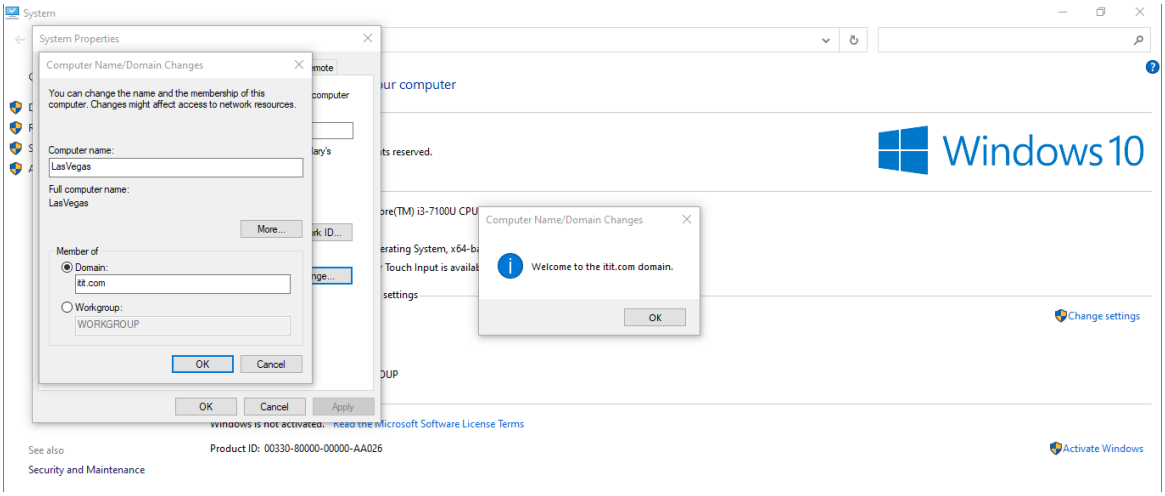


Figure 9.37

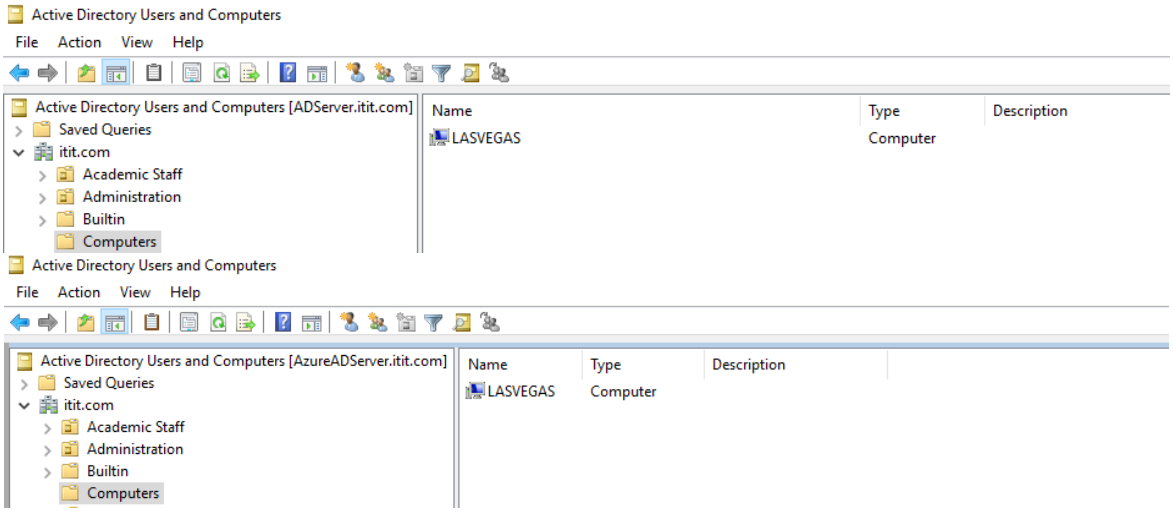


Figure 9.38

Step 16 – Now we will try to login from the first user.

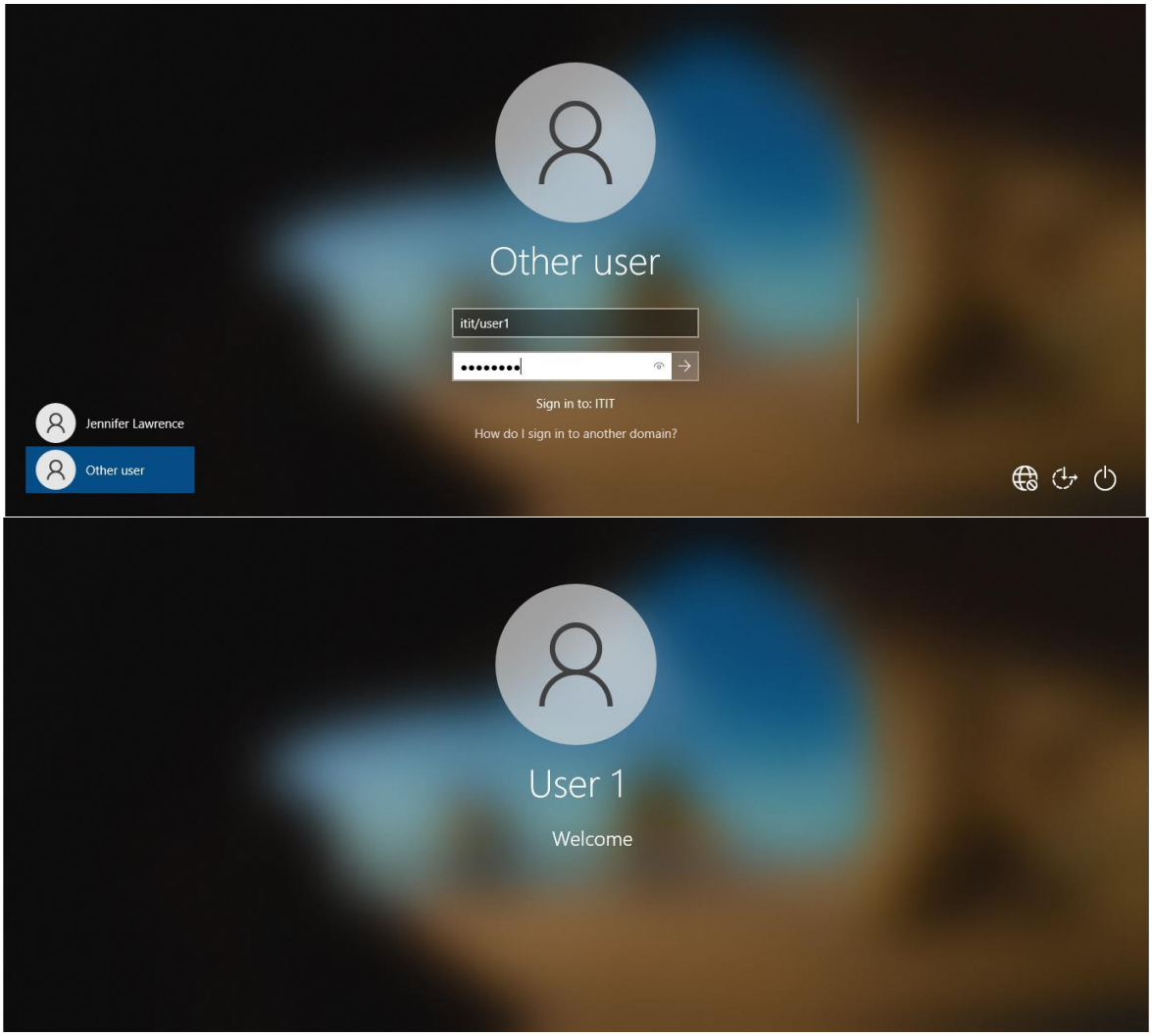


Figure 9.39

Step 17 – Now we will shut down the on – premises domain controller and try to login from the second user by authenticating from the Azure AD Server.

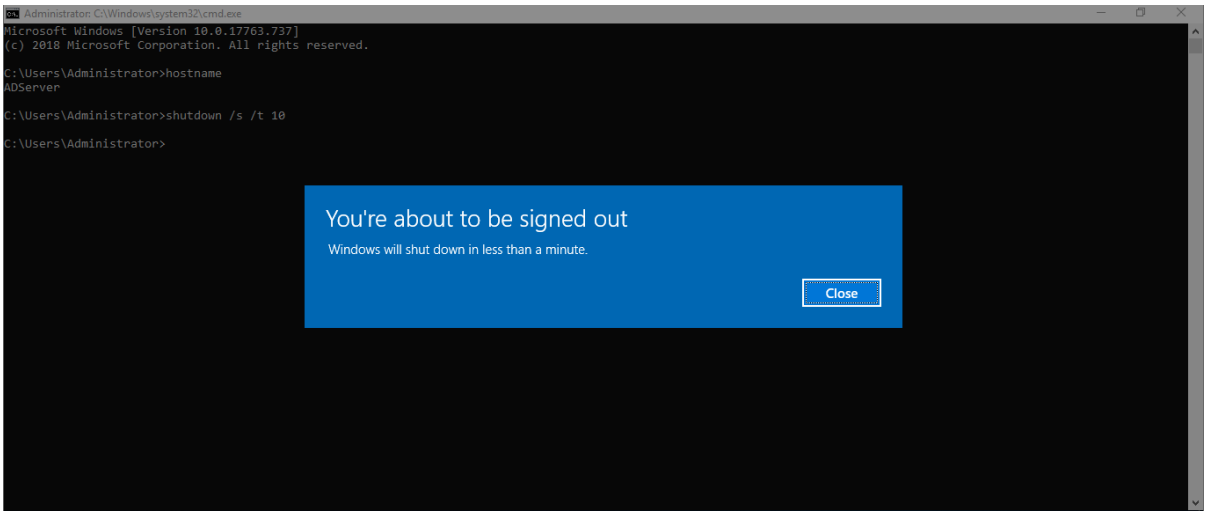


Figure 9.41

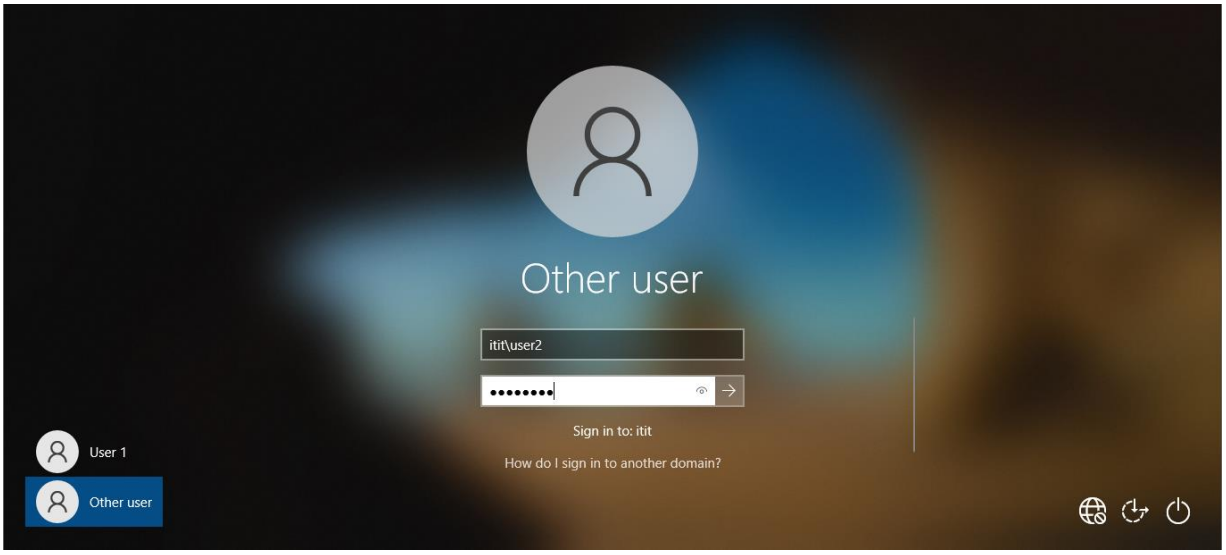


Figure 9.41

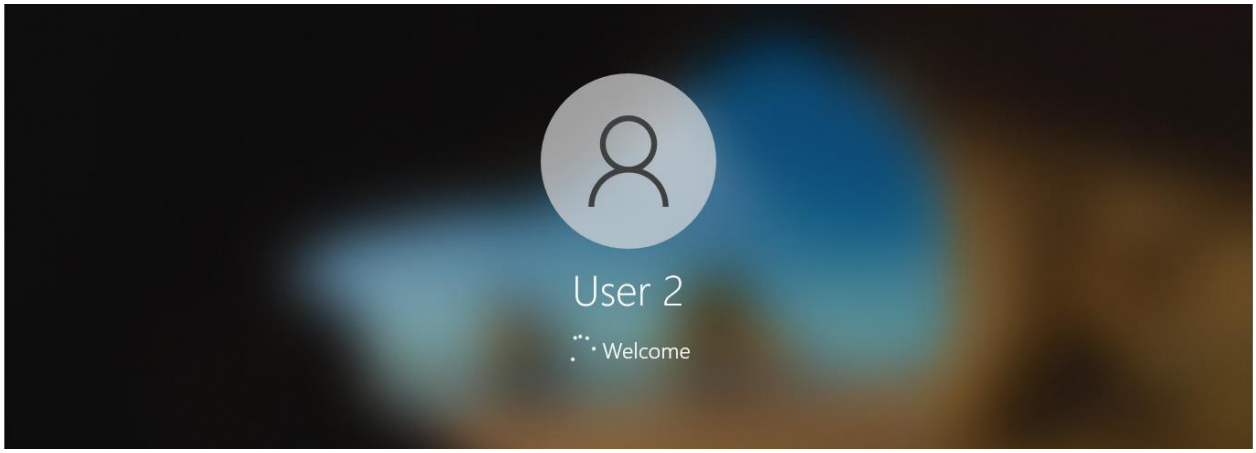


Figure 9.42

Firewall Deployment

For the Perimeter firewalls we decided to install pfsense. There will be two perimeter firewalls with High availability enabled in both of them.

Step 1 – We have configured the External Firewall 1 as follows.

```
FreeBSD/amd64 (External-Firewall-1.itit.com) (ttyv0)

VMware Virtual Machine - Netgate Device ID: 1d9dd7c68c75f2a26e9a

*** Welcome to pfSense 2.4.5-RELEASE-p1 (amd64) on External-Firewall-1 ***

 WAN (wan)       -> em0        -> v4/DHCP4: 192.168.43.60/24
 FIREWALL1TOCORE1 (lan) -> em1        -> v4: 172.16.12.65/30
 FIREWALL1TOCORE2 (opt1) -> em2        -> v4: 172.16.12.73/30
 DMZ (opt2)      -> em3        -> v4: 10.10.10.249/30
 HALINK (opt3)   -> em4        -> v4: 192.168.99.1/30

 0) Logout (SSH only)                  9) pfTop
 1) Assign Interfaces                 10) Filter Logs
 2) Set interface(s) IP address       11) Restart webConfigurator
 3) Reset webConfigurator password    12) PHP shell + pfSense tools
 4) Reset to factory defaults         13) Update from console
 5) Reboot system                     14) Enable Secure Shell (sshd)
 6) Halt system                       15) Restore recent configuration
 7) Ping host                         16) Restart PHP-FPM
 8) Shell

Enter an option:
```

Figure 10.1

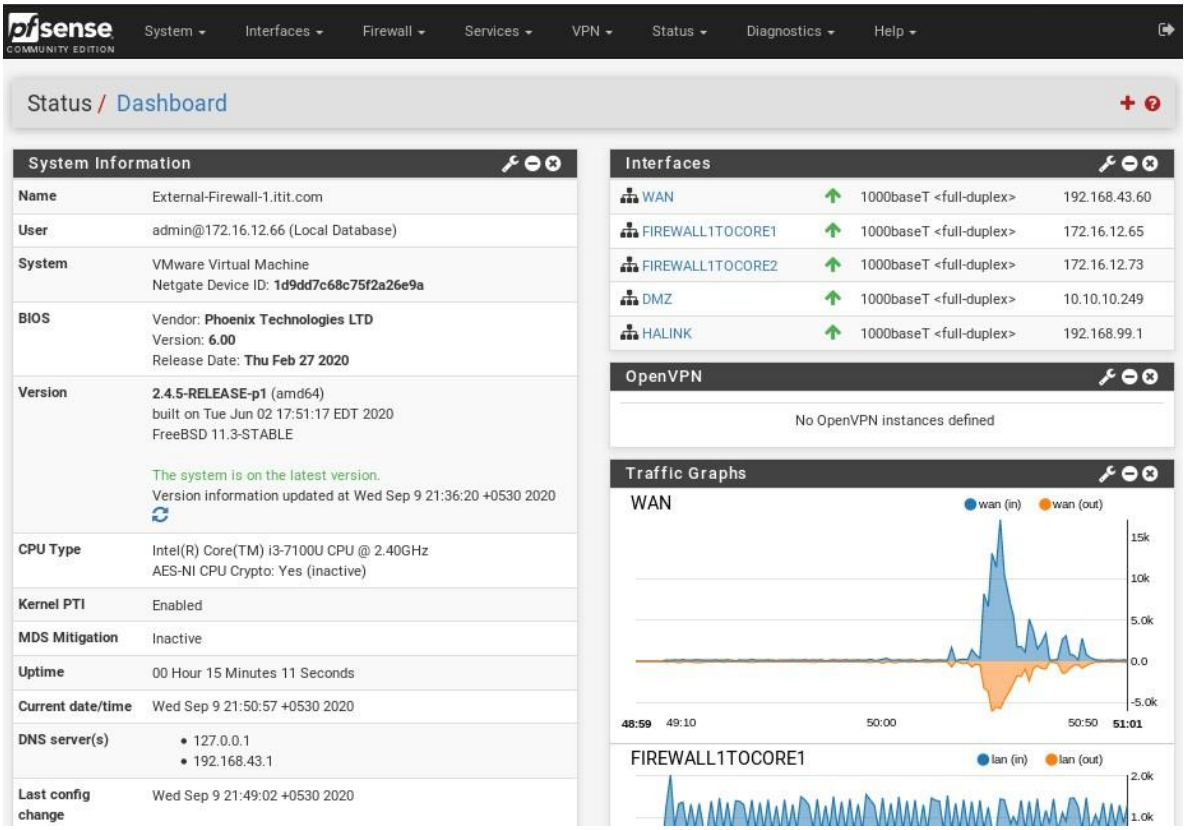


Figure 10.2

Step 2 – There will be five interfaces altogether including the WAN, Firewall 1 to Core switch 1, Firewall 1 to Core switch 2, DMZ and HA link in between the firewalls.

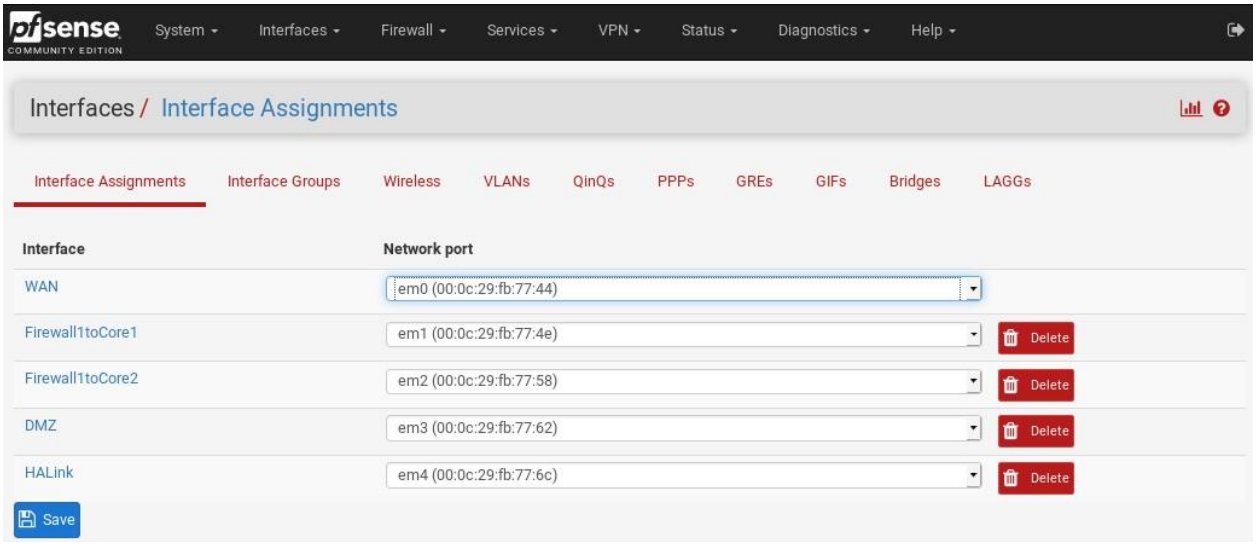


Figure 10.3

Step 3 – To enable the inbound traffic to be routed to the internal VLANs we need to enable routing in the firewall. For that we need to assign a gateway for the firewall. (IP addresses of the Core switches)

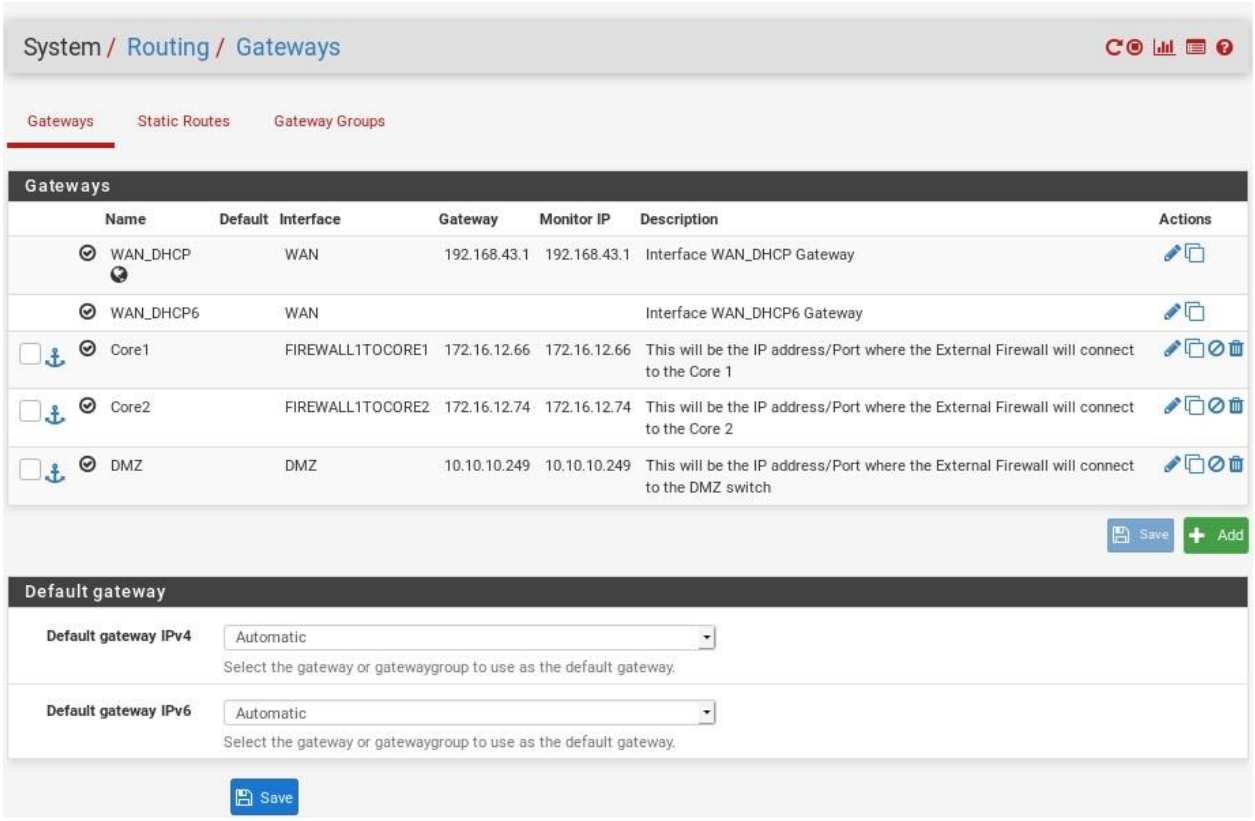


Figure 10.4

Step 4 – After specifying a gateway we entered all the network addresses of the internal VLANs to be routed.

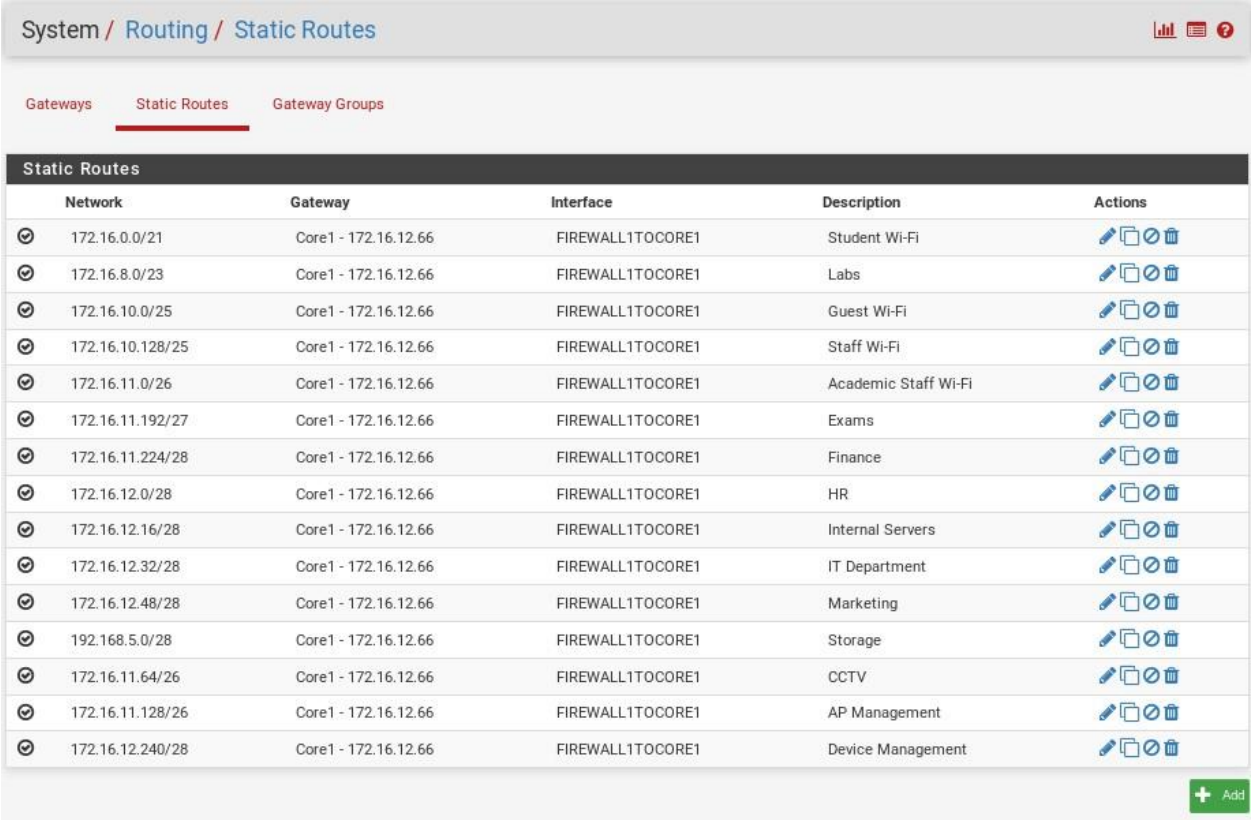


Figure 10.5

Step 5 – Now we will create the firewall rules for the LAN interfaces. For security reasons all other traffic except DNS, HTTP & HTTPS are blocked by the firewall.
NOTE: For the Server VLAN there will be an ANY to ANY rule only for the Azure subnet which is in cloud. All other traffic except that are blocked by the perimeter firewall.

Firewall / Rules / FIREWALL1TOCORE1

Floating | WAN | FIREWALL1TOCORE1 | FIREWALL1TOCORE2 | DMZ | HALINK

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✔	0 /1.27 MiB	*	*	*	FIREWALL1TOCORE1 Address	80	*	*		Anti-Lockout Rule	
Finance											
✔	0 /0 B	IPv4 TCP/UDP	172.16.11.224/28	*	*	53 (DNS)	*	none		Finance DNS Rule	
✔	0 /0 B	IPv4 TCP	172.16.11.224/28	*	*	80 (HTTP)	*	none		Finance HTTP Rule	
✔	0 /0 B	IPv4 TCP	172.16.11.224/28	*	*	443 (HTTPS)	*	none		Finance HTTPS Rule	
HR											
✔	0 /0 B	IPv4 TCP/UDP	172.16.12.0/28	*	*	53 (DNS)	*	none		HR DNS Rule	
✔	0 /0 B	IPv4 TCP	172.16.12.0/28	*	*	80 (HTTP)	*	none		HR HTTP Rule	
✔	0 /0 B	IPv4 TCP	172.16.12.0/28	*	*	443 (HTTPS)	*	none		HR HTTPS Rule	
Internal Servers !!!											
✔	0 /0 B	IPv4 *	172.16.12.16/28	*	10.10.0.0/16	*	*	none		Azure Rule	
✋	0 /0 B	IPv4 *	172.16.12.16/28	*	*	*	*	none		All other traffic from Servers are blocked.	
IT Department											
✔	0 /0 B	IPv4 TCP/UDP	172.16.12.32/28	*	*	53 (DNS)	*	none		IT DNS Rule	
✔	0 /0 B	IPv4 TCP	172.16.12.32/28	*	*	80 (HTTP)	*	none		IT HTTP Rule	
✔	0 /0 B	IPv4 TCP	172.16.12.32/28	*	*	443 (HTTPS)	*	none		IT HTTPS Rule	
Marketing											
✔	0 /0 B	IPv4 TCP/UDP	172.16.12.48/28	*	*	53 (DNS)	*	none		Marketing DNS Rule	
✔	0 /0 B	IPv4 TCP	172.16.12.48/28	*	*	80 (HTTP)	*	none		Marketing HTTP Rule	
✔	0 /0 B	IPv4 TCP	172.16.12.48/28	*	*	443 (HTTPS)	*	none		Marketing HTTPS Rule	
LAB											
✔	0 /0 B	IPv4 TCP/UDP	172.16.8.0/23	*	*	53 (DNS)	*	none		LAB DNS Rule	
✔	0 /0 B	IPv4 TCP	172.16.8.0/23	*	*	80 (HTTP)	*	none		LAB HTTP Rule	
✔	0 /0 B	IPv4 TCP	172.16.8.0/23	*	*	443 (HTTPS)	*	none		LAB HTTPS Rule	

Figure 10.6

Student Wi-Fi											
✔	0 /590 KiB	IPv4 TCP/UDP	172.16.0.0/21	*	*	53 (DNS)	*	none		Student Wi-Fi DNS Rule	
✔	0 /0 B	IPv4 TCP	172.16.0.0/21	*	*	80 (HTTP)	*	none		Student Wi-Fi HTTP Rule	
✔	0 /0 B	IPv4 TCP	172.16.0.0/21	*	*	443 (HTTPS)	*	none		Student Wi-Fi HTTPS Rule	

Figure 10.7

Step 6 – Now we will create the VPN tunnel which we use to communicate with Azure through a secure channel.
Key Exchange version – IKEv2
Internet protocol – IPv4
Interface – WAN
Remote gateway – Public IP address of the Azure VPN gateway.
Authentication Method – Mutual PSK (We only use a pre-shared key which is the same for every user and after securing the channel the user authentication).

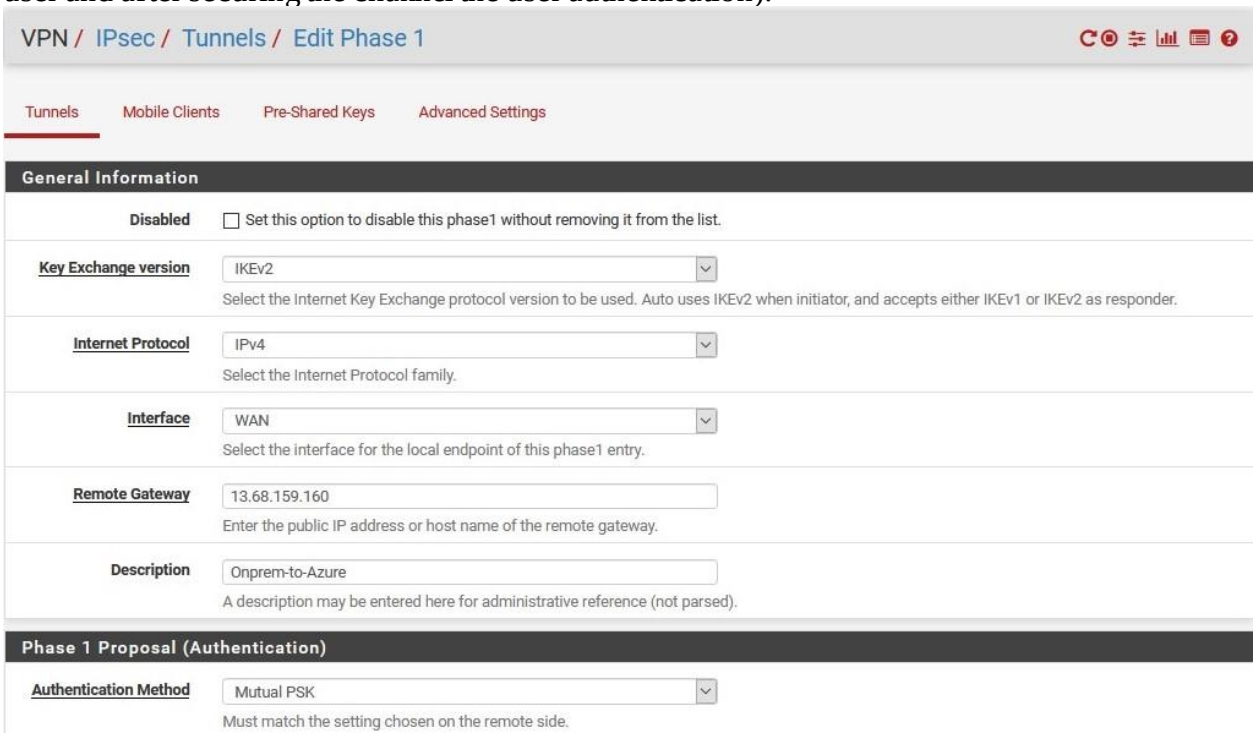


Figure 10.8

Step 7 – Select My identifier & Peer identifier as IP address. For the Pre – Shared key use the same which we used when creating the Azure VPN gateway. This key must match on both peers. For Phase 1 encryption algorithm select AES with the following attributes (Key length = 256 bits, Hash = SHA 256 & DH group = 1024 bit).

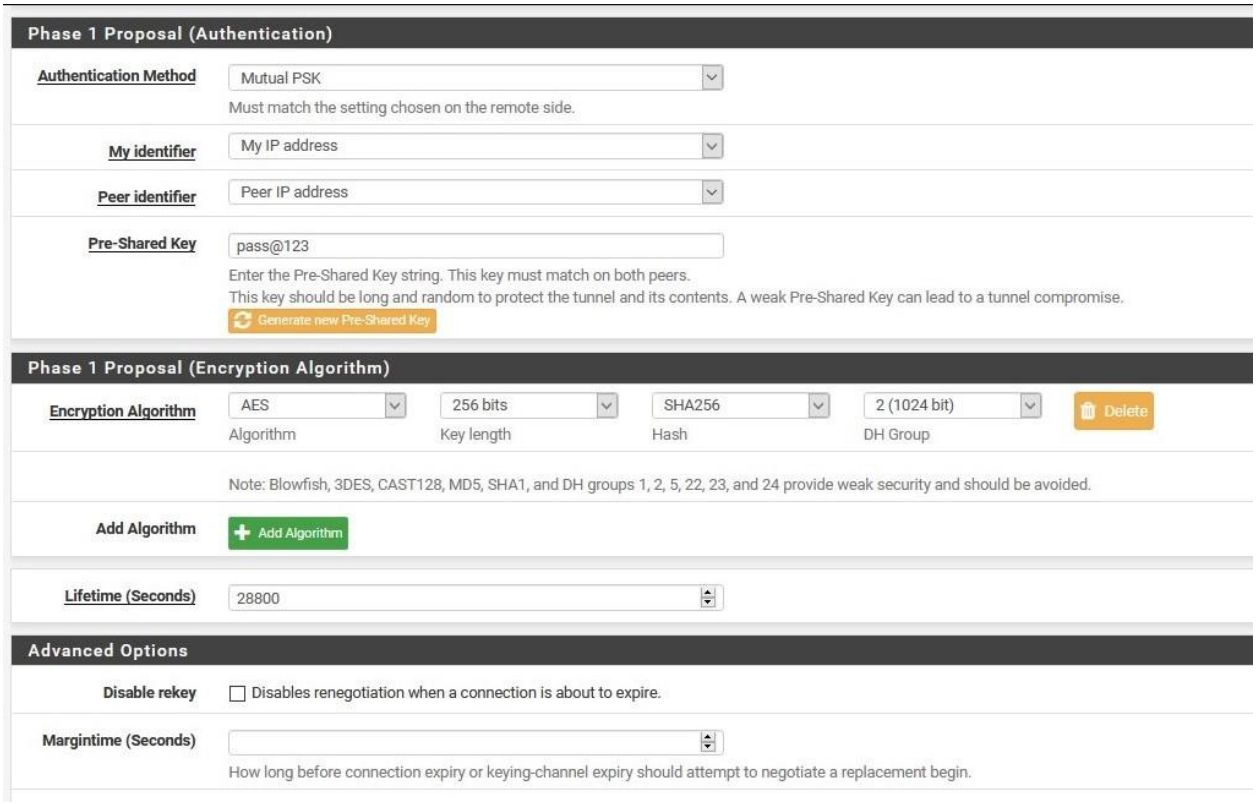


Figure 10.9

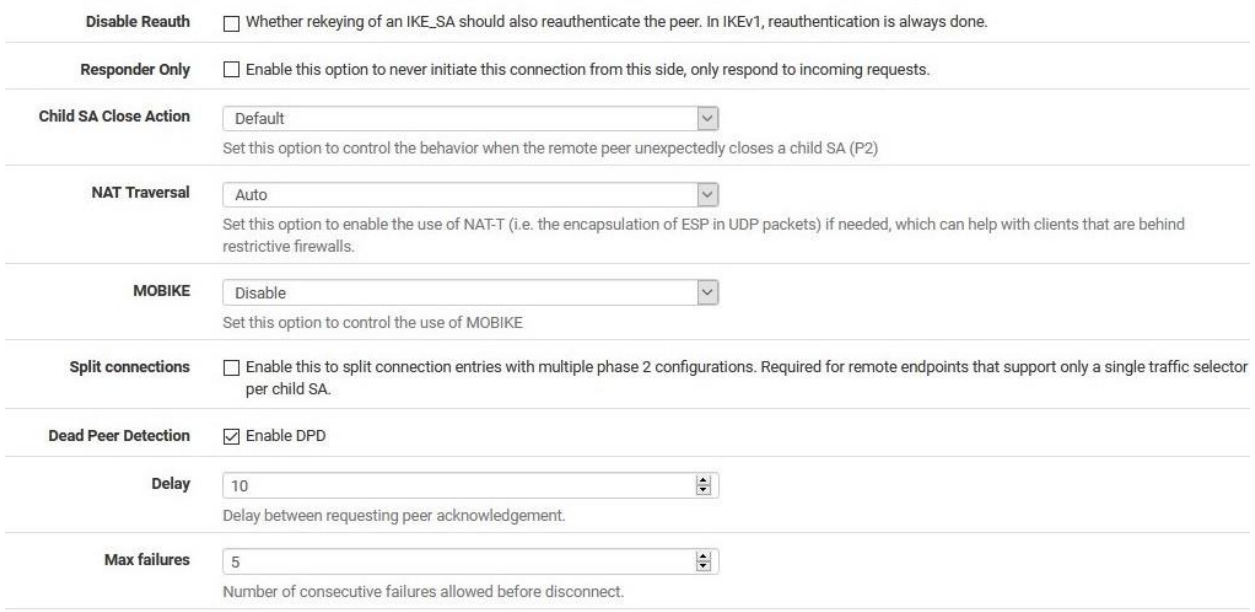


Figure 10.11

Step 8 – When creating the tunnel phase 2 we should specify each other networks on both sides (On premises & Azure).
For the remote networks specify the Azure VNET network address. For encryption algorithms select AES & AES 128 – GCM. For HASH algorithms select SHA 1, SHA 256.

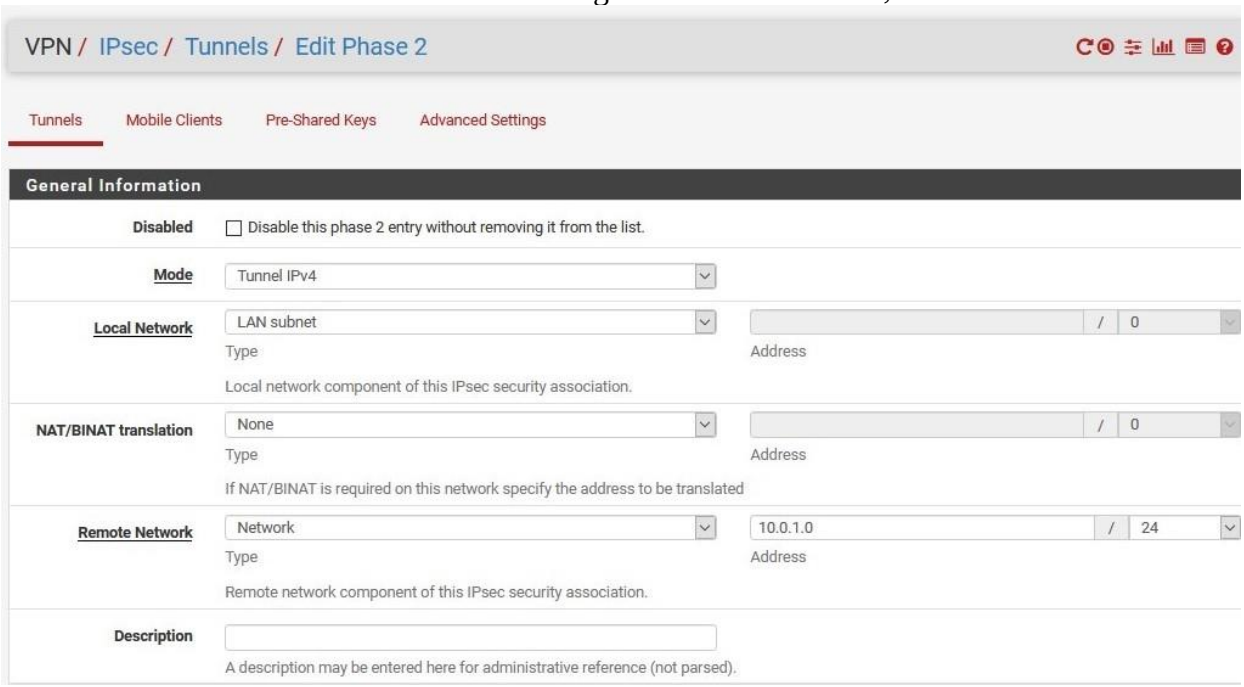


Figure 10.12

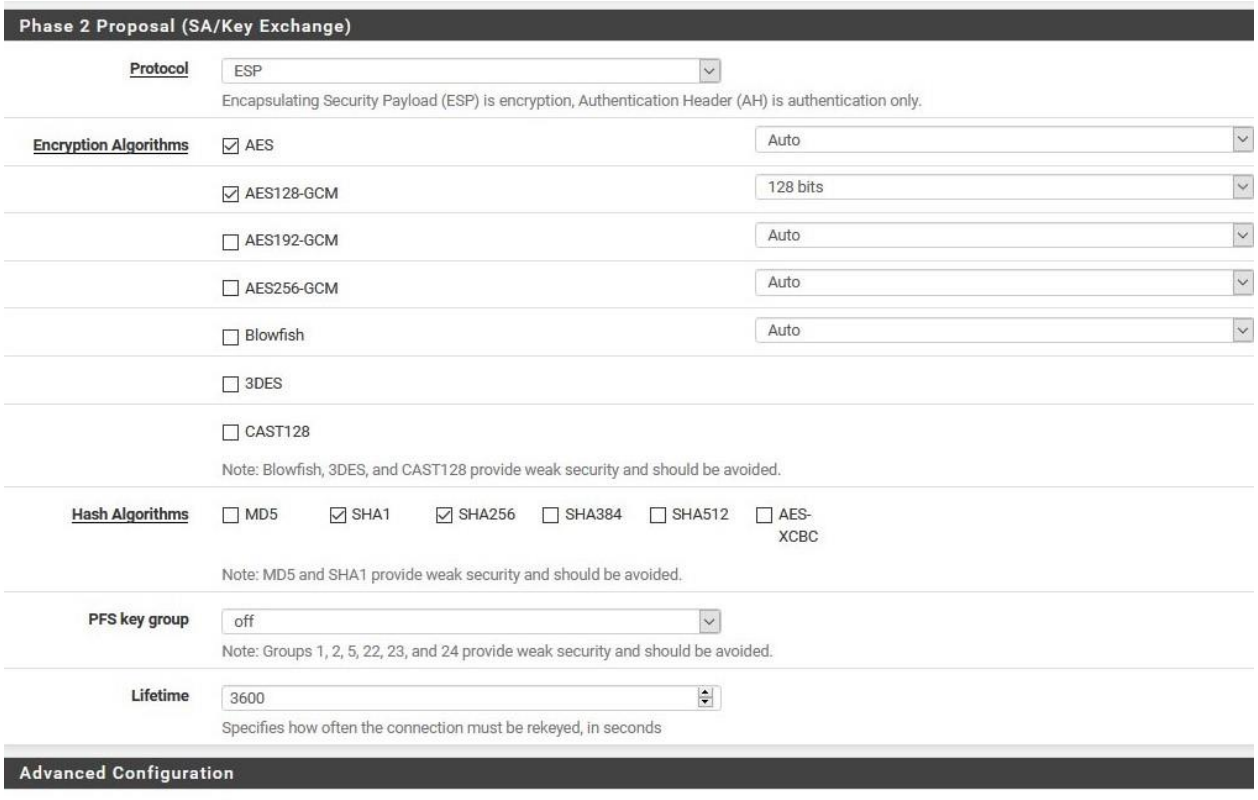


Figure 10.13

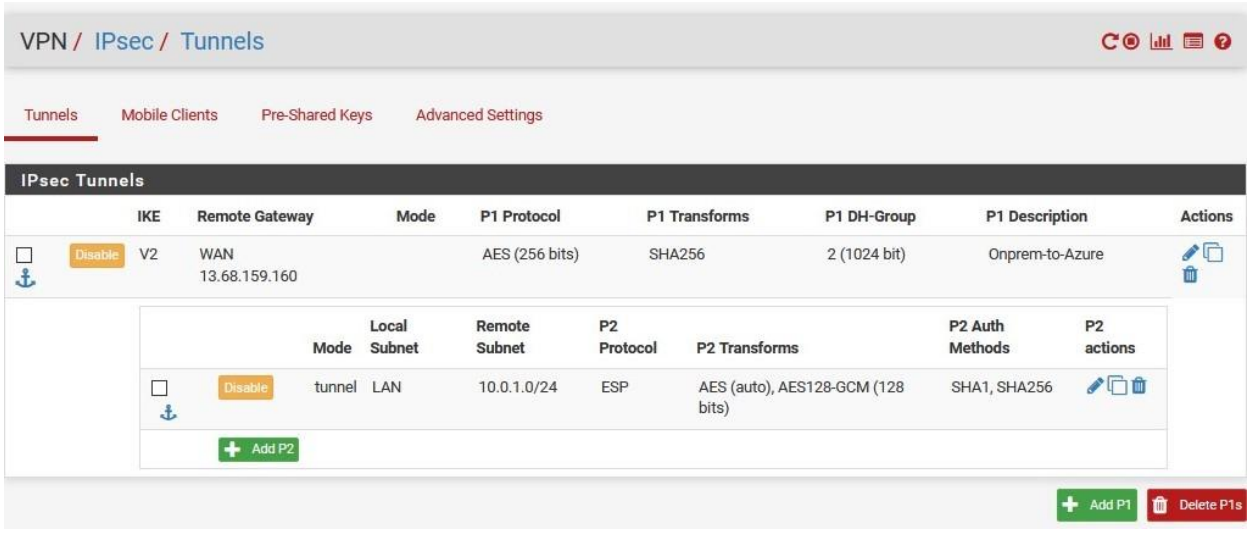


Figure 10.14

Step 9 – Create an ANY to ANY rule for the IPsec VPN tunnel.

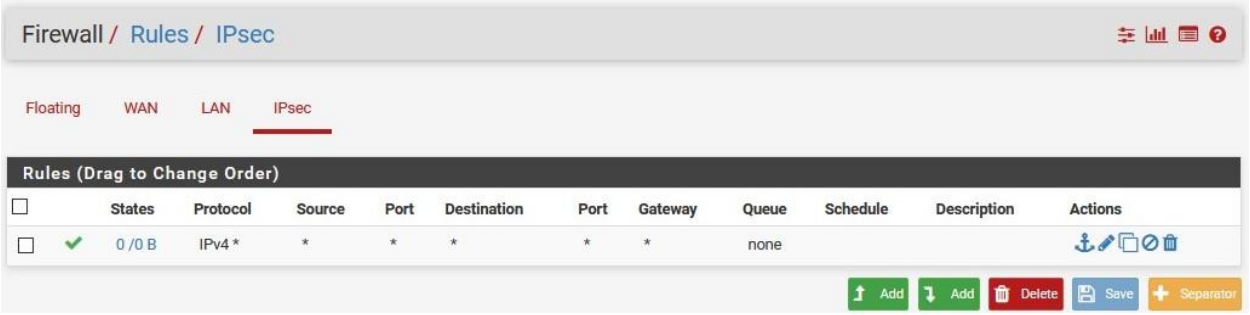


Figure 10.15

Step 10 – Create another Phase 2 entry for internal VLANs.

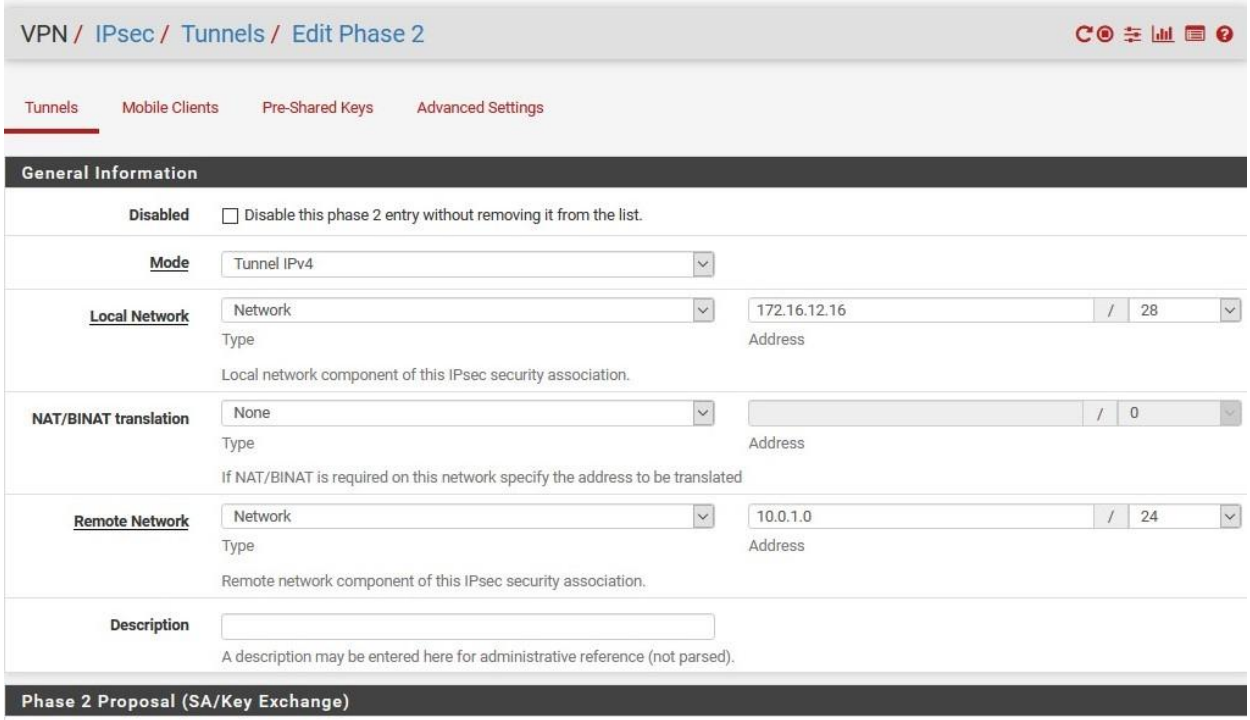


Figure 10.16

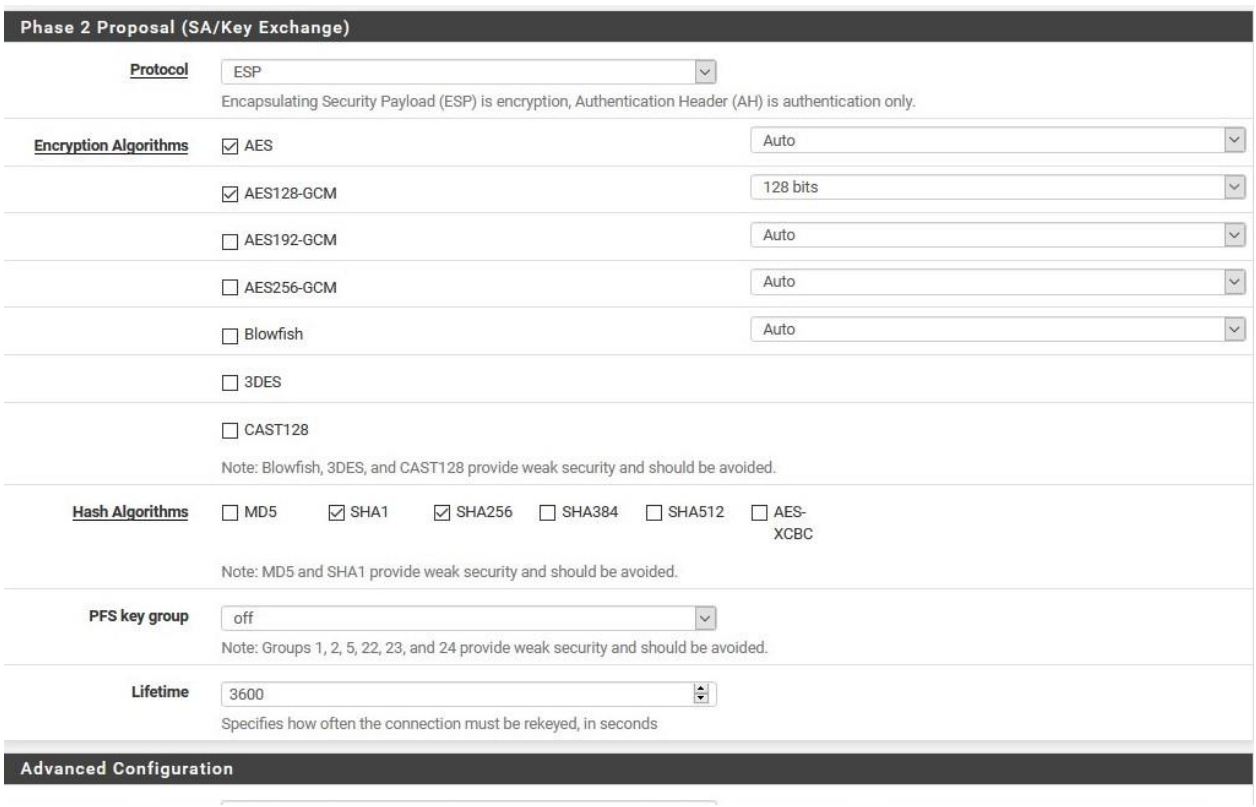


Figure 10.17

Step 11 – As you can see the after doing the necessary changes we can communicate through the tunnel.

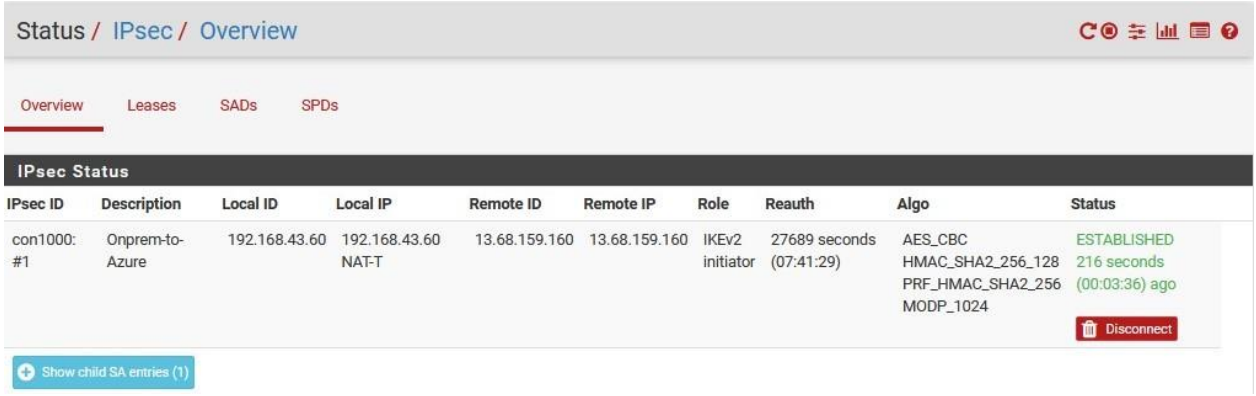


Figure 10.18

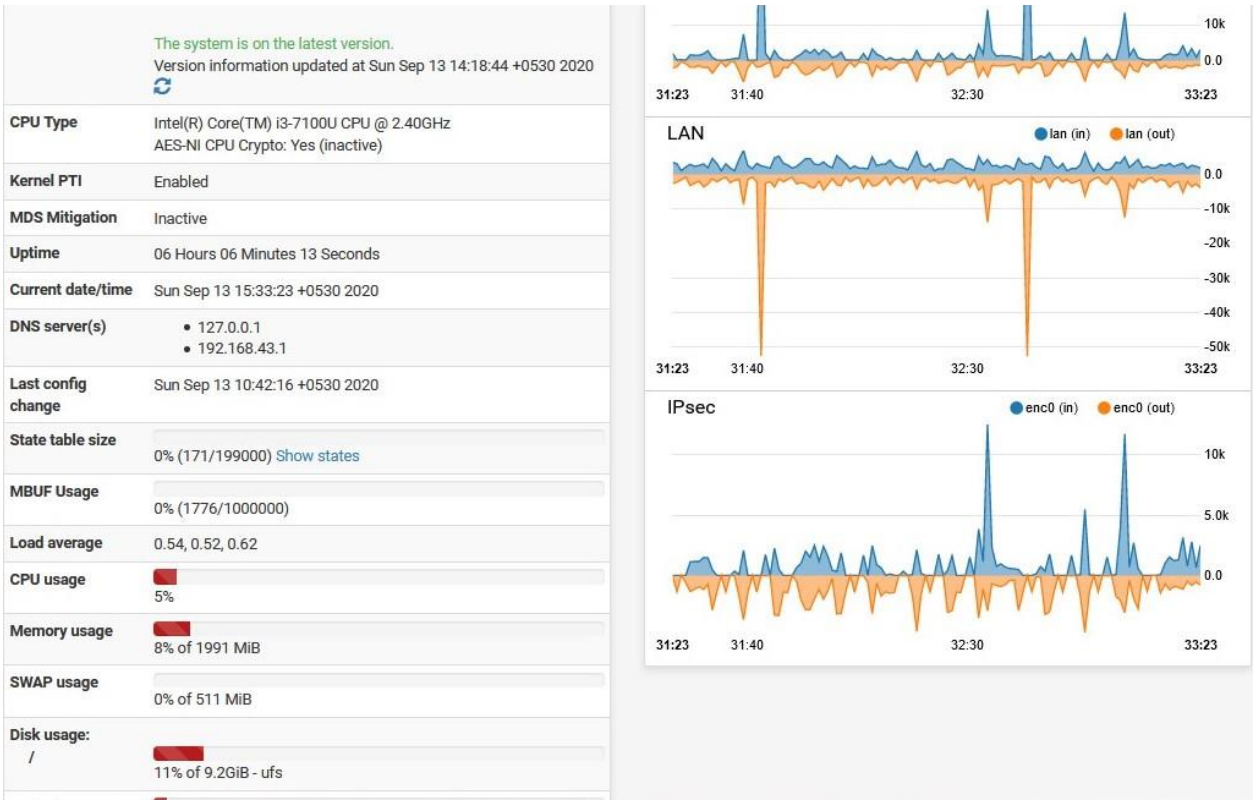

The system is on the latest version.
Version information updated at Sun Sep 13 14:18:44 +0530 2020
CPU Type
Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz
AES-NI CPU Crypto: Yes (inactive)
Kernel PTI
Enabled
MDS Mitigation
Inactive
Uptime
06 Hours 06 Minutes 13 Seconds
Current date/time
Sun Sep 13 15:33:23 +0530 2020
DNS server(s)
127.0.0.1
192.168.43.1
Last config change
Sun Sep 13 10:42:16 +0530 2020
State table size
0% (171/199000) Show states
MBUF Usage
0% (1776/1000000)
Load average
0.54, 0.52, 0.62
CPU usage
5%
Memory usage
8% of 1991 MiB
SWAP usage
0% of 511 MiB
Disk usage:
/
11% of 9.2GiB - ufs
10k
0.0
31:23
31:40
32:30
33:23
LAN
lan (in)
lan (out)
0.0
-10k
-20k
-30k
-40k
-50k
31:23
31:40
32:30
33:23
IPsec
enc0 (in)
enc0 (out)
10k
5.0k
0.0
31:23
31:40
32:30
33:23

Figure 10.19

Conclusion

Through the concepts utilized in this context, is with the co-network architecture and design processors. Through them, the institute accomplishes redundancy, infrastructure flexibility, growth and minimizes down time.

By including a fully redundant connectivity with strategic and long term planning and infrastructure the higher educational institute of ITIT, network infrastructure is geared for maximizing fault tolerance and provides high availability for its users

Furthermore having duel redundant physical devices in its place, and the powerful; real time visibility products and virtualizing technologies, such as VM center hypervisor, VSphere and network storage devices are used to prevent and minimize network outages.
Further research project to improve the network latency and performance would be within the primary domain controllers that authenticate the user within the whole building will be done by cloud technologies.

Through this it will eliminate thousands of authentication traffic from the WAV circuits. This makes a significant difference and maintains a network bandwidth resources in the network infrastructure

This educational organization achieved its objectives by creating a new network redundancy with the usage of cloud technologies and increased bandwidth speed and reduced network utilization through virtualization.
Often times when performing a massive network upgrade, its good practice to take into consideration alternative backup plans and total costs associated with them to compare them to the primary plan.
The planning phase is key to take into account, every hidden planned and unplanned expense and make strategic, informed decisions based on them

Thus in Summarization , this project reflects upon the design considerations and implementation techniques of an infrastructural flexibility , layered Protection , High Availability and Redundancy to deliver the best networking solution and seamless connections with efficient use of network resources to assure the best learning experience to the users of ITIT Educational Institute.

Reference

https://www.router-switch.com/

https://www.geeksforgeeks.org

https://stackoverflow.com/

https://docs.microsoft.com/en-us/azure/cloud/

https://docs.microsoft.com/en-us/azure/virtual/

https://www.dell.com/en-us/work/shop/servers-storage-and-networking/

https://community.cisco.com/t5/networking-documents/how-to-configure-a-gre-tunnel

https://scholar.google.com/

https://www.ciscopress.com/articles

https://www.pfsense.org/

https://subnettingpractice.com/vlsm.html

https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.esxi.install.doc

https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-esxi-67-installation-setup-guide.pdf

https://geek-university.com/vmware-esxi/basic-esxi-configuration/

https://www.howtoforge.com/samba-server-installation-and-configuration-on-centos-7

https://www.tecmint.com/install-samba4-on-centos-7-for-file-sharing-on-windows/

https://www.tecmint.com/install-zimbra-collaboration-suite-on-centos-rhel/

http://docs.zimbra.com/docs/os/6.0.10/single_server_install

https://computingforgeeks.com/install-zimbra-mail-server-on-centos-rhel/

https://docs.oracle.com/cd/E19316-01/820-3746/gisdn/index.html

https://www.rebeladmin.com/2014/07/step-by-step-guide-to-setup-active-directory-on-windows-server-2019/

https://docs.netgate.com/pfsense/en/latest/config/

https://www.tecmint.com/installation-and-configuration-of-pfsense-firewall-router/

https://azure.microsoft.com/en-us/solutions/backup-and-disaster-recovery/

https://creately.com/lp/activity-diagram-tool/

https://online.visual-paradigm.com/

https://www.edrawsoft.com/edraw-max

https://www.lorextechnology.com/hd-ip-camera-system/N-h4qcu7

https://www.cctvcameraworld.com/ip-cameras.html

https://pro.sony/en_SI/products/ip-cameras

https://techthoughts.info/synology-disk-station-manager-guide/

https://www.howtogeek.com/318018/how-to-set-up-and-get-started-with-your-synology-nas/

https://www.cisco.com/c/dam/en_us/about/ciscoitatwork/

https://www.starwindsoftware.com/blog/vcenter-server-appliance-6-7-u1-installation-and-configuration-guide

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

https://masteringvmware.com/installing-the-vcenter-server-6-step-by-step/

https://infohub.delltechnologies.com/l/deployment-guide-vmware-vsphere/installing-esxi-and-deploying-vcenter-server-appliance-on-the-management-server