



ST4065CEM COMPUTER SYSTEMS AND NETWORKS ETHICAL HACKING AND CYBERSECURITY



SUBMITTED BY:
DINESH JOSHI
SUDI: 240391
CUID: 15945644

SUBMITTED TO
ROHAN JOSHI
(MODULE LEADER)

Acknowledgement

I would like to thank Mr. Rohan Joshi for his guidance and help during this endeavor. His guidance and encouragement were really helpful in completing the assignment properly. I appreciate his time and effort in supporting me throughout the process.

Abstracts

This document analyzes other significant topics on networking like network design, security threats and comparison of protocols. Advantages, disadvantages, and counteractive steps of the TCP and UDP have been compared to each other. Network security is investigated through practical protections and attacks like STP, DHCP spoofing and MITM. The aim of a three tier network architecture was to service a total of five departments using the 172.16.99.0/16 IP block with DHCP and DNS facilitated and VLANs blocked to separate traffic. The device configuration has hostnames, user account, password encryption, access through Telnet/SSH, EtherChannel, redundancy, RSTP, and disabling DTP. Finally the topic on secure and efficient network management is addressed as well as the use of Linux firewall configuration via iptables, VPN concepts including the use of network operating systems they include windows server operating system and Linux.

Table of Contents

Introduction.....	1
1.1. Comparison of TCP and UDP.....	2
Overview.....	2
Key Differences: TCP and UDP.....	3
Strength and Weakness.....	4
TCP.....	4
UDP.....	4
Sustainability for Application.....	5
Security Vulnerabilities.....	6
Mitigation Measures.....	7
1.2. Network Security Mechanisms Against Common Network Attacks.....	8
Man-in-the-Middle (MITM) Attack.....	8
DHCP Spoofing Attack.....	8
STP (Spanning Tree Protocol) Attacks.....	9
Demonstration of MITM.....	9
Objective.....	9
Mitigation.....	13
2. Three-Tier Network Deployment.....	14
Subnetting and VLAN Allocation Table.....	14
2.1. Access Layer Switch Configuration (Layer 2 Switch).....	15
ACCESS-1_bitter!honey (SALES).....	15
2.2. Distribution Layer Switch Configuration (Layer 3 Switch).....	16
Trunk and VTP Configuration.....	16
Etherchannel Configuration using PAgP in Distribution layer.....	18
Configuration Default Gateways (SVIs).....	18
Configuration of Routing Protocol.....	20
2.3. Core Layer Switch Configuration.....	21
Etherchannel Configuration using LACP.....	21
Configuration IP Addresses.....	21
Configuring Routing Protocol in Core layer.....	22
2.4. Configuring EDGE and ISP (Router).....	23
Configuring IP Addresses in EDGE and ISP.....	23
Configuring Routing Protocol in EDGP and ISP.....	24
2.5. Configuration of Internal Servers.....	25
3. Secure Network Deployment.....	29
3.1. Assigning Hostname and Description to Interface.....	30

3.2. Assigning IP Address.....	31
3.3. Creating User Account with Secure Password.....	32
3.4. Implementing Telnet and SSH with Local Credentials.....	33
3.5. Configuring MOTD.....	34
3.6. Difference Between SSH and Telnet.....	34
3.7. Implementing Encryption Techniques.....	36
3.8. Implementing RSTP.....	36
3.9. Configuring Etherchannel.....	37
3.10. Disabling DTP.....	38
4. Network Security Practices and Technologies in Linux and Enterprise Environments...	
40	
4.1. IPtables Configuration.....	40
Best Practices for Firewall Configuration.....	41
iptables Configuration Showcase.....	43
4.2. VPN Components and Security.....	45
Working Mechanism of VPN.....	45
Key Components and Technologies in VPN.....	46
4.3. Understanding Network Operating Systems.....	47
Overview.....	47
Popular NOS.....	47
Key Feature.....	48
Common Protocol.....	48
Conclusion.....	49
References.....	50
Appendix.....	52

List of Figures

Figure 1: TCP VS UDP.....	2
Figure 2: Key Difference between TCP and UDP.....	3
Figure 3: When to Use TCP and UDP.....	5
Figure 4: Network Scanning.....	10
Figure 5: ARP Spoofing and Packet spoofing.....	11
Figure 6: Traffic Sniffing.....	12
Figure 7: Target Interacted Files.....	12
Figure 8: Mitigation Measure against MITM.....	13
Figure 9: Three Tier Network Architecture.....	14
Figure 10: VLAN 10 Sales.....	15
Figure 11: Trunk VLAN 10 SALES.....	16
Figure 12: trunk in DIST-1_bitter!honey.....	17
Figure 13: VTP Status.....	17
Figure 14: Etherchannel Status in Distribution layer.....	18
Figure 15: VLANs Default Gateways.....	18
Figure 16: VLANs Interface Configuration.....	19
Figure 17: Routing Protocol Status in Distribution layer.....	20
Figure 18: Core layer etherchannel configuration.....	21
Figure 19: Configure IP Addresses in Core layer.....	21
Figure 20: IP routing in Core layer.....	22
Figure 21: EDGE_ROUTER_bitter!honey IP Addresses Configuration.....	23
Figure 22: ISP_bitter!honey IP Address Configuration.....	23
Figure 23: Routing Protocol of ISP_bitter!honey.....	24
Figure 24: Pinging ISP from end device.....	25
Figure 25: DHCP Configuration.....	25
Figure 26: Dynamically assigned IP to PC3 via DHCP Server.....	26
Figure 27: Web Server.....	27
Figure 28: Accessing Web Page.....	28
Figure 29: Network Architecture.....	29
Figure 30: Assigned Hostname and Description to Interface.....	31
Figure 31: Assigning IP Address.....	31
Figure 32: Creating User with secure password.....	32
Figure 33: Implementing SSH and Telnet.....	33
Figure 34: MOTD banner.....	34
Figure 35: Telnet Login.....	35
Figure 36: SSH Login.....	35

Figure 37: Encrypted Password.....	36
Figure 38: RSPT Implemention.....	36
Figure 39: Configuration of Etherchannel.....	37
Figure 40: Turning off DTP.....	38
Figure 41: Disabled DTP.....	39
Figure 42: Setting Default Policies.....	40
Figure 43: Setting up Inbound Rules.....	40
Figure 44: Setting up Outbound Rules.....	41
Figure 45: Enabling logs.....	41
Figure 46: Best Practice.....	42
Figure 47: Opening Ports.....	43
Figure 48: SSH Connection Successful.....	43
Figure 49: Failed HTTP Connection.....	44
Figure 50: How VPN Works.....	45
Figure 51: VPN Components and Technologies.....	46
Figure 52: NOS Model.....	47
Figure 53: Key Feature Distinguishing NOS from Traditional OS.....	48

Introduction

This assignment covers core networking and security principles. It initially compared TCP and UDP. UDP is faster but less reliable, whereas TCP is more reliable because it makes connections and checks for errors. The release addresses vulnerabilities that could jeopardize network integrity, such as flooding and session hijacking. Furthermore, it discusses common attacks like as Man-in-the-Middle, DHCP spoofing, and STP exploits, as well as suggestions for how to defend against them. This network design is appropriate for smaller enterprises. It uses VLANs, DHCP, and DNS to provide efficient business-to-business communication, thanks to its three-tier design and exceptional subnetting. There are also protocols for protecting network devices, such as specifying permission levels, using encrypted passwords, and allowing external access via SSH and Telnet. Finally, it shows how to set up a firewall with iptables and how to use virtual private networks (VPNs) to securely connect to the internet from a distance. This study usually links fundamental theory with experience to ensure network security.

1.1. Comparison of TCP and UDP

Overview

Transmission Control Protocol (TCP) and User Data-gram Protocol (UDP) are two core protocols at the transport layer of the Internet protocol suite. They enable communication between devices but differ significantly in their design, reliability, speed, and use cases. ([BasuMallick, 2022](#))

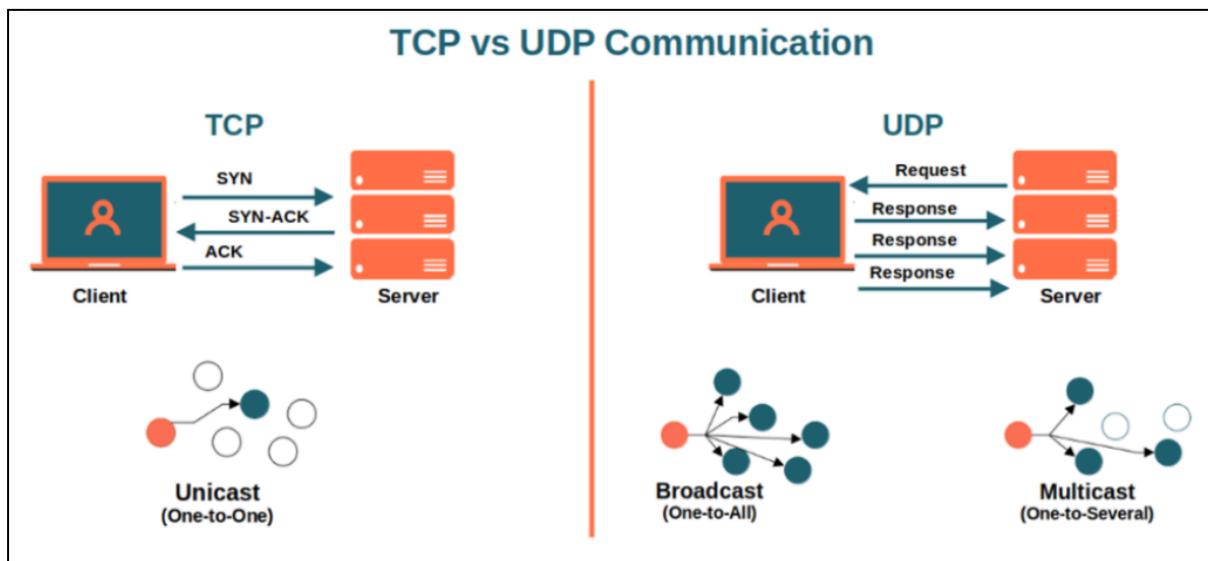


Figure 1: TCP VS UDP

Key Differences: TCP and UDP

Feature	TCP	UDP
Connection Type	Connection-oriented — The device must establish a connection before transmitting data (handshake included) and close the connection after transmission.	Connectionless — No connection and no handshake are required to send data.
Reliability	Reliable — Data packet delivery is guaranteed.	Unreliable — Datagram delivery isn't guaranteed.
Speed	Slower than UDP as it follows many steps to ensure accuracy.	Faster than TCP as it's a much simpler protocol.
Header Size	Variable — 20-60 bytes header length.	Fixed — 8 bytes.
Suitability	Web browsing. File transfer. Email (SMTP, IMAP/POP)	VPN. Video and music streaming. Online games, multiplayer games. Live broadcasts and video conferencing. Domain name system (DNS) queries. Voice over IP (VoIP).
Communication Information Type	Stateful — The client and the server keep the information about the session.	Stateless — The server doesn't keep any information about the session.

Figure 2: Key Difference between TCP and UDP

Strength and Weakness

TCP

Strength:

1. Guarantees reliable, in-order delivery of data.
2. Provides flow and congestion control to manage network traffic.
3. Suitable for applications where data integrity and accuracy are critical, such as web browsing, email, and file transfers.

Weaknesses:

1. Higher latency due to connection setup (three-way handshake) and error correction mechanisms.
2. More resource-intensive, leading to slower performance compared to UDP.
3. Less suitable for real-time or latency-sensitive applications like live streaming or online gaming.

UDP

Strength:

1. Low latency and minimal protocol overhead, resulting in faster data transmission.
2. Well-suited for real-time applications where speed is prioritized over reliability, such as video streaming, online gaming, and VoIP.
3. Simple protocol design, making it efficient for broadcast and multi-cast communications.

Weaknesses:

1. No guarantee of delivery, order, or error correction data may be lost, duplicated, or received out of order.
2. Lacks congestion and flow control, which can lead to network congestion.
3. Not suitable for applications that require reliable and accurate data delivery.

Sustainability for Application

TCP and UDP are both widely used transport layer protocols, but their differing characteristics make each more suitable for specific types of applications.

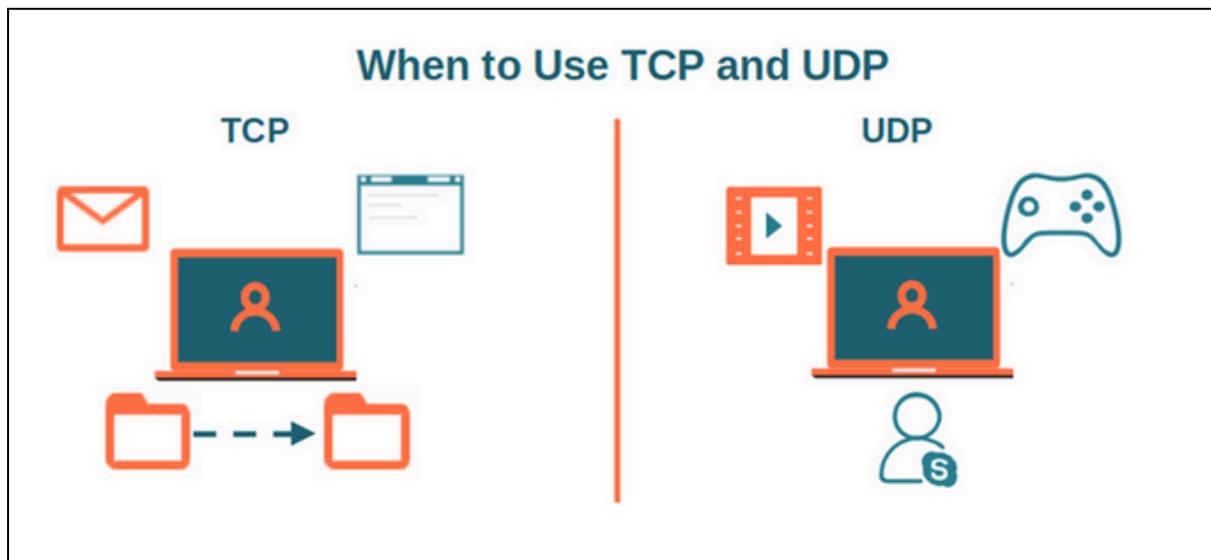


Figure 3: When to Use TCP and UDP

TCP: Best for Reliability

1. Rate limiting.
2. Firewalls and intrusion detection systems.
3. Application-level request validation.
4. Network-level anti-DDoS solutions.

UDP: Best for Speed and Real-Time Communication

1. Streaming Media(Audio, Video): Supports real-time playback even if some data packets are lost.
2. Domain Name System(DNS): Queries: Allows rapid, lightweight requests and responses.
3. Voice Over IP(VoIP): Delivers voice data quickly for real-time conversations.
4. Online Gaming: Prioritizes fast updates and minimal lag over perfect accuracy.

Security Vulnerabilities

TCP: Session Hijacking

TCP's connection-oriented nature requires a handshake to establish a session. Attackers can exploit this by intercepting or predicting session information, enabling them to hijack an active session and inject malicious data or commands. ([TCP Session Hijacking, 2009](#))

UDP: Flooding Attack

UDP's connectionless design makes it susceptible to flooding attacks, such as UDP flood or amplification attacks. Attackers can overwhelm a target by sending large volumes of UDP packets, causing denial of service. (["What is a UDP flood DDoS attack?," 2022](#))

Mitigation Measures

TCP: Session Hijacking can be mitigated through several key measures. Using SSL/TLS encryption secures data in transit, making it unreadable to attackers. Randomizing sequence numbers prevents prediction and unauthorized session control. Session timeouts help close inactive sessions, reducing exposure time. Lastly, multi-factor authentication adds an extra verification layer, making unauthorized access more difficult.

UDP: Flooding Attack can be reduced by applying rate limiting, using firewalls and IDS to block suspicious traffic, enabling application-level request validation, and deploying network-level anti-DDoS solutions to filter out attacks.

1.2. Network Security Mechanisms Against Common Network Attacks

Network security is crucial in today's interconnected digital world to secure personal information and sustain trustworthy communication. Cybercriminals utilize flaws in network setups and protocols to undertake illicit acts. This study explores three key sorts of network assaults: **STP attacks**, **DHCP spoofing**, and **Man-in-the-Middle** (MITM) attacks. It also explains how network security technologies can aid in resisting these risks. ([GeeksforGeeks, 2023](#))

Man-in-the-Middle (MITM) Attack

Attack Overview

An MITM is a general term where threat actor position himself between user and application either to eavesdrop or to impersonate one of other parties, making it appear as if a normal exchange of information is underway. Two instances of vulnerabilities that are misused include inadequate encryption and insufficient protocol authentication. ([Imperva, 2019](#))

Preventive Measures

1. Use robust encryption technologies like TLS/SSL.
2. Verify the identification of communication endpoints by using mutual authentication.
3. Use network segmentation and secure Wi-Fi settings.
4. To monitor anomalous traffic, install intrusion detection systems (IDS).

DHCP Spoofing Attack

Attack Overview

In a DHCP spoofing attack, an attacker sets up a rogue DHCP server on the network. The rogue server provides incorrect network configuration details (such as IP addresses or DNS servers) to clients, redirecting traffic for eavesdropping or further attacks. ([What Is DHCP Spoofing? How It Works & Examples | Twingate, 2020](#))

Preventive Measures

1. Set up network switches with DHCP snooping to filter unwanted DHCP packets.
2. Only set up network ports for trusted devices.
3. For sensitive systems, use static IP addressing whenever possible.
4. Monitor DHCP logs for anomalies.

STP (Spanning Tree Protocol) Attacks

Attack Overview

STP attacks target the Spanning Tree Protocol, which is used to prevent loops in network topologies. By sending malicious Bridge Protocol Data Units (BPDUs), an attacker can manipulate the STP process to take over the role of the root bridge, causing traffic redirection or network disruptions. ([Danielsekot, 2019](#))

Preventive Measures

1. Disable unused switch ports.
2. Use BPDU guard and root guard.
3. Configure STP parameters carefully.
4. Monitor the network for suspicious activity.

Demonstration of MITM

Objective: To demonstrate an ARP -based Man-in-the-Middle (MITM) attack using **Ettercap**. The goal is to intercept and analyze network packets exchanged between a target device and the gateway, showcasing how an attacker can eavesdrop on or manipulate communication within a local network.

Attacker IP : 192.168.1.80

Target IP : 192.168.1.82

Gateway : 192.168.1.254

Step 1: Scan a network to find targets.

```
[nott@honey in ~/tryhackme via * v8.4.7 via * v3.13.3 took 0s
└─$ sudo nmap -sn 192.168.1.0/24 -T4
[sudo] password for nott:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-17 01:16 +0545
Nmap scan report for 192.168.1.64
Host is up (0.0079s latency).
MAC Address: 3C:CD:57:94:D7:FC (Beijing Xiaomi Mobile Software)
Nmap scan report for 192.168.1.82
Host is up.
MAC Address: 5C:3A:45:8A:C3:B5 (Chongqing Fugui Electronics)
Nmap scan report for 192.168.1.254
Host is up (0.014s latency).
MAC Address: 04:75:F9:5A:93:B0 (Taicang T&W Electronics)
Nmap scan report for 192.168.1.80
Host is up.
Nmap done: 256 IP addresses (4 hosts up) scanned in 2.65 seconds

[nott@honey in ~/tryhackme via * v8.4.7 via * v3.13.3 as 🤖 took 5s
└─$ ]
```

The screenshot shows a terminal window displaying the output of an Nmap scan. The scan results are highlighted with red boxes and annotated with arrows pointing to specific entries:

- A red arrow labeled "Target Address" points to the entry for host 192.168.1.64, which is identified as "Beijing Xiaomi Mobile Software".
- A red arrow labeled "Gateway Address" points to the entry for host 192.168.1.254, which is identified as "Chongqing Fugui Electronics".
- A red arrow labeled "Attacker Address" points to the entry for host 192.168.1.80, which is identified as "Taicang T&W Electronics".

Figure 4: Network Scanning

Step 2: Launching ARP Spoofing using ettercap

```

[nott@honey ~] via v23.11.1 via v3.13.3 took 9s
[●] x sudo ettercap -T -i wlp3s0 -M arp:remote //192.168.1.82// //192.168.1.254// 
[sudo] password for nott:
ettercap 0.8.3.1 copyright 2001-2020 Ettercap Development Team

Listening on:
wlp3s0 -> 1C:CE:51:05:91:4A
    192.168.1.80/255.255.255.0
    fe80::b1b2:18e3:55c9:3d3f/64
    2400:1a00:b030:580:e782:ac2f:d0e5:d6d/64

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
Privileges dropped to EUID 65534 EGID 65534...

34 plugins
42 protocol dissectors
57 ports monitored
28230 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services

Scanning for merged targets (2 hosts)...
* |=====| 100.00 %

3 hosts added to the hosts list...

ARP poisoning victims:

GROUP 1 : 192.168.1.82 5C:3A:45:8A:C3:B5
GROUP 2 : 192.168.1.254 04:75:F9:5A:93:B0
Starting Unified sniffing...

Text only Interface activated...
Hit 'h' for inline help

Thu Jul 17 00:33:43 2025 [900418]
UDP 192.168.1.82:5353 --> 224.0.0.251:5353 | (43)
....._microsoft_mcc._tcp.local.....

```

The screenshot shows the terminal output of the ettercap command. Annotations highlight specific parts of the output:

- A red box highlights the command line: `[●] x sudo ettercap -T -i wlp3s0 -M arp:remote //192.168.1.82// //192.168.1.254//`. A red arrow points from this box to the text: "-T text mode -i interface -M arp poisoning <target_ip> <gateway_ip>".
- A red box highlights the "Listening on:" section. A red arrow points from this box to the text: "Listening Interface".
- A red box highlights the "ARP poisoning victims:" section. A red arrow points from this box to the text: "Start Sniffing".
- A red box highlights the bottom line of the output: `....._microsoft_mcc._tcp.local.....`. A red arrow points from this box to the text: "Packet Info".

Figure 5: ARP Spoofing and Packet spoofing

Step 3: Viewing target traffics in wireshark

Filtering IP address and HTTP protocol

No.	Time	Source	Destination	Protocol	Length	Info
1194	3.929238147	192.168.1.82	192.168.1.89	HTTP	489	GET /ftp_flag.txt HTTP/1.1
1210	3.958622659	192.168.1.80	192.168.1.82	HTTP	158	HTTP/1.0 304 Not Modified
10594	31.890983455	192.168.1.82	192.168.1.89	HTTP	489	GET /ftp_flag.txt HTTP/1.1
10596	31.891682152	192.168.1.80	192.168.1.82	HTTP	158	HTTP/1.0 304 Not Modified
23344	70.284038900	192.168.1.82	192.168.1.80	HTTP	396	GET / HTTP/1.1
23352	70.293680291	192.168.1.80	192.168.1.82	HTTP	1011	HTTP/1.0 200 OK (text/html)
62288	189.062419289	192.168.1.82	192.168.1.80	HTTP	443	GET /bricks.thm.txt HTTP/1.1
62304	189.088080153	192.168.1.80	192.168.1.82	HTTP	441	HTTP/1.0 200 OK (text/plain)
69674	289.190942252	192.168.1.82	192.168.1.80	HTTP	435	GET /hash.txt HTTP/1.1
69677	289.191717193	192.168.1.80	192.168.1.82	HTTP	593	HTTP/1.0 200 OK (text/plain)

Figure 6: Traffic Sniffing

Step 4: Analyzing target request files.

Packet ^	Hostname	Content Type	Size	Filename
23352	192.168.1.80	text/html	2,417 bytes	/
62304	192.168.1.80	text/plain	3,307 bytes	bricks.thm.txt
69677	192.168.1.80	text/plain	539 bytes	hash.txt
85716	ping.archlinux.org	text/plain	25 bytes	nm-check.txt
88657	ping.archlinux.org	text/plain	25 bytes	nm-check.txt
163539	192.168.1.80	text/plain	292 bytes	new.txt
187944	ping.archlinux.org	text/plain	25 bytes	nm-check.txt
190899	ping.archlinux.org	text/plain	25 bytes	nm-check.txt
293793	ping.archlinux.org	text/plain	25 bytes	nm-check.txt
296643	ping.archlinux.org	text/plain	25 bytes	nm-check.txt

Target requested Files

Figure 7: Target Interacted Files

Mitigation

45 2.538085033	192.168.1.80	142.250.193.129	QUIC	1399 Initial, DCID=dd5f9738e3c58bea00, SCID=3a5318, PKN: 0, CRYPTO
46 2.541179341	142.250.193.129	192.168.1.80	TLSv1.3	1466 Application Data
47 2.541205190	192.168.1.80	142.250.193.129	TCP	54 43726 -> 443 [ACK] Seq=1336 Ack=17276 Win=58880 Len=0
48 2.541600890	142.250.193.129	192.168.1.80	TLSv1.3	1466 Application Data
49 2.541621419	192.168.1.80	142.250.193.129	TCP	54 43726 -> 443 [ACK] Seq=1336 Ack=18688 Win=58880 Len=0
50 2.553550311	142.250.193.129	192.168.1.80	TLSv1.3	1466 Application Data
51 2.553582792	192.168.1.80	142.250.193.129	TCP	54 43726 -> 443 [ACK] Seq=1336 Ack=20100 Win=58880 Len=0
52 2.553945510	142.250.193.129	192.168.1.80	TLSv1.3	1466 Application Data
53 2.553956221	192.168.1.80	142.250.193.129	TCP	54 43726 -> 443 [ACK] Seq=1336 Ack=21512 Win=58880 Len=0

Figure 8: Mitigation Measure against MITM

2. Three-Tier Network Deployment

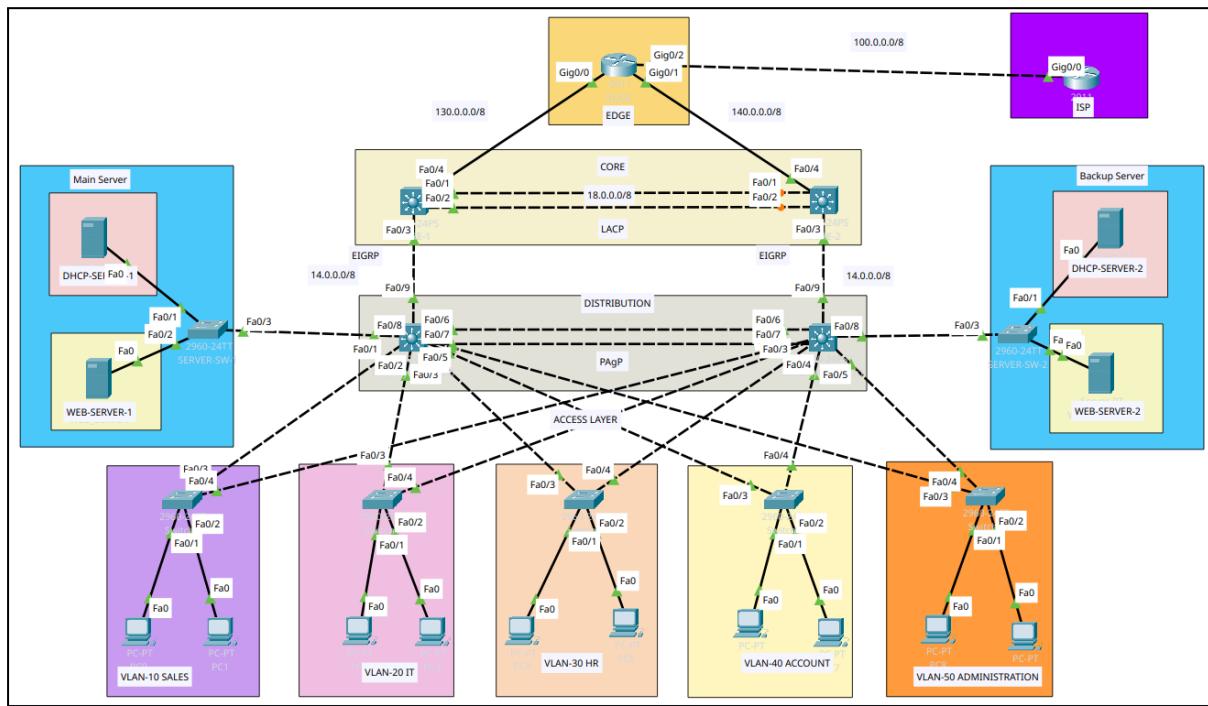


Figure 9: Three Tier Network Architecture

Subnetting and VLAN Allocation Table

VLAN NAME	NETWORK ADDRESS	SUBNET MASK	FIRST USABLE IP	LAST USABLE IP	BROADCAST ADDRESS	USABLE IPS	MAX DHCP USERS
VLAN-10_SALES	172.16.99.0	255.255.255.128 (/25)	172.16.99.1	172.16.99.126	172.16.99.127	126	125
VLAN-20_IT	172.16.99.128	255.255.255.128 (/25)	172.16.99.129	172.16.99.254	172.16.99.255	126	125
VLAN-30_HR	172.16.100.0	255.255.255.224 (/27)	172.16.100.1	172.16.100.30	172.16.100.31	30	29
VLAN-40_ACCOUNT	172.16.100.32	255.255.255.240 (/28)	172.16.100.33	172.16.100.46	172.16.100.47	14	13
VLAN-50_ADMINISTRATION	172.16.100.64	255.255.255.192 (/26)	172.16.100.65	172.16.100.126	172.16.100.127	62	61

2.1. Access Layer Switch Configuration (Layer 2 Switch)

In this layer five switches are configured for each five departments i.e, Sales, IT, HR, Account and Administration. Each switch are given unique VLANs id 10, 20, 30, 40, 50, according to five departments. Interface which are connected to end devices, configured as Access mode with respective VLANs and interface which are connected to distribution layer, configured as Trunk mode to carry multiple VLANs traffic.

ACCESS-1_bitter!honey (SALES)

ACCESS-1_bitter!honey(config)#			
ACCESS-1_bitter!honey(config)#do sh vlan bri			
VLAN	Name	Status	Ports
1	default	active	Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gig0/1, Gig0/2
10	SALES	active	Fa0/1, Fa0/2
20	IT	active	
30	HR	active	
40	ACCOUNT	active	
50	ADMINISTRATION	active	
100	SERVER-1	active	
1002	fdci-default	active	
1003	token-ring-default	active	
1004	fddinet-default	active	
1005	trnet-default	active	
ACCESS-1_bitter!honey(config)#			

Figure 10: VLAN 10 Sales

In this layer 2 switch, VLAN was configured with id 10 and name as SALES. Then assigned a port to carry single VLAN traffic which is using access mode. Similarly all the layer 2 switches with VLANs i.e IT, HR, Account and Administration are configured respectively.

Once the individual VLANs are configured on each switch, trunking is implemented on interconnecting links Fa0/1 and Fa0/2. Trunk link are network connecting configured to carry traffic from multiple VLANs simultaneously. These link use 802.1q encapsulation to insert VLAN tags to Ethernet frames, thus maintaining VLAN separation across the network.

```

ACCESS-1_bitter!honey(config)#
ACCESS-1_bitter!honey(config)#do sh int trunk
Port      Mode       Encapsulation  Status      Native vlan
Fa0/3    on        802.1q         trunking     1
Fa0/4    on        802.1q         trunking     1

Port      Vlans allowed on trunk
Fa0/3    1-1005
Fa0/4    1-1005

Port      Vlans allowed and active in management domain
Fa0/3    1,10,20,30,40,50,100
Fa0/4    1,10,20,30,40,50,100

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/3    10,20,30,40,50,100
Fa0/4    1,10,20,30,40,50,100

```

Allowed VLANs

Figure 11: Trunk VLAN 10 SALES

2.2. Distribution Layer Switch Configuration (Layer 3 Switch)

Trunk and VTP Configuration

The DIST-1_bitter!honey and DIST-2_bitter!honey switch connected to access layer switch via interface F0a/1-5. These multilayer switch are configured to support 802.1Q trunking and manage VLANs by joining to VTP domain name DINESH operating in version 2. This configuration ensures effective handling of VLAN traffic and maintains consistent VLAN information across the entire network.

```

DIST-1_bitter!honey(config)#do sh int trunk
Port      Mode       Encapsulation  Status      Native vlan
Fa0/1    on        802.1q        trunking   1
Fa0/2    on        802.1q        trunking   1
Fa0/3    on        802.1q        trunking   1
Fa0/4    on        802.1q        trunking   1
Fa0/5    on        802.1q        trunking   1

Port      Vlans allowed on trunk
Fa0/1    1-1005
Fa0/2    1-1005
Fa0/3    1-1005
Fa0/4    1-1005
Fa0/5    1-1005

Port      Vlans allowed and active in management domain
Fa0/1    1,10,20,30,40,50,100
Fa0/2    1,10,20,30,40,50,100
Fa0/3    1,10,20,30,40,50,100
Fa0/4    1,10,20,30,40,50,100
Fa0/5    1,10,20,30,40,50,100

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/1    1
Fa0/2    1
Fa0/3    1,10,20,30,40,50,100
Fa0/4    1
Fa0/5    1

```

Figure 12: trunk in DIST-1_bitter!honey

```

DIST-1_bitter!honey(config)#do sh vtp status
VTP Version capable      : 1 to 2
VTP version running      : 2
VTP Domain Name          : DINESH
VTP Pruning Mode         : Disabled
VTP Traps Generation     : Disabled
Device ID                 : 0001.4392.2600
Configuration last modified by 0.0.0.0 at 3-1-93 00:00:00
Local updater ID is 172.16.99.126 on interface Vl10 (lowest numbered VLAN interface found)

Feature VLAN :
-----
VTP Operating Mode       : Server
Maximum VLANs supported locally : 1005
Number of existing VLANs    : 11
Configuration Revision     : 117
MD5 digest                : 0x4E 0x34 0x60 0x01 0x37 0x34 0x99 0xF0
                             0x63 0xD3 0x17 0xBD 0x84 0xD8 0x0E 0x02

DIST-1_bitter!honey(config)#

```

Figure 13: VTP Status

Etherchannel Configuration using PAgP in Distribution layer

```
DIST-1_bitter!honey(config)#  
DIST-1_bitter!honey(config)#do sh etherchannel su  
Flags: D - down P - 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  
u - unsuitable for bundling  
w - waiting to be aggregated  
d - default port  
  
Number of channel-groups in use: 1  
Number of aggregators: 1  
  
Group Port-channel Protocol Ports  
-----+-----+-----+-----  
1 Po1(SU) PAgP Fa0/6(P) Fa0/7(P)  
DIST-1_bitter!honey(config)#
```

Figure 14: Etherchannel Status in Distribution layer

Next, EtherChannel is set up between the DIST-1_bitter!honey and DIST-2_bitter!honey switches by bundling interfaces Fa0/6 and Fa0/7. The configuration uses PAgP, with auto mode on one switch and desirable mode on the other, allowing the EtherChannel to form dynamically. The primary benefits of implementing etherchannel to maintain redundancy and load balancing to prevent from fail over.

Configuration Default Gateways (SVIs)

```
DIST-1_bitter!honey(config)#  
DIST-1_bitter!honey(config)#do sh ip int bri | exclude un  
Interface IP-Address OK? Method Status Protocol  
FastEthernet0/8 10.0.0.1 YES manual up up  
FastEthernet0/9 14.0.0.1 YES manual up up  
Vlan10 172.16.99.126 YES manual up up  
Vlan20 172.16.99.254 YES manual up up  
Vlan30 172.16.100.30 YES manual up up  
Vlan40 172.16.100.46 YES manual up up  
Vlan50 172.16.100.110 YES manual up up  
DIST-1_bitter!honey(config)#
```

Figure 15: VLANs Default Gateways

On DIST-1_bitter!honey, IP addresses are assigned to each VLAN interface using VLSM, ensuring that each department receives just enough addresses for its host requirements. These

IPs act as the default gateways for the devices within each department. The Fa0/8 port is specifically set up as the gateway for the DHCP and web servers, while Fa0/9 connects the switch to the core network. A similar setup is implemented on DIST-2_bitter!honey, with the only variation being the use of a different server gateway address.

```
interface Vlan10
  mac-address 0060.4794.b801
  ip address 172.16.99.126 255.255.255.128
  ip helper-address 10.0.0.2
  ip helper-address 11.0.0.2
!
interface Vlan20
  mac-address 0060.4794.b802
  ip address 172.16.99.254 255.255.255.128
  ip helper-address 10.0.0.2
  ip helper-address 11.0.0.2
!
interface Vlan30
  mac-address 0060.4794.b803
  ip address 172.16.100.30 255.255.255.224
  ip helper-address 10.0.0.2
  ip helper-address 11.0.0.2
!
interface Vlan40
  mac-address 0060.4794.b804
  ip address 172.16.100.46 255.255.255.240
  ip helper-address 10.0.0.2
  ip helper-address 11.0.0.2
!
interface Vlan50
  mac-address 0060.4794.b805
  ip address 172.16.100.110 255.255.255.192
  ip helper-address 10.0.0.2
  ip helper-address 11.0.0.2
```

Figure 16: VLANs Interface Configuration

In addition to assigning IP addresses to VLAN interfaces, IP helper-addresses are set up to function as **DHCP relay agents**, enabling devices within each VLAN to reach internal servers across different subnets. In this configuration, 10.0.0.2 serves as the primary helper address, while 11.0.0.2 acts as the secondary. An identical setup has been applied on DIST-2_bitter!honey to maintain consistent DHCP relay support across both distribution switches.

Configuration of Routing Protocol

```

DIST-1_bitter!honey#sh ip route
Codes: C - connected S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    10.0.0.0/8 is directly connected, FastEthernet0/8
D    11.0.0.0/8 [90/35840] via 14.0.0.2, 03:01:38, FastEthernet0/9
C    14.0.0.0/8 is directly connected, FastEthernet0/9
D    18.0.0.0/8 [90/30720] via 14.0.0.2, 03:01:38, FastEthernet0/9
D    100.0.0.0/8 [90/30976] via 14.0.0.2, 03:01:38, FastEthernet0/9
D    130.0.0.0/8 [90/30720] via 14.0.0.2, 03:01:38, FastEthernet0/9
D    140.0.0.0/8 [90/33280] via 14.0.0.2, 03:01:38, FastEthernet0/9
      172.16.0.0/16 is variably subnetted, 5 subnets, 4 masks
C      172.16.99.0/25 is directly connected, Vlan10
C      172.16.99.128/25 is directly connected, Vlan20
C      172.16.100.0/27 is directly connected, Vlan30
C      172.16.100.32/28 is directly connected, Vlan40
C      172.16.100.64/26 is directly connected, Vlan50

```

Figure 17: Routing Protocol Status in Distribution layer

DIST-1_bitter!honey uses the EIGRP routing protocol to manage big networks in a smart way. It has directly connected networks in its routing table, like 10.0.0.0/8 and 14.0.0.0/8. It learns about new routes, like 11.0.0.0/8, 18.0.0.0/8, 100.0.0.0/8, 130.0.0.0/8, and 140.0.0.0/8, from a neighbor at 14.0.0.2. There are also different subnets in the 172.16.0.0/16 network. DIST-2_bitter!honey uses a similar EIGRP setup to make sure that routing is always the same on the network.

2.3. Core Layer Switch Configuration

Etherchannel Configuration using LACP

```
CORE-1_bitter!honey(config)#do sh etherchannel su
Flags:  D - down          P - 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
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port

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

Group  Port-channel  Protocol      Ports
-----+-----+-----+
2      Po2(RU)       LACP          Fa0/1(P) Fa0/2(P)
```

Figure 18: Core layer etherchannel configuration

Both CORE-1_bitter!honey and CORE-2_bitter!honey are configured to bundle their Fa0/1 and Fa0/2 interfaces into a single EtherChannel using the Link Aggregation Control Protocol (LACP), with both sides set to active mode to ensure dynamic negotiation and link redundancy.

Configuration IP Addresses

```
CORE-1_bitter!honey(config)#do sh ip int bri | ex un
Interface          IP-Address      OK? Method Status      Protocol
Port-channel2      18.0.0.1        YES manual up        up
FastEthernet0/3     14.0.0.2        YES manual up        up
FastEthernet0/4     130.0.0.1       YES manual up        up
CORE-1_bitter!honey(config)#
```

Figure 19: Configure IP Addresses in Core layer

CORE-1_bitter!honey switch, interface Po2, Fa0/3 and Fa0/4 are configured to assign IP addresses to forward traffic and IP routing. In layer 3 switch to assign IPs need to disable

switchport mode, all the interfaces are disabled with switchport mode. And then ip routing was enabled. Afterward IP addresses are assigned to IP addresses.

Configuring Routing Protocol in Core layer

```
CORE-1_bitter!honey(config)#do sh ip route
codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

D  10.0.0.0/8 [90/30720] via 14.0.0.1, 03:42:22, FastEthernet0/3
D  11.0.0.0/8 [90/33280] via 18.0.0.2, 03:42:22, Port-channel2
C  14.0.0.0/8 is directly connected, FastEthernet0/3
C  18.0.0.0/8 is directly connected, Port-channel2
D  100.0.0.0/8 [90/28416] via 130.0.0.2, 03:42:22, FastEthernet0/4
C  130.0.0.0/8 is directly connected, FastEthernet0/4
D  140.0.0.0/8 [90/30720] via 130.0.0.2, 03:42:22, FastEthernet0/4
      [90/30720] via 18.0.0.2, 03:42:22, Port-channel2
    172.16.0.0/16 is variably subnetted, 5 subnets, 4 masks
D    172.16.99.0/25 [90/25628160] via 14.0.0.1, 03:42:22, FastEthernet0/3
D    172.16.99.128/25 [90/25628160] via 14.0.0.1, 03:42:22, FastEthernet0/3
D    172.16.100.0/27 [90/25628160] via 14.0.0.1, 03:42:22, FastEthernet0/3
D    172.16.100.32/28 [90/25628160] via 14.0.0.1, 03:42:22, FastEthernet0/3
D    172.16.100.64/26 [90/25628160] via 14.0.0.1, 03:42:22, FastEthernet0/3
```

Figure 20: IP routing in Core layer

CORE-1_bitter!honey uses the EIGRP routing protocol to manage big networks in a smart way. It has directly connected networks in its routing table, like 10.0.0.0/8 and 14.0.0.0/8. It learns about new routes, like 11.0.0.0/8, 18.0.0.0/8, 100.0.0.0/8, 130.0.0.0/8, and 140.0.0.0/8, from a neighbor at 14.0.0.2. There are also different subnets in the 172.16.0.0/16 network. CORE-2_bitter!honey uses a similar EIGRP setup to make sure that routing is always the same on the network.

2.4. Configuring EDGE and ISP (Router)

Configuring IP Addresses in EDGE and ISP

EDGE_ROUTER_bitter!honey(config)#do sh ip int bri ex un				
Interface	IP-Address	OK?	Method	Status
GigabitEthernet0/0	130.0.0.2	YES	manual	up
GigabitEthernet0/1	140.0.0.2	YES	manual	up
GigabitEthernet0/2	100.0.0.1	YES	manual	up

Figure 21: EDGE_ROUTER_bitter!honey IP Addresses Configuration

The EDGE_ROUTER_bitter!honey connects the internal network to the internet. Interfaces Gig0/0 and Gig0/1 are linked to the internal network through the core layer, while Gig0/2 connects to the ISP_bitter!honey for external traffic. All interfaces are active with assigned IPs for proper routing.

ISP_bitter!honey(config)#do sh ip int br ex un				
Interface	IP-Address	OK?	Method	Status
GigabitEthernet0/0	100.0.0.2	YES	manual	up

Figure 22: ISP_bitter!honey IP Address Configuration

The ISP_bitter!honey router has one active interface, GigabitEthernet0/0, with IP 100.0.0.2. It is manually configured and fully operational. This interface connects to the EDGE_ROUTER_bitter!honey, providing internet access to the internal network.

Configuring Routing Protocol in EDGP and ISP

```
ISP_bitter!honey(config)#
ISP_bitter!honey(config)#do sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

D    10.0.0.0/8 [90/33536] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
D    11.0.0.0/8 [90/33536] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
D    14.0.0.0/8 [90/30976] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
D    18.0.0.0/8 [90/30976] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
      100.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C      100.0.0.0/8 is directly connected, GigabitEthernet0/0
L      100.0.0.2/32 is directly connected, GigabitEthernet0/0
D    130.0.0.0/8 [90/5376] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
D    140.0.0.0/8 [90/5376] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
      172.16.0.0/16 is variably subnetted, 5 subnets, 4 masks
D      172.16.99.0/25 [90/25630976] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
D      172.16.99.128/25 [90/25630976] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
D      172.16.100.0/27 [90/25630976] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
D      172.16.100.32/28 [90/25630976] via 100.0.0.1, 05:03:00, GigabitEthernet0/0
D      172.16.100.64/26 [90/25630976] via 100.0.0.1, 05:03:00, GigabitEthernet0/0

ISP_bitter!honey(config)#

```

Figure 23: Routing Protocol of ISP_bitter!honey

To facilitate smooth routing, data forwarding, and inter-VLAN operations, the show ip route command on ISP_bitter!honey displays a combination of directly connected routes and EIGRP-learned paths. For reliable and effective network communication, EDGE_ROUTER_bitter!honey is set up similarly.

```
C:\>ping 100.0.0.2 ← ISP IP Address
Pinging 100.0.0.2 with 32 bytes of data:
Reply from 100.0.0.2: bytes=32 time<1ms TTL=252
Ping statistics for 100.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figure 24: Pinging ISP from end device.

2.5. Configuration of Internal Servers

DHCP										
Interface	FastEthernet0	Service	<input checked="" type="radio"/> On	<input type="radio"/> Off						
Pool Name	serverPool									
Default Gateway	0.0.0.0									
DNS Server	0.0.0.0									
Start IP Address :	10	0	0	0						
Subnet Mask:	255	0	0	0						
Maximum Number of Users :	512									
TFTP Server:	0.0.0.0									
WLC Address:	0.0.0.0									
Add		Save			Remove					
Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max User	TFTP Server	WLC Address			
VLAN-50_ADMINISTR...	172.16.100.110	8.8.8.8	172.16.100.65	255.255.255....	61	0.0.0.0	0.0.0.0			
VLAN-40_ACCOUNT	172.16.100.46	8.8.8.8	172.16.100.33	255.255.255....	14	0.0.0.0	0.0.0.0			
VLAN-30_HR	172.16.100.30	8.8.8.8	172.16.100.1	255.255.255....	30	0.0.0.0	0.0.0.0			
VLAN-20_IT	172.16.99.254	8.8.8.8	172.16.99.129	255.255.255....	126	0.0.0.0	0.0.0.0			
VLAN-10_SALES	172.16.99.126	8.8.8.8	172.16.99.1	255.255.255....	126	0.0.0.0	0.0.0.0			

Figure 25: DHCP Configuration

DHCP (Dynamic Host Configuration Protocol) automatically assigns IP addresses and network settings to devices, eliminating manual configuration and reducing errors. In our design, the DHCP server is configured with five pools, one per departmental VLAN using VLSM-derived subnets from the 172.16.99.0/16 block. This ensures efficient IP utilization and consistent gateway and DNS settings for Sales (VLAN 10), IT (VLAN 20), , HR (VLAN 30), Account (VLAN 40) and Administration (VLAN 50).

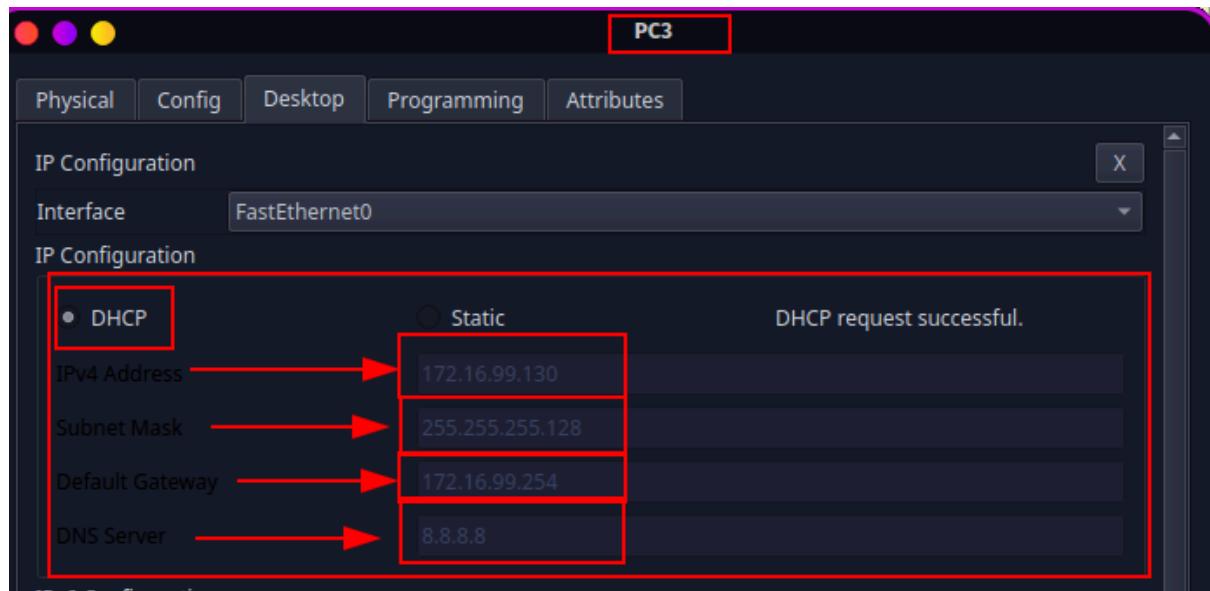


Figure 26: Dynamically assigned IP to PC3 via DHCP Server

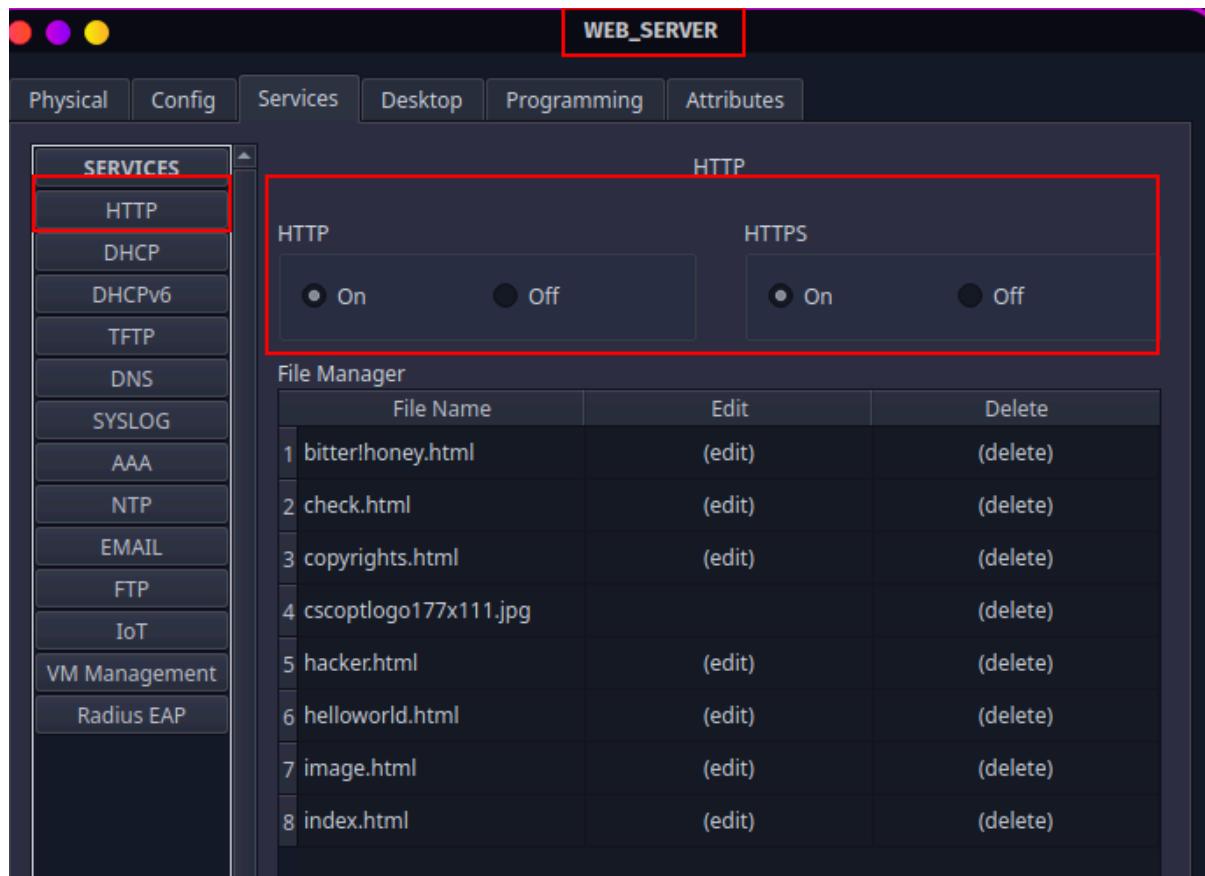


Figure 27: Web Server

The web server is properly configured and currently operational, providing access to all departments. It supports both HTTP and HTTPS protocols. Identical configurations are applied to the backup web server to ensure consistency and reliability.

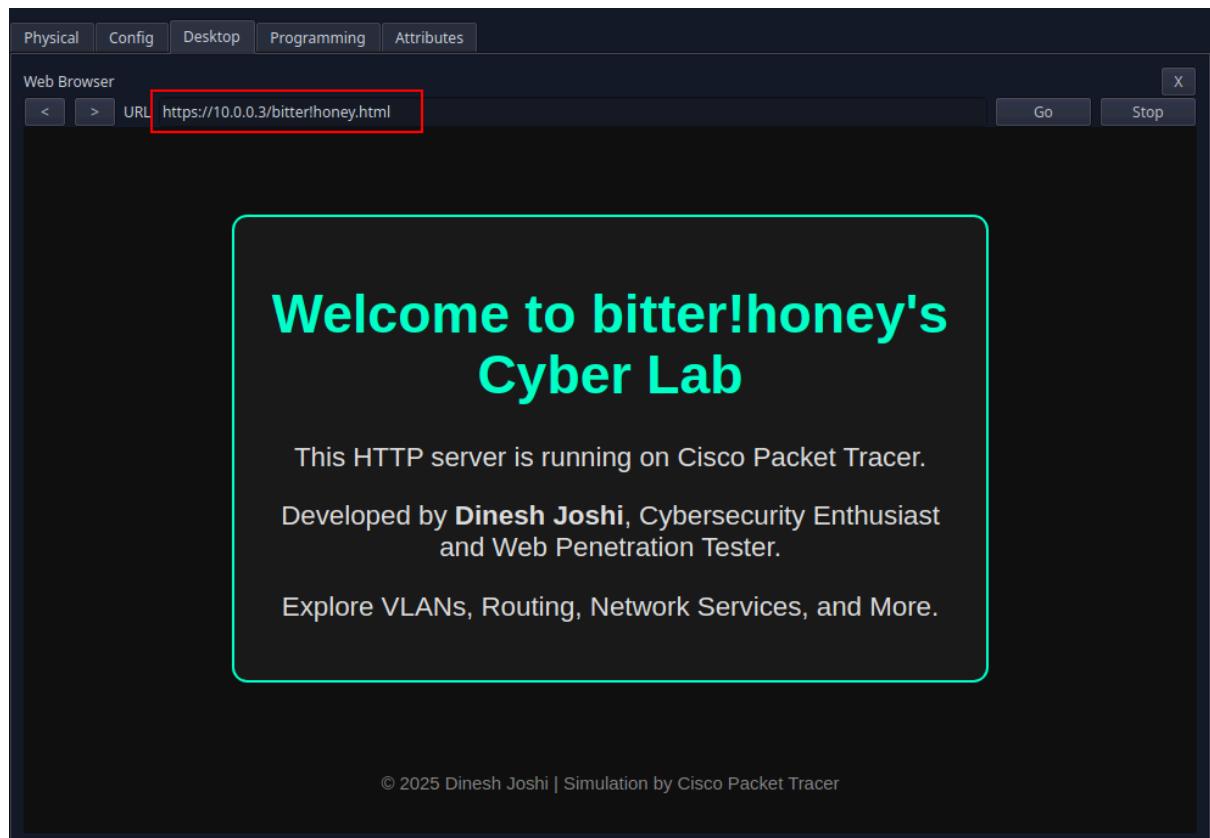


Figure 28: Accessing Web Page

3. Secure Network Deployment

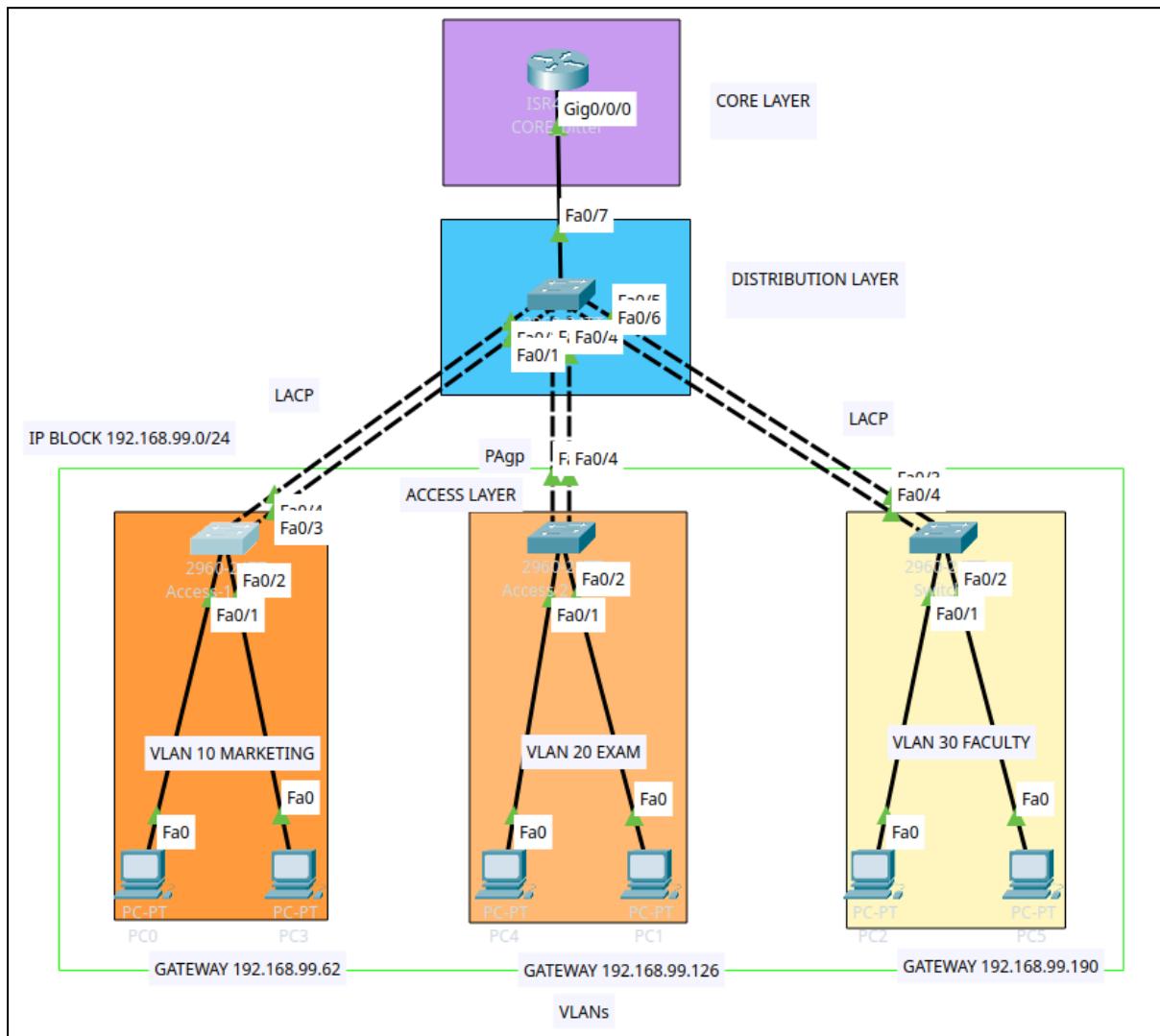


Figure 29: Network Architecture

The three-tier network topology is implemented to provide scalable, modular and resilient network design that separate functions across different layers. At the core layer (ISR4331) router interface gig0/0/0 provides a gateway to the external network. The middle distribution layer features a Layer 3 switch that terminates the core link, enforces policies, and performs inter-VLAN routing. Finally, the access layer comprises three 2960 switches, each dedicated to a single departmental VLAN Marketing, Exam, and Faculty with each VLAN in its own /26 subnet and the distribution switch's SVI as its default gateway. By separating core

routing, distribution policy management, and user-facing access, this design optimizes performance, simplifies debugging, and maintains uninterrupted connectivity.

3.1. Assigning Hostname and Description to Interface

```
ACCESS-SW-1_bitter!honey(config)#hostname ACCESS-SW-1_bitter!honey
ACCESS-SW-1_bitter!honey(config)#
ACCESS-SW-1_bitter!honey(config)#
ACCESS-SW-1_bitter!honey(config-if-range)#int r f0/1-2
ACCESS-SW-1_bitter!honey(config-if-range)#description "CONNECTED TO THE END DEVICES"
ACCESS-SW-1_bitter!honey(config-if-range)#
ACCESS-SW-1_bitter!honey(config-if-range)#int r f0/3-4
ACCESS-SW-1_bitter!honey(config-if-range)#description "CONNECTED TO DISTRIBUTION LAYER"
ACCESS-SW-1_bitter!honey(config-if-range)#
ACCESS-SW-1_bitter!honey(config-if-range)#

```

```
DIST-SW_bitter!honey(config-if-range)#hostname DIST-SW_bitter!honey
DIST-SW_bitter!honey(config)#
DIST-SW_bitter!honey(config-if-range)#int r f0/1-6
DIST-SW_bitter!honey(config-if-range)#description "CONNECTED TO ACCESS LAYER"
DIST-SW_bitter!honey(config-if-range)#

```

```
Current configuration : 2384 bytes
!
version 15.4
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname CORE_bitter!honey
!
!
```

```

!
interface GigabitEthernet0/0/0
description "CONNECTED TO DIST-SW-bitter!honey"
no ip address
duplex auto
speed auto
!
interface GigabitEthernet0/0/0.10
description "DEFAULT GATEWAY FOR VLAN10"
encapsulation dot1Q 10
ip address 192.168.99.62 255.255.255.192
!
interface GigabitEthernet0/0/0.20
description "DEFAULT GATEWAY FOR VLAN20"
encapsulation dot1Q 20
ip address 192.168.99.126 255.255.255.192
!
interface GigabitEthernet0/0/0.30
description "DEFAULT GATEWAY FOR VLAN30"
encapsulation dot1Q 30
ip address 192.168.99.190 255.255.255.192
!
```

Figure 30: Assigned Hostname and Description to Interface

Each device in the network was allocated a descriptive hostname, such as CORE_bitter!honey, to ensure that logs, configurations, and monitoring outputs are easily recognized. Additionally, interface descriptions were applied to all active ports, explicitly stating their purpose and the linked segment. This organized naming and annotation method greatly enhances network visibility, troubleshooting efficiency, and change management, especially in larger or layered network settings.

3.2. Assigning IP Address

Interface	IP-Address	OK?	Method	Status	Protocol
GigabitEthernet0/0/0.10	192.168.99.62	YES	manual	up	up
GigabitEthernet0/0/0.20	192.168.99.126	YES	manual	up	up
GigabitEthernet0/0/0.30	192.168.99.190	YES	manual	up	up

Figure 31: Assigning IP Address

All devices on the 192.168.99.0/24 network, which was divided into smaller subnets for the VLAN of each department, were assigned IP addresses. In order to facilitate inter-VLAN

communication and preserving organized and segregated traffic between departments, router subinterfaces were configured.

3.3. Creating User Account with Secure Password

```
!
hostname CORE_bitter!honey
!
!
!
enable secret 5 $1$mERr$MSCsGn1R8P02jnd.nLgVl/
!
!
!
!
!
ip cef
no ipv6 cef
!
!
!
username ADMIN privilege 15 secret 5 $1$mERr$tKAvQPi4nwql8vUTe8ogt/
username operator privilege 7 secret 5 $1$mERr$tKAvQPi4nwql8vUTe8ogt/
username user privilege 2 secret 5 $1$mERr$tKAvQPi4nwql8vUTe8ogt/
!
```

Figure 32: Creating User with secure password

Three user are created in CORE_bitter!honey and other devices as well with different privilege level. This helps maintain operational integrity while enabling team collaboration.

3.4. Implementing Telnet and SSH with Local Credentials

```
CORE_bitter!honey(config-line)#ip domain-name dinesh.com
CORE_bitter!honey(config)#crypto key generate rsa general-keys modulus 2048
% You already have RSA keys defined named CORE_bitter!honey.dinesh.com
% They will be replaced.

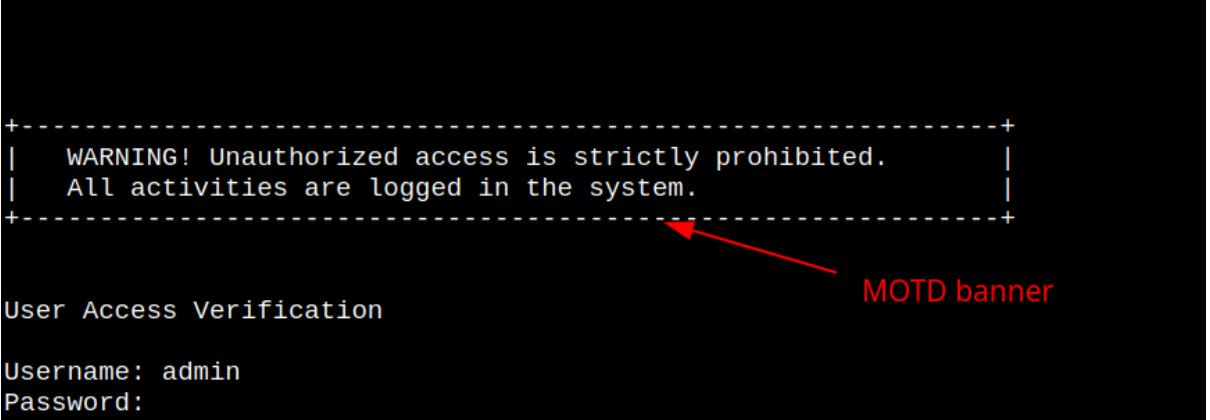
% The key modulus size is 2048 bits
% Generating 2048 bit RSA keys, keys will be non-exportable...[OK]
*Mar 2 2:58:17.373: %SSH-5-ENABLED: SSH 2 has been enabled
CORE_bitter!honey(config)#
CORE_bitter!honey(config)#ip ssh version 2
CORE_bitter!honey(config)#ip ssh time-out 60
CORE_bitter!honey(config)#ip ssh authentication-retries 2
CORE_bitter!honey(config)#
CORE_bitter!honey(config)#line vty 0 15
CORE_bitter!honey(config-line)#login local
CORE_bitter!honey(config-line)#transport input ssh
CORE_bitter!honey(config-line)#transport input telnet
CORE_bitter!honey(config-line)#exec-timeout 10 0
CORE_bitter!honey(config-line)#logging synchronous
CORE_bitter!honey(config-line)#

```

Figure 33: Implementing SSH and Telnet

On the VTY lines, both SSH and Telnet remote-access techniques were enabled; however, SSH version 2 was required to ensure encrypted sessions. To ensure that login credentials are centrally managed on each switch and router, devices authenticate using the local user database. In order to prevent unwanted local access, console access also requires login authentication.

3.5. Configuring MOTD



A screenshot of a terminal window on a black background. At the top, there is a dashed rectangular box containing the text:

```
+-----+  
| WARNING! Unauthorized access is strictly prohibited. |  
| All activities are logged in the system. |  
+-----+
```

A red arrow points from the text "MOTD banner" to the top section of the terminal output. Below this, the text "User Access Verification" is displayed. At the bottom, there are two lines for entering credentials:

```
Username: admin  
Password:
```

Figure 34: MOTD banner

A standardized MOTD banner was configured on every device to warn unauthorized users that access is monitored and strictly prohibited. This legal notice serves to remind administrators of the significance of using the correct login credentials, discourage casual intruders, and fortify the organization's position in the event of unauthorized access.

3.6. Difference Between SSH and Telnet

Feature	Telnet	SSH (Secure Shell)
Protocol & Port	TCP port 23; communicates in clear text	TCP port 22; uses encrypted communication
Security & Encryption	No encryption; data and credentials are exposed	Uses strong encryption (symmetric + asymmetric)
Authentication	Plain-text passwords sent over the network	Supports password-based and key-based authentication
Session Capabilities	Basic terminal access only	Offers remote shell, SFTP, port forwarding, etc.
Vulnerabilities	Susceptible to packet sniffing and MITM attacks	Resistant to attacks when using SSHv2 and secure ciphers
Additional Capabilities	Supports secure features like SCP, SFTP, and port forwarding	Limited to basic terminal access only

```
C:\>
C:\>
C:\>telnet 192.168.99.126
Trying 192.168.99.126 ...Open
+-----+
|  WARNING! Unauthorized access is strictly prohibited.      |
|  All activities are logged in the system.                  |
+-----+ MOTD banner

User Access Verification

Username: admin
Password:
CORE_bitter!honey#
CORE_bitter!honey#en
CORE_bitter!honey#configt
Translating "configt"...domain server (255.255.255.255) % Name lookup aborted
CORE_bitter!honey#conf t
Enter configuration commands, one per line. End with CNTL/Z.
CORE_bitter!honey(config)#do sh ip int bri | ex un
Interface          IP-Address      OK? Method Status      Protocol
GigabitEthernet0/0/0.10192.168.99.62  YES NVRAM   up
GigabitEthernet0/0/0.20192.168.99.126 YES NVRAM   up
GigabitEthernet0/0/0.30192.168.99.190  YES NVRAM   up
CORE_bitter!honey(config)#
```

Figure 35: Telnet Login

```
C:\>
C:\>ssh -l admin 192.168.99.126 ← login as admin
Password:

+-----+
|  WARNING! Unauthorized access is strictly prohibited.      |
|  All activities are logged in the system.                  |
+-----+ MOTD banner

CORE_bitter!honey#sh ip int bri | ex un
Interface          IP-Address      OK? Method Status      Protocol
GigabitEthernet0/0/0.10192.168.99.62  YES manual  up
GigabitEthernet0/0/0.20192.168.99.126 YES manual  up
GigabitEthernet0/0/0.30192.168.99.190  YES manual  up
CORE_bitter!honey#
```

Figure 36: SSH Login

3.7. Implementing Encryption Techniques

```
CORE_bitter!honey(config)#enable secret 0 dinesh
CORE_bitter!honey(config)#
CORE_bitter!honey(config)#
CORE_bitter!honey(config)#service password-encryption
CORE_bitter!honey(config)#
CORE_bitter!honey(config)#
```

Figure 37: Encrypted Password

3.8. Implementing RSTP

```
DIST-SW bitter!honey(config)#do sh spanning-tree su
Switch is in rapid-pvst mode
Root bridge for: default MARKETING EXAM FACULTY
Extended system ID          is enabled
Portfast Default            is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default          is disabled
EtherChannel misconfig guard is disabled
UplinkFast                  is disabled
BackboneFast                is disabled
Configured Pathcost method used is short

-----
```

Name	Blocking	Listening	Learning	Forwarding	STP	Active
VLAN0001	7	0	0	3	10	
VLAN0010	6	0	0	4	10	
VLAN0020	6	0	0	4	10	
VLAN0030	6	0	0	4	10	
4 vlans	25	0	0	15	40	

Figure 38: RSPT Implementation

Rapid-PVST is used by the switch to prevent loops and to accelerate convergence in each VLAN. It is the root bridge for VLANs 1, 10, 20, and 30 and has 40 active STP ports. Some of these ports forward traffic, while others are blocked to prevent loops. RSTP ensures speedy recovery during network disruptions by avoiding the delays connected with regular STP. For improved security, it is recommended to turn on technologies like PortFast and BPDU Guard to protect against STP-related threats and unauthorized devices.

3.9. Configuring Etherchannel

```
ACCESS-SW-1_bitter!honey#sh etherchannel su
Flags: D - down      P - 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
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port

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

Group  Port-channel  Protocol    Ports
-----+-----+-----+
1      Po1(SU)      LACP        Fa0/3(P)  Fa0/4(P)
```

```
DIST-SW_bitter!honey(config)#do sh etherchannel su
Flags: D - down      P - 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
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port

Number of channel-groups in use: 3
Number of aggregators: 3

Group  Port-channel  Protocol    Ports
-----+-----+-----+
1      Po1(SU)      LACP        Fa0/1(P)  Fa0/2(P)
2      Po2(SU)      PAgP        Fa0/3(P)  Fa0/4(P)
3      Po3(SU)      LACP        Fa0/5(P)  Fa0/6(P)
```

Figure 39: Configuration of Etherchannel

Etherchannel offers link redundancy and enhances bandwidth by integrating numerous physical Ethernet connections into a single logical interface. Using load-balancing methods that may be altered, it distributes traffic among member links as effectively as feasible. In the event of a link breakdown, it rapidly switches over to provide high availability.

3.10. Disabling DTP

```
DIST-SW_bitter!honey(config-if-range)#int r f0/1-7  
DIST-SW_bitter!honey(config-if-range)#switchport mode trunk  
DIST-SW_bitter!honey(config-if-range)#{  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up  
%LINK-3-UPDOWN: Interface Port-channel1, changed state to down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface Port-channel1, changed state to down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up  
%LINK-3-UPDOWN: Interface Port-channel2, changed state to down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface Port-channel2, changed state to down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/5, changed state to down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/5, changed state to up  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/6, changed state to down  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/6, changed state to up  
%LINK-5-CHANGED: Interface Port-channel2, changed state to up  
%LINEPROTO-5-UPDOWN: Line protocol on Interface Port-channel2, changed state to up  
DIST-SW_bitter!honey(config-if-range)#{  
DIST-SW_bitter!honey(config-if-range)#{  
%LINK-5-CHANGED: Interface Port-channel1, changed state to up  
%LINEPROTO-5-UPDOWN: Line protocol on Interface Port-channel1, changed state to up
```

Figure 40: Turning off DTP

```

interface FastEthernet0/4
description "CONNECTED TO DISTRIBUTION LAYER"
switchport mode trunk
switchport nonegotiate
channel-group 1 mode active
!
interface FastEthernet0/5
switchport mode trunk
switchport nonegotiate
!
interface FastEthernet0/6
switchport mode trunk
switchport nonegotiate
!
interface FastEthernet0/7
switchport mode trunk
switchport nonegotiate
!
ACCESS-SW-1_bitter!honey#

```

Figure 41: Disabled DTP

Dynamic Trunking Protocol (DTP) is used by Cisco switches to make trunk connections automatically. By avoiding VLAN-hopping and unauthorized trunking, administrators can increase network security by disabling DTP and making sure that trunk links are only manually set.

4. Network Security Practices and Technologies in Linux and Enterprise Environments

4.1. IPtables Configuration

iptables is a command-line firewall utility in Linux used to configure the kernel's **netfilter** framework. It manages incoming and outgoing traffic using rules that specify what to do with packets (ACCEPT, DROP, etc.). ([What Is Network Security? Definition and Types | Fortinet, 2022](#))

Set Default Policies

```
(bitternot@honey)-[~]$ sudo iptables -P INPUT DROP
(bitternot@honey)-[~]$ sudo iptables -P FORWARD DROP
(bitternot@honey)-[~]$ sudo iptables -P OUTPUT ACCEPT
(bitternot@honey)-[~]$
```

Figure 42: Setting Default Policies

Set Inbound Rules

```
(bitternot@honey)-[~]$ sudo iptables -A INPUT -p tcp --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
(bitternot@honey)-[~]$ sudo iptables -A OUTPUT -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT
```

Figure 43: Setting up Inbound Rules

Set Outbound Rules

```
(bitternot@honey):~$ sudo iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT  
(bitternot@honey):~$ sudo iptables -A OUTPUT -p tcp --dport 22 -j ACCEPT
```

Figure 44: Setting up Outbound Rules

Enable Logs

```
(bitternot@honey):~$ sudo iptables -A INPUT -p tcp --dport 80 -j LOG --log-prefix "HTTP BLOCKED: "  
(bitternot@honey):~$ sudo iptables -A INPUT -p tcp --dport 80 -j DROP
```

Figure 45: Enabling logs

Best Practices for Firewall Configuration

The ideal way to set up a firewall is to allow only the most necessary connections and utilize the basic settings that halt all traffic. The rules for your firewall should be looked over and adjusted often to keep up with the current security needs. By permitting logging, you can maintain an eye on actions that seem unusual, and fail2ban and other tools can give further safety. It is also necessary to execute stateful inspection, protect access to the firewall, and keep track of any rule revisions in order to maintain a network environment safe and under control.

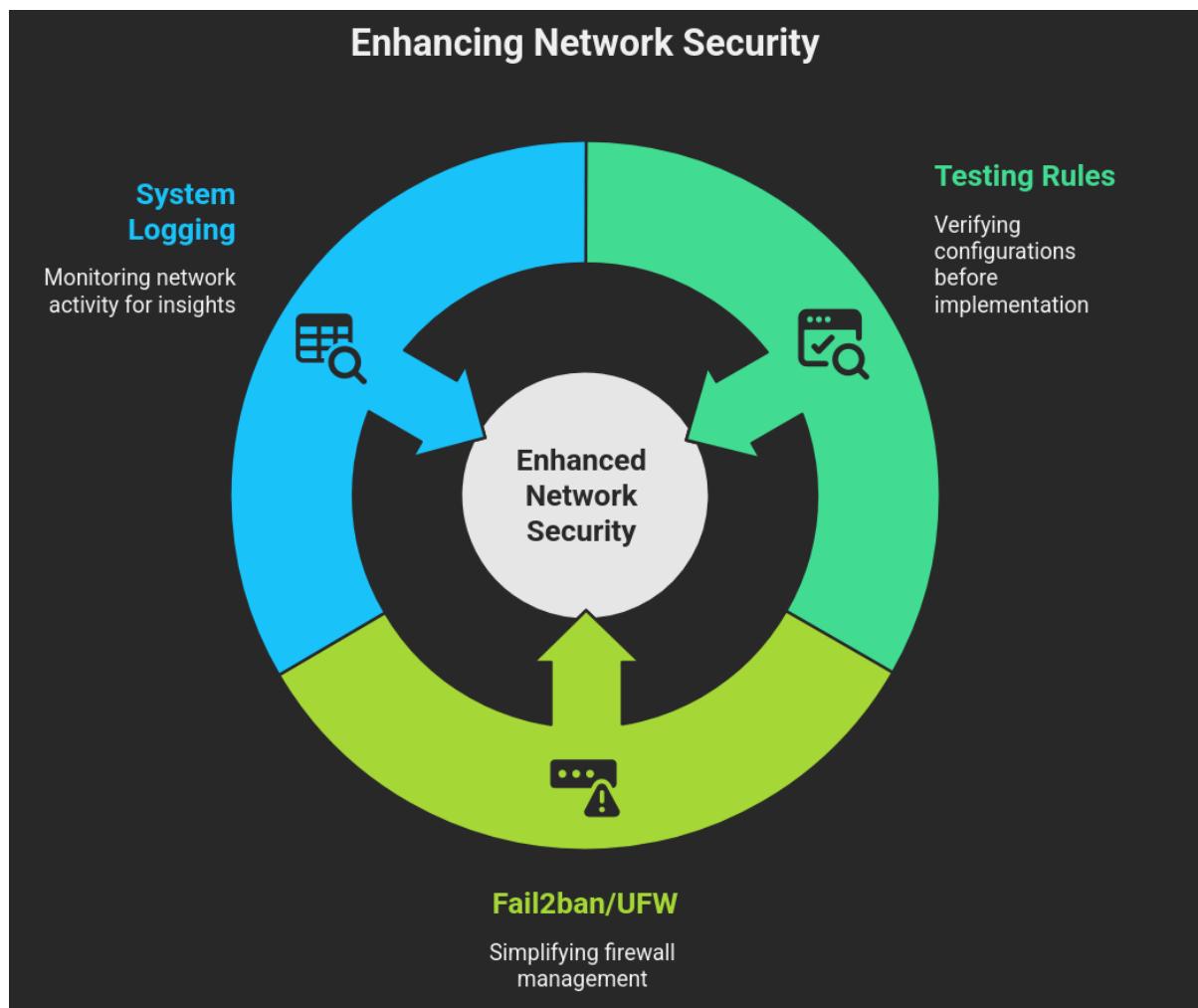


Figure 46: Best Practice

iptables Configuration Showcase

```
(bitternot@honey)-[~]
$ netstat -tuln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp      0      0 0.0.0.0:22               0.0.0.0:*              LISTEN
tcp      0      0 0.0.0.0:80               0.0.0.0:*              LISTEN
tcp6     0      0 :::22                  ::::*                  LISTEN
```

Figure 47: Opening Ports

The SSH connection is successful because the iptable rules allow inbound SSH traffic.

```
(bitternot@honey)-[/var/log]
$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST  mtu 1500
  inet 172.17.0.2  netmask 255.255.0.0  broadcast 172.17.255.255
    ether de:c1:d9:c2:e9:09  txqueuelen 0  (Ethernet)
      RX packets 709  bytes 77276 (75.4 KiB)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 659  bytes 76910 (75.1 KiB)
      TX errors 0  dropped 0  overruns 0  carrier 0
      collisions 0

[bitternot@honey)-[/var/log]
$
```

(bitternot@honey)-[~]

```
[not@honey ~] via v23.11.1 via v3.13.3
[not@honey ~] * ssh bitternot@172.17.0.2
bitternot@172.17.0.2's password:
Linux honey 6.14.9-zen1-1-zen #1 ZEN SMP PREEMPT_DYNAMIC Thu, 29 May 2025 21:42:00 +0000 x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in
the individual files in /usr/share/doc/*copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Last login: Tue Jul 22 17:16:42 2025 from 172.17.0.1
(Message from Kali developers)

This is a minimal installation of Kali Linux, you likely
want to install supplementary tools. Learn how:
→ https://www.kali.org/docs/troubleshooting/common-minimum-s
etup/

(Run: "touch ~/.hushlogin" to hide this message)
```

Figure 48: SSH Connection Successful

HTTP connection fails and logged because the iptable drop inbound HTTP traffic.

The screenshot shows a terminal window with two panes. The left pane displays log entries from the file /var/log/syslog, specifically from rsyslogd. One entry shows a connection attempt to https://www.rsyslog.com. The right pane shows a command-line interface where curl is used to attempt a connection to http://172.17.0.2/. A red box highlights the curl command, and a red arrow points from it to the text "Trying to connect to http server but failed to connect" located below the command.

```
(bitternot@honey)-:/var/log]
$ sudo tail -n 2 -f syslog
2025-07-22T17:15:30.631044+00:00 honey rsyslogd: [origin software="rsyslogd" swVersion="8.2504.0" x-pid="167" x-info="https://www.rsyslog.com"] start
Jul 22 23:55:30 bitternot kernel: [123456.789012] HTTP BLOCKED: IN=eth0 OUT= MAC=8e:93:cc:0a:b8:7a:de:c1
:d9:c2:e9:09 SRC=172.17.0.2 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=54321 DF PROTO=TCP SPT=56
789 DPT=80 WINDOW=92200 RES=0x00 SYN URGP=0
]

not@honey:~ via 192.168.1.1 via v3.13.3 took 0s
-X curl http://172.17.0.2/
Trying to connect to http server but failed to connect
```

Figure 49: Failed HTTP Connection

4.2. VPN Components and Security

VPN stands for "Virtual Private Network" and describes the opportunity to establish a protected network connection when using public networks. VPNs encrypt your internet traffic and disguise your online identity. This makes it more difficult for third parties to track your activities online and steal data. The encryption takes place in real time. ([Kaspersky, 2023](#))

Working Mechanism of VPN

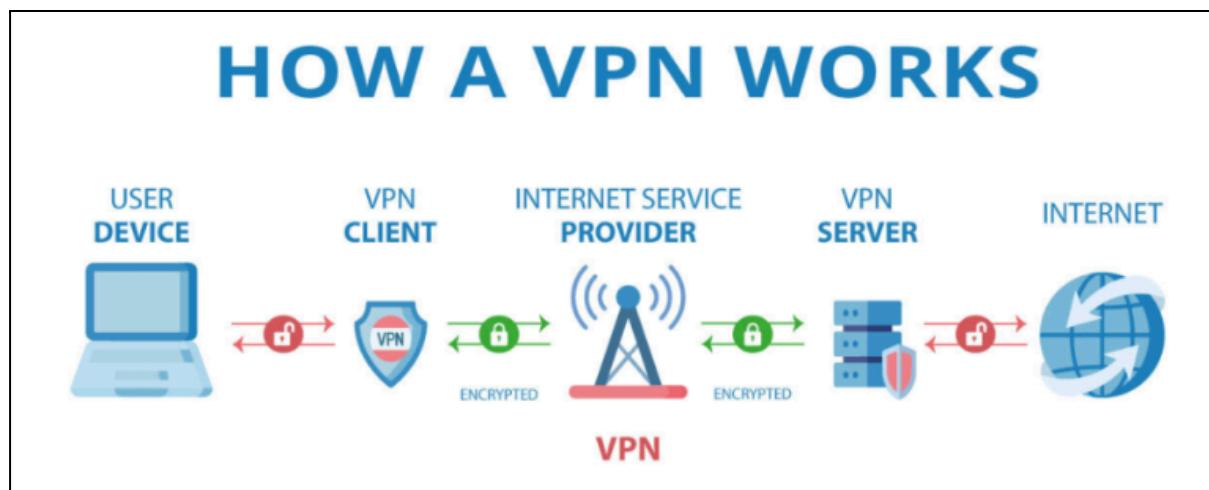


Figure 50: How VPN Works

Between a device and a remote server, a virtual private network (VPN) builds an encrypted tunnel. This tunnel protects data from interception, masks the originating IP address, and secures internet contact. After going over the VPN server, encrypted data is decrypted and sent on to its final position.

Key Components and Technologies in VPN

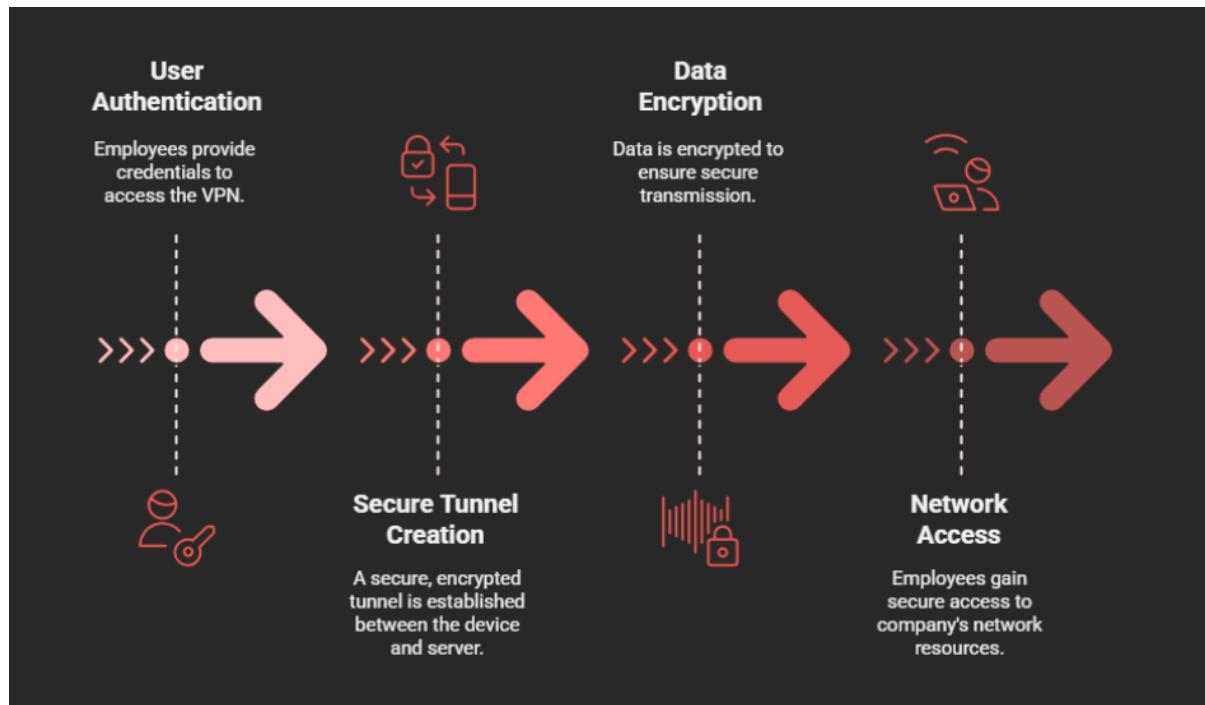


Figure 51: VPN Components and Technologies

A virtual private network, or VPN, creates a protected, encrypted connection over a public network between a device and a remote server. By masking data from hackers and illegal access, it facilitates safe communication. VPNs transfer data securely using **tunneling** protocols like OpenVPN, L2TP/IPsec, and PPTP. External parties cannot read the data thanks to **encryption** methods like AES and 3DES. User identification is proven using **authentication** measures such as digital certificates or passwords. These technologies work together to ensure safe and confidential internet data delivery. ([Netalit, 2023](#))

4.3. Understanding Network Operating Systems

Overview

The network operating system (NOS) in a network regulates communication and resource sharing among linked devices. It manages file or printer sharing, user access, and security. By leveraging common protocols to link hardware and software, NOS aids organizations in efficiently administering and preserving their networks. ([Yasar & Lewis, 2023](#))

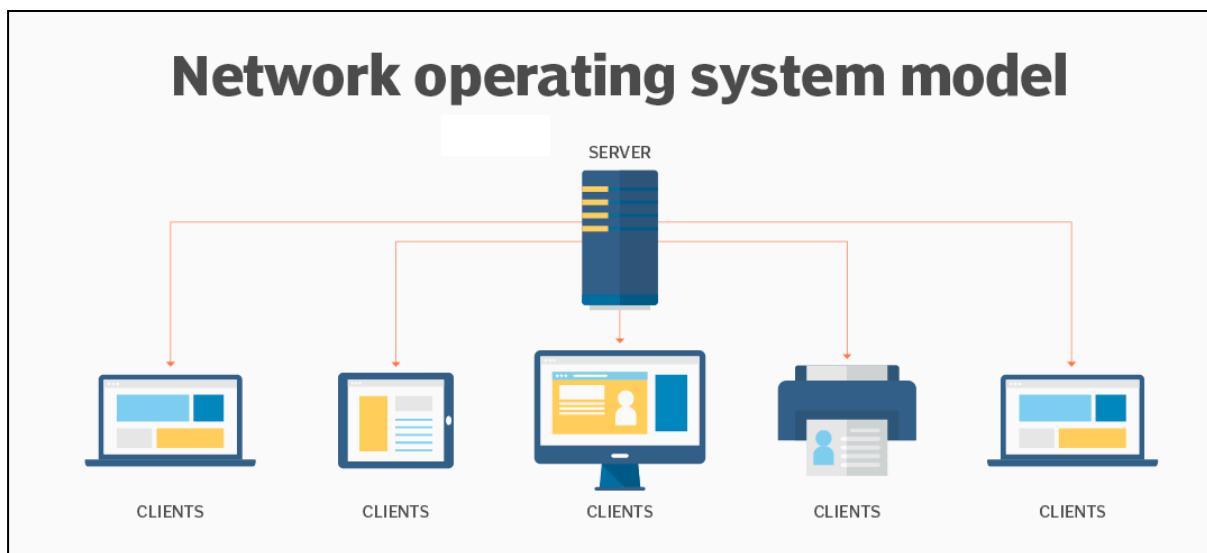


Figure 52: NOS Model

Popular NOS

Microsoft Windows Server is a popular network operating system that offers centralized management, file sharing, and remote access. Linux-based systems like Red Hat and Ubuntu are open-source, scalable, and highly configurable for enterprise use.

Key Feature

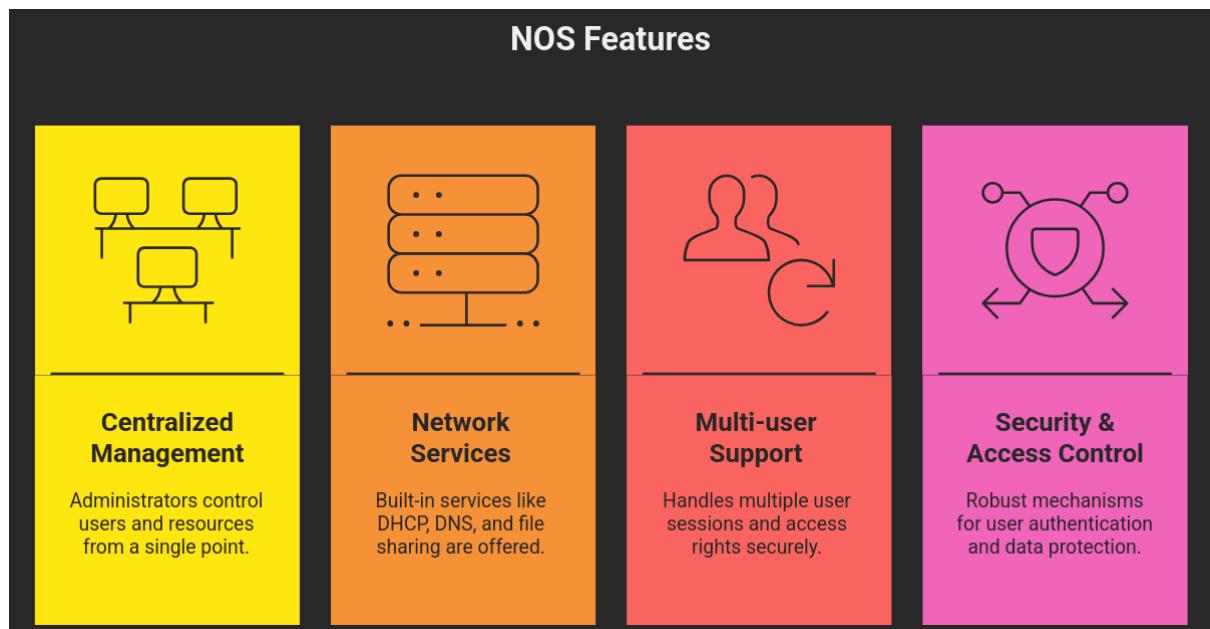


Figure 53: Key Feature Distinguishing NOS from Traditional OS

Common Protocol

The core set of protocols that facilitates communication between devices on a network is dubbed TCP/IP (Transmission Control Protocol/Internet Protocol). File and printer sharing is the major role of SMB (Server Message Block) in Windows environments, whereas NFS (Network File System) fulfills the same duty on Unix/Linux systems. User credentials and other distributed directory information are retrieved and processed by LDAP (Lightweight Directory Access Protocol). Furthermore, to assist flawless network operations, the DNS and DHCP protocols govern hostname resolution and dynamic IP address assignment, respectively. ([Yasar, 2025](#))

Conclusion

In summary, the use of effective design concepts, robust security mechanisms, and appropriate communication protocols are crucial to establishing a safe and trustworthy network configuration. One can preserve both performance and security by learning the distinctions between protocols, addressing weaknesses, and establishing efficient configurations. VLANs, firewalls, encryption, tunneling, authentication, and redundancy are some of the technologies that provide safe and uninterrupted communication. As a result, a well-designed network preserves essential information from new hazards while concurrently boosting operating efficiency.

References

BasuMallick, C. (2022, April 18). *Differences between TCP and UDP*. Spiceworks.

<https://www.spiceworks.com/tech/networking/articles/tcp-vs-udp/>

Danielsekot. (2019, February 11). *Spanning Tree Protocol attacks: protective measures*.

ProSec GmbH.

<https://www.prosec-networks.com/en/blog/spanning-tree-protokoll-angriffe-3-attacks-und-schutzmassnahmen/>

GeeksforGeeks. (2023, March 23). *Basic network attacks in computer network*.

GeeksforGeeks.

<https://www.geeksforgeeks.org/computer-networks/basic-network-attacks-in-computer-network/>

Imperva. (2019, December 29). *What is MITM (Man in the middle) attack* | Imperva.

Learning Center.

<https://www.imperva.com/learn/application-security/man-in-the-middle-attack-mitm/>

Kaspersky. (2023, July 31). *What is VPN? How it works, types of VPN*. www.kaspersky.com.

<https://www.kaspersky.com/resource-center/definitions/what-is-a-vpn>

Masas, R. (2023, December 21). *What is MITM (Man in the Middle) Attack* | Imperva.

Learning Center.

<https://www.imperva.com/learn/application-security/man-in-the-middle-attack-mitm/>

Netalit. (2023, July 13). *How does a VPN work? Check Point Software*.

<https://www.checkpoint.com/cyber-hub/network-security/what-is-vpn/how-does-a-vpn-work/>

TCP session hijacking. (2009, August 1).

<https://www.usna.edu/Users/cs/choi/it432/lec/l04/lec.html>

What is a UDP flood DDoS attack? (2022, May 1). Akamai.

<https://www.akamai.com/glossary/what-is-udp-flood-ddos-attack>

What is DHCP Spoofing? How It Works & Examples | Twingate. (2020, July 27).

<https://www.twingate.com/blog/glossary/dhcp%20spoofing>

What is network security? Definition and types | Fortinet. (2022, July 2). Fortinet.

<https://www.fortinet.com/resources/cyberglossary/what-is-network-security>

Yasar, K. (2025, February 27). *15 common network protocols and their functions explained*.

Search Networking.

<https://www.techtarget.com/searchnetworking/feature/12-common-network-protocols-and-their-functions-explained>

Yasar, K., & Lewis, S. (2023, March 8). *network operating system (NOS)*. Search

Networking.

<https://www.techtarget.com/searchnetworking/definition/network-operating-system>

Appendix

Task-2

Access-2_bitter!honey

```
Switch(config)#hostname ACCESS-2_bitter!honey
ACCESS-2_bitter!honey(config)#vlan 20
ACCESS-2_bitter!honey(config-vlan)#name IT
ACCESS-2_bitter!honey(config-vlan)#int range f0/1-2
ACCESS-2_bitter!honey(config-if-range)#switchport mode access
ACCESS-2_bitter!honey(config-if-range)#switchport access vlan 20
ACCESS-2_bitter!honey(config-if-range)#do sh vlan bri
```

VLAN Name Status Ports

VLAN Name	Status	Ports
1 default	active	Fa0/3, Fa0/4, Fa0/5, Fa0/6 Fa0/7, Fa0/8, Fa0/9, Fa0/10 Fa0/11, Fa0/12, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24, Gig0/1, Gig0/2
20 IT	active	Fa0/1, Fa0/2
1002 fddi-default	active	
1003 token-ring-default	active	
1004 fddinet-default	active	
1005 trnet-default	active	

vlan brief

```
ACCESS-2_bitter!honey(config-if-range)#
ACCESS-2_bitter!honey(config)#int range f0/3-4
ACCESS-2_bitter!honey(config-if-range)#switchport mode trunk
```

```
ACCESS-2_bitter!honey(config)#
ACCESS-2_bitter!honey(config)#int range f0/3-4
ACCESS-2_bitter!honey(config-if-range)#switchport mode trunk
```

```
ACCESS-2_bitter!honey(config-if-range)#do sh int trunk
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up
```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/3	on	802.1q	trunking	1
Fa0/4	on	802.1q	trunking	1

```
Port      Vlans allowed on trunk
Fa0/3    1-1005
Fa0/4    1-1005
```

```
Port      Vlans allowed and active in management domain
Fa0/3    1,20
Fa0/4    1,20
```

```
Port      Vlans in spanning tree forwarding state and not pruned
Fa0/3    none
Fa0/4    none
```

Access-3_bitter!honey

```

Enter configuration commands, one per line. End with Ctrl/Z.
Switch(config)#hostname ACCESS-3_bitter!honey
ACCESS-3_bitter!honey(config)#vlan 30
ACCESS-3_bitter!honey(config-vlan)#name HR
ACCESS-3_bitter!honey(config-vlan)#int range f0/1-2
ACCESS-3_bitter!honey(config-if-range)#switchport mode access
ACCESS-3_bitter!honey(config-if-range)#switchport access vlan 30
ACCESS-3_bitter!honey(config-if-range)#do sh vlan bri

VLAN Name          Status    Ports
----- -----
1     default       active    Fa0/3, Fa0/4, Fa0/5, Fa0/6
                      Fa0/7, Fa0/8, Fa0/9, Fa0/10
                      Fa0/11, Fa0/12, Fa0/13, Fa0/14
                      Fa0/15, Fa0/16, Fa0/17, Fa0/18
                      Fa0/19, Fa0/20, Fa0/21, Fa0/22
                      Fa0/23, Fa0/24, Gig0/1, Gig0/2
30    HR            active    Fa0/1, Fa0/2
1002  radd1-default active
1003  token-ring-default active
1004  fddinet-default   active
1005  trnet-default    active
ACCESS-3_bitter!honey(config-if-range)#

```

```

ACCESS-3_bitter!honey(config-if-range)#
ACCESS-3_bitter!honey(config-if-range)#int range f0/3-4
ACCESS-3_bitter!honey(config-if-range)#switchport mode trunk
1      Enable Trunking

ACCESS-3_bitter!honey(config-if-range)#do sh int trunk
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up

Port      Mode       Encapsulation  Status      Native vlan
Fa0/3    on         802.1q        trunking    1
Fa0/4    on         802.1q        trunking    1
2      Status

Port      Vlans allowed on trunk
Fa0/3    1-1005
Fa0/4    1-1005

Port      Vlans allowed and active in management domain
Fa0/3    1,30
Fa0/4    1,30
3      Vlan allowed

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/3    none
Fa0/4    none

```

Access-4_bitter!honey

```

SWITCH(config)#hostname ACCESS-4_bitter!honey
ACCESS-4_bitter!honey(config)#vlan 40
ACCESS-4_bitter!honey(config-vlan)#name ACCOUNT
ACCESS-4_bitter!honey(config-vlan)#int range f0/1-2
ACCESS-4_bitter!honey(config-if-range)#switchport mode access
ACCESS-4_bitter!honey(config-if-range)#switchport access vlan 40
ACCESS-4_bitter!honey(config-if-range)#do sh vlan bri

```

VLAN Name	Status	Ports
1 default	active	Fa0/3, Fa0/4, Fa0/5, Fa0/6 Fa0/7, Fa0/8, Fa0/9, Fa0/10 Fa0/11, Fa0/12, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24, Gig0/1, Gig0/2
40 ACCOUNT	active	Fa0/1, Fa0/2
1002 fddi-default	active	
1003 token-ring-default	active	
1004 fddinet-default	active	
1005 trnet-default	active	

```

ACCESS-4_bitter!honey(config)#
ACCESS-4_bitter!honey(config)#int range f0/3-4
ACCESS-4_bitter!honey(config-if-range)#switchport mode trunk
ACCESS-4_bitter!honey(config-if-range)#do sh int trunk

```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/3	on	802.1q	trunking	1
Fa0/4	on	802.1q	trunking	1

Port Vlans allowed on trunk
Fa0/3 1-1005
Fa0/4 1-1005

Port Vlans allowed and active in management domain
Fa0/3 1,40
Fa0/4 1,40

Port Vlans in spanning tree forwarding state and not pruned
Fa0/3 1,40
Fa0/4 1,40

Access-5_bitter!honey

```
Switch(config)#hostname ACCESS-5_bitter!honey
ACCESS-5_bitter!honey(config)#vlan 50
ACCESS-5_bitter!honey(config-vlan)#name ADMINISTRATION
ACCESS-5_bitter!honey(config-vlan)#int range f0/1-2
ACCESS-5_bitter!honey(config-if-range)#switchport mode access
ACCESS-5_bitter!honey(config-if-range)#switchport access vlan 50
ACCESS-5_bitter!honey(config-if-range)#do sh vlan bri
```

VLAN Name Status Ports

1 default	active	Fa0/3, Fa0/4, Fa0/5, Fa0/6 Fa0/7, Fa0/8, Fa0/9, Fa0/10 Fa0/11, Fa0/12, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24, Gig0/1, Gig0/2
50 ADMINISTRATION	active	Fa0/1, Fa0/2
1002 fddi-default	active	
1003 token-ring-default	active	
1004 fddinet-default	active	
1005 trnet-default	active	

1 Creating Vlan

2 Enable access mode

3 Vlan brief

```
ACCESS-5_bitter!honey(config)#int range f0/3-4
ACCESS-5_bitter!honey(config)#switchport mode trunk
ACCESS-5_bitter!honey(config-if-range)#do sh int trunk
Port      Mode       Encapsulation  Status      Native vlan
Fa0/3    on        802.1q         trunking    1
Fa0/4    on        802.1q         trunking    1

Port      Vlans allowed on trunk
Fa0/3    1-1005
Fa0/4    1-1005

Port      Vlans allowed and active in management domain
Fa0/3    1,50
Fa0/4    1,50

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/3    1,50
Fa0/4    1,50
```

Dist-2_bitter!honey

```
DIST-2_bitter!honey(config)#int range f0/1-5
DIST-2_bitter!honey(config-if-range)#switchport trunk encapsulation dot1q
DIST-2_bitter!honey(config-if-range)#switchport mode trunk
DIST-2_bitter!honey(config-if-range)#do sh int trunk
Port      Mode      Encapsulation  Status      Native vlan
Fa0/1    on       802.1q        trunking    1
Fa0/2    on       802.1q        trunking    1
Fa0/3    on       802.1q        trunking    1
Fa0/4    on       802.1q        trunking    1
Fa0/5    on       802.1q        trunking    1      Enable Trunking

Port      Vlans allowed on trunk
Fa0/1    1-1005
Fa0/2    1-1005
Fa0/3    1-1005
Fa0/4    1-1005
Fa0/5    1-1005

Port      Vlans allowed and active in management domain
Fa0/1    1,10,20,30,40,50
Fa0/2    1,10,20,30,40,50
Fa0/3    1,10,20,30,40,50
Fa0/4    1,10,20,30,40,50
Fa0/5    1,10,20,30,40,50

Port      Vlans in spanning tree forwarding state and not pruned
```

1 Enable Trunking

2 Allowed Vlans

```

interface Vlan10
mac-address 0001.638e.9a01
ip address 172.16.99.126 255.255.255.128
ip helper-address 11.0.0.2
ip helper-address 10.0.0.2
!
interface Vlan20
mac-address 0001.638e.9a02
ip address 172.16.99.254 255.255.255.128
ip helper-address 11.0.0.2
ip helper-address 10.0.0.2
!
interface Vlan30
mac-address 0001.638e.9a03
ip address 172.16.100.30 255.255.255.224
ip helper-address 11.0.0.2
ip helper-address 10.0.0.2
!
interface Vlan40
mac-address 0001.638e.9a04
ip address 172.16.100.46 255.255.255.240
ip helper-address 11.0.0.2
ip helper-address 10.0.0.2
!
interface Vlan50
mac-address 0001.638e.9a05
ip address 172.16.100.110 255.255.255.192
ip helper-address 11.0.0.2
ip helper-address 10.0.0.2
!
router eigrp 30
network 172.16.0.0
network 11.0.0.0
network 14.0.0.0
no auto-summary

```

Core-2_bitter!honey

```

CORE-2_bitter!honey#sh etherchannel su
Flags: D - down      P - 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
      u - unsuitable for bundling
      w - waiting to be aggregated
      d - default port

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

Group  Port-channel  Protocol     Ports
-----+-----+-----+
2      Po2(RU)       LACP        Fa0/1(P)  Fa0/2(P)

```

```
CORE-2 bitter!honey#sh ip int br | ex un
Interface          IP-Address      OK? Method Status          Protocol
Port-channel2      18.0.0.2        YES manual up           up
FastEthernet0/3    14.0.0.3        YES manual up           up
FastEthernet0/4    140.0.0.1       YES manual up          up
CORE-2 bitter!honey#
```

```
router eigrp 30
  network 14.0.0.0
  network 18.0.0.0
  network 140.0.0.0
  auto-summary
!
```

DHCP-1_SERVER



DHCP-2_SERVER



Task-3

Access-SW-2_bitter!honey

```
ACCESS-SW-2_bitter!honey#sh vlan br
VLAN Name Status Ports
---- -- -- -----
1 default active Fa0/5, Fa0/6, Fa0/7, Fa0/8
                  Fa0/9, Fa0/10, Fa0/11, Fa0/12
                  Fa0/13, Fa0/14, Fa0/15, Fa0/16
                  Fa0/17, Fa0/18, Fa0/19, Fa0/20
                  Fa0/21, Fa0/22, Fa0/23, Fa0/24
                  Gig0/1, Gig0/2
10 MARKETING active
20 EXAM active Fa0/1, Fa0/2
30 FACULTY active
1002 fddi-default active
1003 token-ring-default active
1004 fddinet-default active
1005 trnet-default active
ACCESS-SW-2_bitter!honey#
```

```
ACCESS-SW-2_bitter!honey#sh int trunk
Port      Mode      Encapsulation  Status      Native vlan
Po2       on        802.1q         trunking    1

Port      Vlans allowed on trunk
Po2       1-1005

Port      Vlans allowed and active in management domain
Po2       1,10,20,30

Port      Vlans in spanning tree forwarding state and not pruned
Po2       1,10,20,30
```

```
ACCESS-SW-2_bitter!honey#sh etherchannel su
Flags: D - down      P - 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
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port

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

Group Port-channel Protocol Ports
-----+-----+-----+
2      Po2(SU)      PAgP     Fa0/3(P) Fa0/4(P)
ACCESS-SW-2_bitter!honey#
```

```
spanning-tree mode rapid-pvst
spanning-tree extend system-id
!
interface Port-channel2
    switchport mode trunk
    switchport nonegotiate
!
interface FastEthernet0/1
    description "CONNECTED TO END DEVICES"
    switchport access vlan 20
    switchport mode access
!
interface FastEthernet0/2
    description "CONNECTED TO END DEVICES"
    switchport access vlan 20
    switchport mode access
!
interface FastEthernet0/3
    description "CONNECTED TO DIST-SW"
    switchport mode trunk
    switchport nonegotiate
    channel-group 2 mode auto
!
interface FastEthernet0/4
    description "CONNECTED TO DIST-SW"
    switchport mode trunk
    switchport nonegotiate
    channel-group 2 mode auto
!
interface FastEthernet0/5
    switchport mode trunk
    switchport nonegotiate
!
interface FastEthernet0/6
    switchport mode trunk
    switchport nonegotiate
```

Access-SW-3_bitter!honey

VLAN	Name	Status	Ports
1	default	active	Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gig0/1, Gig0/2
10	MARKETING	active	
20	EXAM	active	
30	FACULTY	active	Fa0/1, Fa0/2
1002	fdci default	active	
1003	token-ring-default	active	

ACCESS-SW-3_bitter!honey# sh int trunk				
Port	Mode	Encapsulation	Status	Native vlan
Po3	on	802.1q	trunking	1
Port Vlans allowed on trunk				
Po3	1-1005			
Port Vlans allowed and active in management domain				
Po3	1,10,20,30			
Port Vlans in spanning tree forwarding state and not pruned				
Po3	1,10,20,30			
ACCESS-SW-3_bitter!honey#				
ACCESS-SW-3_bitter!honey#				
ACCESS-SW-3_bitter!honey#				
ACCESS-SW-3_bitter!honey# sh etherchannel su				
Flags: D - down P - 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				
u - unsuitable for bundling				
w - waiting to be aggregated				
d - default port				
Number of channel-groups in use: 1				
Number of aggregators: 1				
Group	Port-channel	Protocol	Ports	
3	Po3(SU)	LACP	Fa0/3(P) Fa0/4(P)	

```
spanning-tree mode rapid-pvst
spanning-tree extend system-id
!
interface Port-channel3
    switchport mode trunk
    switchport nonegotiate
!
interface FastEthernet0/1
    description "CONNECTED TO END DEVICES"
    switchport access vlan 30
    switchport mode access
!
interface FastEthernet0/2
    description "CONNECTED TO END DEVICES"
    switchport access vlan 30
    switchport mode access
!
interface FastEthernet0/3
    description "CONNECTED TO DIST SW"
    switchport mode trunk
    switchport nonegotiate
    channel-group 3 mode passive
!
interface FastEthernet0/4
    description "CONNECTED TO DIST SW"
    switchport mode trunk
    switchport nonegotiate
    channel-group 3 mode passive
!
interface FastEthernet0/5
    switchport mode trunk
    switchport nonegotiate
!
interface FastEthernet0/6
    switchport mode trunk
    switchport nonegotiate
!
```

DIST-SW_bitter!honey

VLAN	Name	Status	Ports
1	default	active	Fa0/8, Fa0/9, Fa0/10, Fa0/11 Fa0/12, Fa0/13, Fa0/14, Fa0/15 Fa0/16, Fa0/17, Fa0/18, Fa0/19 Fa0/20, Fa0/21, Fa0/22, Fa0/23 Fa0/24, Gig0/1, Gig0/2
10	MARKETING	active	
20	EXAM	active	
30	FACULTY	active	
1002	fddi-default	active	
1003	token-ring-default	active	
1004	fdtnet-default	active	
1005	trnet-default	active	

DIST-SW_bitter!honey(config)# do sh int trunk				
Port	Mode	Encapsulation	Status	Native vlan
Po1	on	802.1q	trunking	1
Po2	on	802.1q	trunking	1
Po3	on	802.1q	trunking	1
Fa0/7	on	802.1q	trunking	1

Port	Vlans allowed on trunk
Po1	1-1005
Po2	1-1005
Po3	1-1005
Fa0/7	1-1005

Port	Vlans allowed and active in management domain
Po1	1,10,20,30
Po2	1,10,20,30
Po3	1,10,20,30
Fa0/7	1,10,20,30

Port	Vlans in spanning tree forwarding state and not pruned
Po1	10,20,30
Po2	1,10,20,30
Po3	1,10,20,30
Fa0/7	1,10,20,30

```

DIST-SW_bitter!honey(config)#do sh spanning-tree su
Switch is in rapid-pvst mode
Root bridge for: default MARKETING EXAM FACULTY
Extended system ID      is enabled
Portfast Default        is disabled
PortFast BPDU Guard Default  is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default       is disabled
EtherChannel misconfig guard is disabled
UplinkFast              is disabled
BackboneFast             is disabled
Configured Pathcost method used is short

Name          Blocking Listening Learning Forwarding STP Active
-----
VLAN0001      7          0          0          3          10
VLAN0010      6          0          0          4          10
VLAN0020      6          0          0          4          10
VLAN0030      6          0          0          4          10

4 vlans      25         0          0          15         40

```

```

spanning-tree mode rapid-pvst
spanning-tree extend system-id

!
interface Port-channel1
  switchport mode trunk
  switchport nonegotiate
!
interface Port-channel2
  switchport mode trunk
  switchport nonegotiate
!
interface Port-channel3
  switchport mode trunk
  switchport nonegotiate
!
interface FastEthernet0/1
  description "CONNECTED TO ACCESS LAYER"
  switchport mode trunk
  switchport nonegotiate
  channel-group 1 mode passive
!
interface FastEthernet0/2
  description "CONNECTED TO ACCESS LAYER"
  switchport mode trunk
  switchport nonegotiate
  channel-group 1 mode passive
!
interface FastEthernet0/3
  description "CONNECTED TO ACCESS LAYER"
  switchport mode trunk
  switchport nonegotiate
  channel-group 2 mode desirable
!
interface FastEthernet0/4
  description "CONNECTED TO ACCESS LAYER"
  switchport mode trunk
  switchport nonegotiate
  channel-group 2 mode desirable

```

```

banner motd ^C
+-----+
|   WARNING! Unauthorized access is strictly prohibited.      |
|   All activities are logged in the system.                  |
+-----+
^C

```

CORE_bitter!honey

```
hostname CORE_bitter!honey
!
!
!
enable secret 5 $1$mERr$MSCsGn1R8P02jnd.nLgVl/
!
!
!
!
ip cef
no ipv6 cef
!
!
!
username ADMIN privilege 15 secret 5 $1$mERr$tKAvQPi4nwql8vUTe8ogt/
username operator privilege 7 secret 5 $1$mERr$tKAvQPi4nwql8vUTe8ogt/
username user privilege 2 secret 5 $1$mERr$tKAvQPi4nwql8vUTe8ogt/
!
!
!
!
ip ssh version 2
ip ssh authentication-retries 2
ip ssh time-out 60
ip domain-name dinesh.com
!
!
spanning-tree mode rapid-pvst
!
```

```
!
interface GigabitEthernet0/0/0
description "CONNECTED TO DIST-SW-bitter!honey"
no ip address
duplex auto
speed auto
!
interface GigabitEthernet0/0/0.10
description "DEFAULT GATEWAY FOR VLAN10"
encapsulation dot1Q 10
ip address 192.168.99.62 255.255.255.192
!
interface GigabitEthernet0/0/0.20
description "DEFAULT GATEWAY FOR VLAN20"
encapsulation dot1Q 20
ip address 192.168.99.126 255.255.255.192
!
interface GigabitEthernet0/0/0.30
description "DEFAULT GATEWAY FOR VLAN30"
encapsulation dot1Q 30
ip address 192.168.99.190 255.255.255.192
!
interface GigabitEthernet0/0/1
no ip address
duplex auto
speed auto
shutdown
```

```
!
banner motd ^C
+-----+
|   WARNING! Unauthorized access is strictly prohibited.      |
|   All activities are logged in the system.                  |
+-----+
^C
```

```
privilege exec level 2 ping
privilege exec level 7 show
privilege exec level 2 show cdp
privilege exec level 2 show cdp neighbors
privilege exec level 7 show interfaces
privilege exec level 7 show interfaces trunk
privilege exec level 2 show ip
privilege exec level 2 show ip interface
privilege exec level 2 show ip interface brief
privilege exec level 7 show running-config
privilege exec level 7 show startup-config
privilege exec level 2 show version
privilege exec level 7 show vlan-switch
privilege exec level 7 show vlan-switch brief
privilege exec level 7 show vtp
privilege exec level 7 show vtp status
privilege exec level 7 traceroute
```

```
!
```

```
!
```

```
!
```

```
line con 0
logging synchronous
login local
```

```
!
```

```
line aux 0
```

```
!
```

```
line vty 0 4
logging synchronous
login local
transport input telnet
line vty 5 15
logging synchronous
login local
transport input ssh
```

```
!
```

```
!
```

```
CORE_bitter!honey(config)#do sh ip int br | ex un
Interface          IP-Address      OK? Method Status      Protocol
GigabitEthernet0/0/0.10192.168.99.62  YES manual up        up
GigabitEthernet0/0/0.20192.168.99.126  YES manual up        up
GigabitEthernet0/0/0.30192.168.99.190  YES manual up        up
```

C:\>ssh -l admin 192.168.99.190

1 login as admin

Password:

% Login invalid

2 MOTD

Password:

```
+-----+  
| WARNING! Unauthorized access is strictly prohibited. |  
| All activities are logged in the system. |  
+-----+
```

3 Command

```
CORE_bitter!honey#  
CORE_bitter!honey#  
CORE_bitter!honey#en  
CORE_bitter!honey#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
CORE_bitter!honey(config)#do sh ip int bri | ex un  
Interface IP-Address OK? Method Status Protocol  
GigabitEthernet0/0/0.10192.168.99.62 YES NVRAM up  
GigabitEthernet0/0/0.20192.168.99.126 YES NVRAM up  
GigabitEthernet0/0/0.30192.168.99.190 YES NVRAM up  
CORE_bitter!honey(config)#
```

C:\>

C:\>

C:\>telnet 192.168.99.126

Trying 192.168.99.126 ...open

```
+-----+  
| WARNING! Unauthorized access is strictly prohibited. |  
| All activities are logged in the system. |  
+-----+
```

MOTD banner

User Access Verification

Username: admin

Password:

```
CORE_bitter!honey#  
CORE_bitter!honey#en  
CORE_bitter!honey#configt  
Translating "configt"...domain server (255.255.255.255) % Name lookup aborted  
CORE_bitter!honey#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
CORE_bitter!honey(config)#do sh ip int bri | ex un  
Interface IP-Address OK? Method Status Protocol  
GigabitEthernet0/0/0.10192.168.99.62 YES NVRAM up  
GigabitEthernet0/0/0.20192.168.99.126 YES NVRAM up  
GigabitEthernet0/0/0.30192.168.99.190 YES NVRAM up  
CORE_bitter!honey(config)#
```