

PRIME COLLEGE

Khusibun, Kathmandu



LAB REPORT OF NETWORK AND SYSTEM ADMINISTRATION

Submitted by:

Name: Suyash Shrestha

Faculty: BSc.CSIT

Semester: 8th

Submitted to:

Hiranya Bastakoti

CONTENTS

Lab No	Title/Question	Submission Date	Signature
1	Install and configure a server-client setup using VMware, ensuring proper network connectivity between virtual machines		
2	Install and configure CentOS in VMware		
3	Capture, analyze and filter network traffic using Wire shark.		
4	Design and implement VLANs using Cisco Packet Tracer and troubleshooting.		
5	Create and manage user accounts, groups and file permissions in Linux.		
6	Configure network interfaces, assign static and dynamic IPs, enable/disable network services and troubleshoot connectivity issues		
7	Implement firewall rules to control inbound and outbound traffic and test policies.		
8	Setup DNS servers, configure domain name resolution and troubleshoot common issues		
9	Setup DHCP servers and troubleshoot common networking issues.		
10	Install and configure Apache/Nginx web servers.		
11	Install and configure Squid proxy servers		
12	Install, Configure and Test MySQL Server.		
13	Install and Configure mail servers (Postfix/Sendmail), enable SMTP, IMAP/POP3 services		
14	Configure file-sharing protocols (SAMBA for Windows-Linux, NFS for Linux-Linux sharing)		
15	Configure and Setup CUPS for network printing		
16	Install and Configure FTP server for file transfers		

Lab 1

Title: Install and configure a server-client setup using VMware, ensuring proper network connectivity between virtual machines

1. Objectives:

- To understand the concept of virtualization and VMware.
- To understand the importance of virtualization in server-client architecture.
- To understand networking in virtual machines.

2. Theory:

VMware:

VMware is a leading software provider for virtualization technology, enabling multiple operating systems to run on a single physical machine. It provides hypervisors such as VMware Workstation, which create and manage virtual machines.

Virtualization:

Virtualization is the process of creating a virtual instance of a computer system, including hardware, storage, and networking, using a hypervisor. This allows multiple virtual machines to operate independently on the same physical server, improving resource utilization, flexibility and scalability.

Server-Client Architecture:

A server-client model is a computing structure where a central server provides resources and services to multiple clients. In a virtualized environment, both the server and client machines are hosted as virtual machines within VMware.

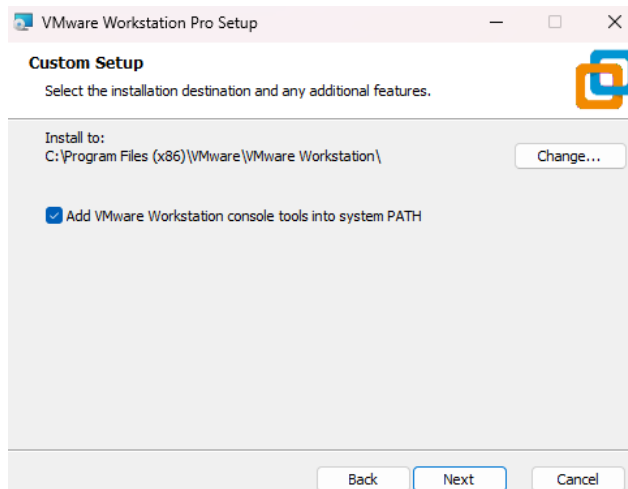
Networking in Virtual Machines:

VMware provides multiple networking options:

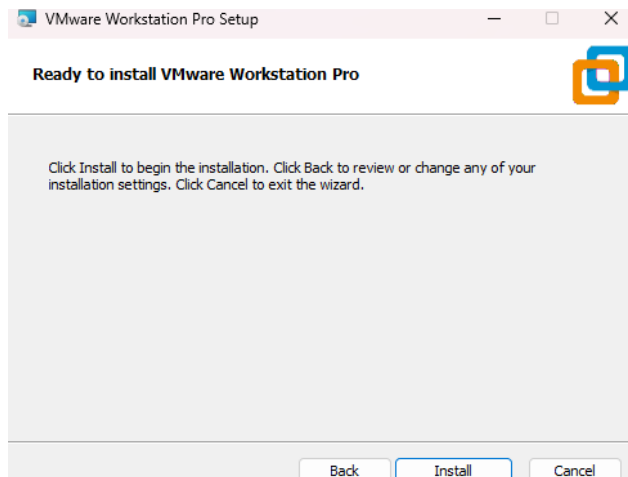
1. Bridged Network: The VM acts like a physical device on the local network.
2. NAT (Network Address Translation): The VM shares the host's IP address to access external networks.
3. Host-Only Network: The VM can communicate only with other VMs and the host machine.

3. Installation Steps:

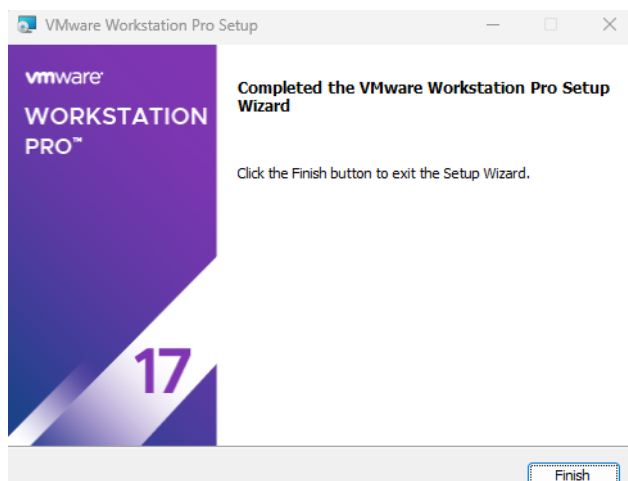
1. Download VMware Workstation Pro executable file.
2. Run the executable setup file and click Next.
3. Go on clicking the default options and add it to path.



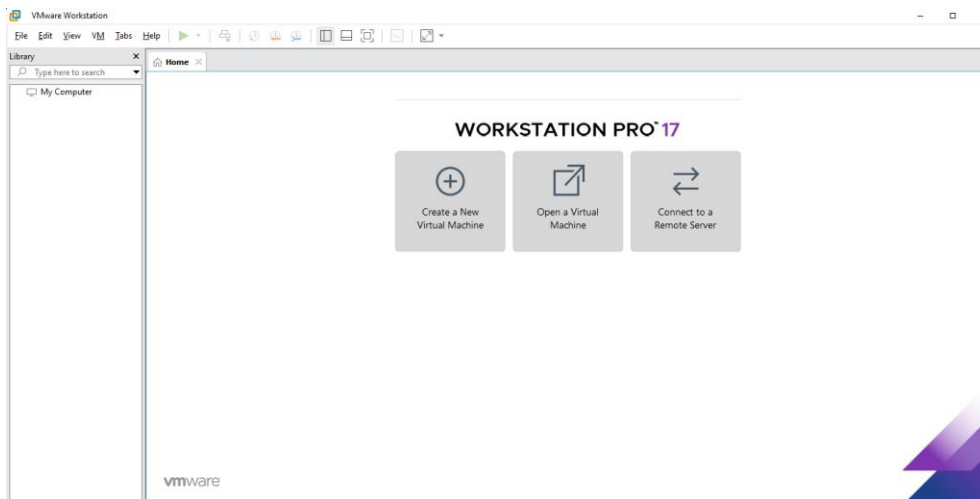
4. Finally, after all the setups, click on install.



5. Finally, click on finish after all the setup has been completed.



6. The following UI is seen after the VMware has been successfully installed.



7. Now, create two virtual machines with the required ISO files.
8. Configure the networking (leave default as NAT).
9. Then, setup the server with necessary services such as DNS, Web servers, etc.
10. Setup the client VM by ensuring it is on the same network as server.
11. Finally, test the network connectivity by using the ping command.

4. Outcome:

To gain hands-on practical knowledge in installing and configuring virtualized server client environment configure networking in VMware and establishing communication between VMs.

5. Conclusion:

By following these steps, we will be able to install and configure server-client using VMware. We successfully established network connectivity between virtual machines, enabling them to communicate efficiently, which can be an important asset for virtualized IT infrastructures.

Lab 2

Title: Install and configure CentOS in VMware.

1. Objectives:

- To learn how to install CentOS in a virtualized environment using VMware.
- To familiarize with the basic configuration of a CentOS virtual machine.

2. Theory:

VMware and Virtualization:

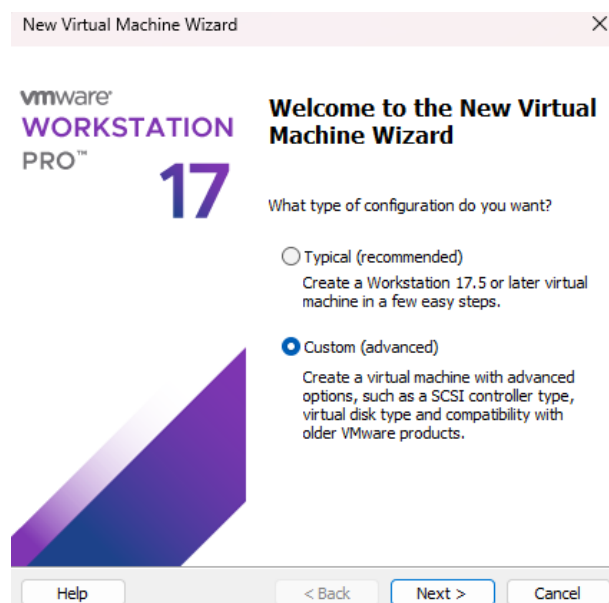
VMware is a leading provider of virtualization technology, which allows multiple OS to run on a single physical machine enabling efficient resource utilization, isolation and simplified system management. Virtualization refers to the creation of a virtual version of computing resources, such as OS, server or storage. Using VMware, users can create VMs that function as independent computers, each with its own OS and applications.

CentOS:

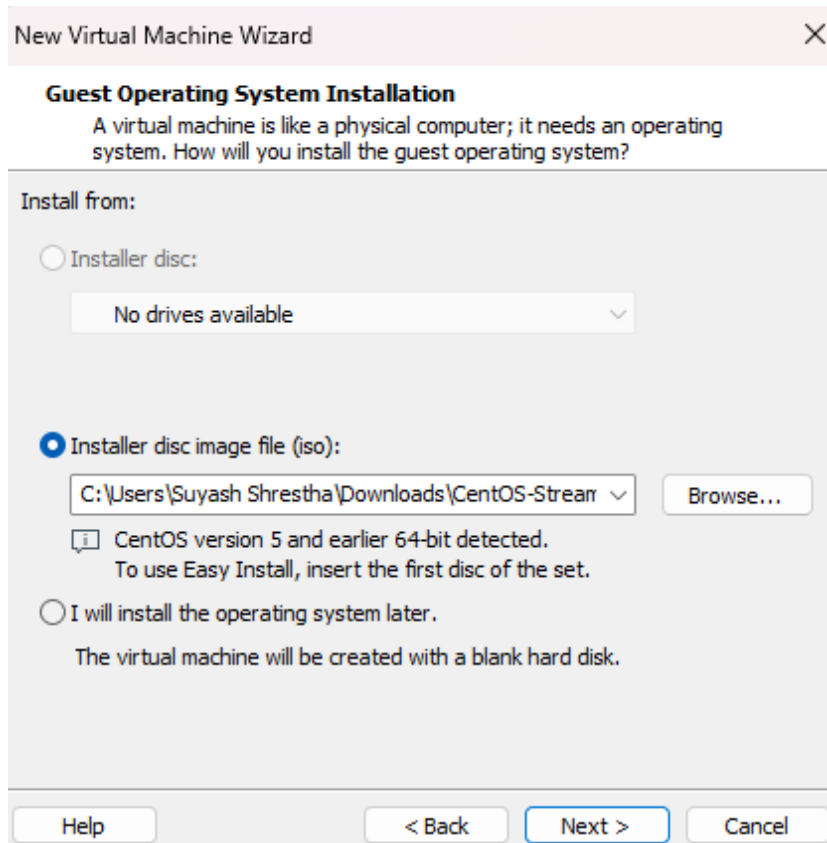
CentOS (Community ENTERprise Operating System) is a Linux distribution derived from Red Hat Enterprise Linux (RHEL). It is widely used for server applications due to its stability, security and enterprise-grade features. CentOS is commonly deployed in virtualized environments for testing, development and production use.

3. Installation Steps:

1. Download CentOS ISO Image.
2. Create a New Virtual Machine.



3. Choose the Operating System (CentOS downloaded earlier).




New Virtual Machine Wizard

Guest Operating System Installation
A virtual machine is like a physical computer; it needs an operating system. How will you install the guest operating system?

Install from:

☐ Installer disc:
No drives available

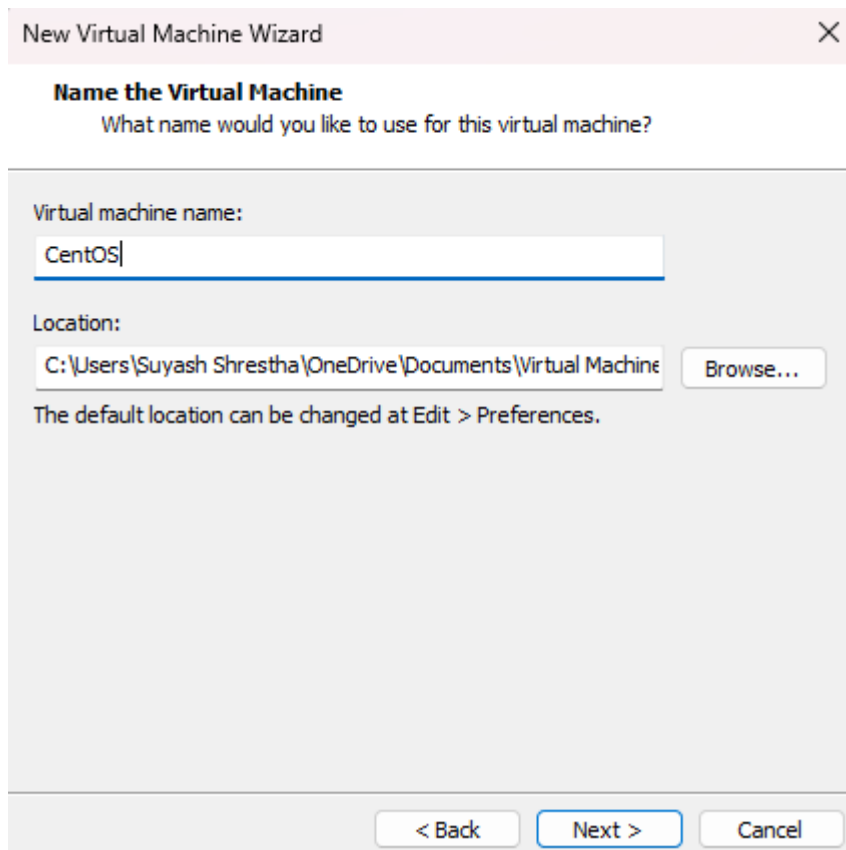
☒ Installer disc image file (iso):
C:\Users\Suyash Shrestha\Downloads\CentOS-Stream
Browse...

 CentOS version 5 and earlier 64-bit detected.
To use Easy Install, insert the first disc of the set.

☐ I will install the operating system later.
The virtual machine will be created with a blank hard disk.

Help < Back Next > Cancel

4. Configure the name and location for the new VM.



New Virtual Machine Wizard

Name the Virtual Machine
What name would you like to use for this virtual machine?

Virtual machine name:
CentOS

Location:
C:\Users\Suyash Shrestha\OneDrive\Documents\Virtual Machine
Browse...

The default location can be changed at Edit > Preferences.

< Back Next > Cancel

5. Configure the memory and processors for the new VM.

New Virtual Machine Wizard

Memory for the Virtual Machine
How much memory would you like to use for this virtual machine?

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

Memory for this virtual machine: MB

128 GB -
64 GB -
32 GB -
16 GB -
8 GB -
4 GB -
2 GB -
1 GB -
512 MB -
256 MB -
128 MB -
64 MB -
32 MB -
16 MB -
8 MB -
4 MB -

Maximum recommended memory:
13.2 GB

Recommended memory:
1 GB

Guest OS recommended minimum:
512 MB

Help < Back Next > Cancel

6. Configure Network Settings.

New Virtual Machine Wizard

Network Type
What type of network do you want to add?

Network connection

☐ Use bridged networking
Give the guest operating system direct access to an external Ethernet network. The guest must have its own IP address on the external network.

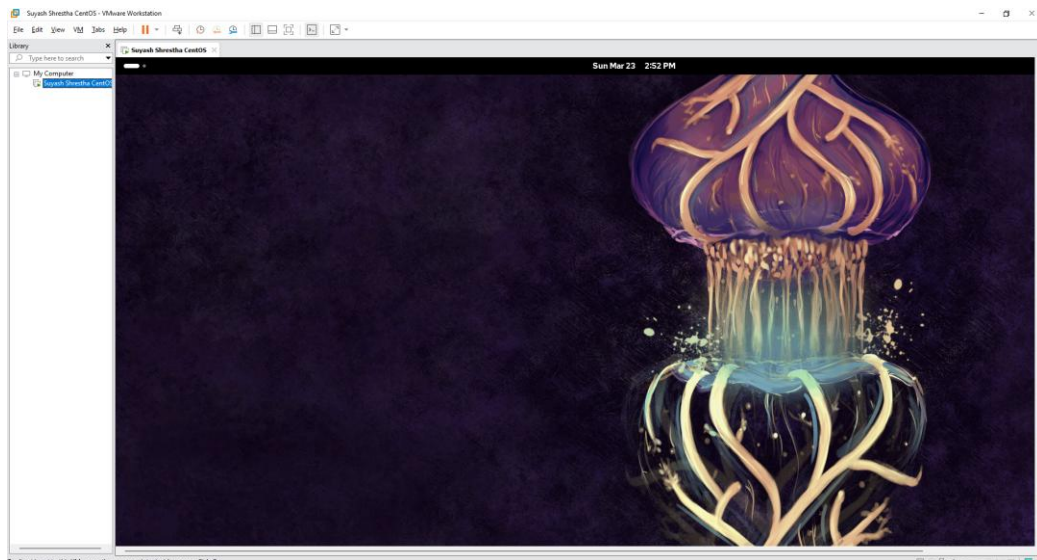
☒ Use network address translation (NAT)
Give the guest operating system access to the host computer's dial-up or external Ethernet network connection using the host's IP address.

☐ Use host-only networking
Connect the guest operating system to a private virtual network on the host computer.

☐ Do not use a network connection

Help < Back Next > Cancel

7. Install CentOS following the onscreen directions.



8. Post-Installation Configuration:

a. Check network connectivity.

```
suyash@localhost:~  
suyash@localhost:~$ ifconfig  
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.49.128 netmask 255.255.255.0 broadcast 192.168.49.255  
    inet6 fe80::20c:29ff:feef:8637 prefixlen 64 scopeid 0x20<link>  
    ether 00:0c:29:ef:86:37 txqueuelen 1000 (Ethernet)  
    RX packets 665 bytes 159207 (155.4 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 550 bytes 59935 (58.5 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 18 bytes 2112 (2.0 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 18 bytes 2112 (2.0 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
suyash@localhost:~$ s
```

b. Verify installed packages.

4. Outcome:

To gain hands-on experience in installing and configuring CentOS within a VMware environment, enhancing understanding of virtual machine management and basic network configuration.

5. Conclusion:

By following these steps, we will be able to install and configure CentOS in VMware, understanding the core aspects of virtualization and system setup.

Lab 3

Title: Capture, analyze, filter network traffic using Wire shark.

1. Objectives:

- To understand the importance of network traffic analysis.
- To learn how to capture and analyze network traffic using Wireshark.
- To familiarize with Wireshark's filtering capabilities for isolating relevant data.

2. Theory:

Wire shark:

Wireshark is a widely used network protocol analyzer that allows for real-time capture and analysis of network traffic. It provides a detailed view of data packets transmitted over a network, enabling users to inspect the protocols in use, diagnose issues and identify vulnerabilities. With Wireshark, users can capture data from various network interfaces and filter traffic to focus on specific information, making it an essential tool for network troubleshooting, security analysis, and network optimization.

3. Steps:

1. Install Wireshark on CentOS:

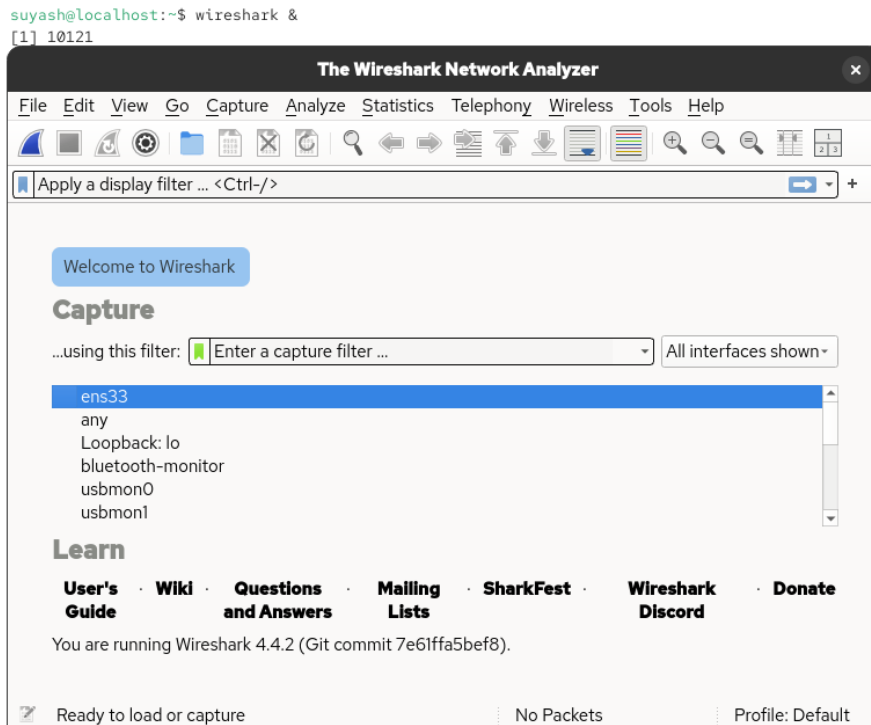
- Open the terminal and run the following commands:
 - i. `sudo yum install epel-release -y`
 - ii. `sudo yum install wireshark -y`

2. Grant Permissions to Capture Packets:

- Add the user to the Wireshark group to allow packet capturing without root access:
 - i. `sudo usermod -aG wireshark $(whoami)`
 - ii. `sudo chmod +x /usr/bin/dumpcap`
- Log out and log back in for the changes to take effect.

3. Start Wireshark:

- Open Wireshark using the GUI or start capturing via the command line:
 - i. `wireshark &`



- If using the terminal, start tshark:

i. `sudo tshark`

```
suyash@localhost:~$ sudo tshark
[sudo] password for suyash:
Running as user "root" and group "root". This could be dangerous.
Capturing on 'ens33'
```

4. Select Network Interface:

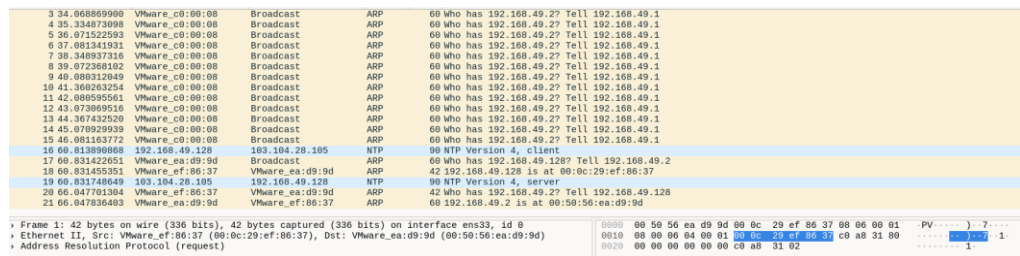
- Choose the network interface (e.g., eth0, wlan0) from which to capture packets.
- To list available interfaces, use:

i. `tshark -D`

```
suyash@localhost:~$ tshark -D
1. ens33
2. any
3. lo (Loopback)
4. bluetooth-monitor
5. usbmon0
6. usbmon1
7. usbmon2
8. nflog
9. nfqueue
10. ciscodump (Cisco remote capture)
11. dpauxmon (DisplayPort AUX channel monitor capture)
12. sdjournal (systemd Journal Export)
13. sshdump (SSH remote capture)
14. udpdump (UDP Listener remote capture)
15. wifidump (Wi-Fi remote capture)
```

5. Start Capturing Network Traffic:

- Begin packet capture by clicking the "Start" button in Wireshark,



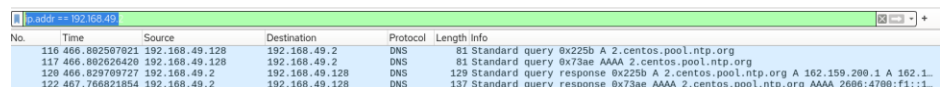
The image shows a Wireshark packet capture interface. The top pane displays a list of captured packets, primarily ARP requests from various VMware interfaces to 192.168.49.1. The middle pane shows the details of the selected packet (No. 16), identifying it as an ARP request from 192.168.49.128 to 192.168.49.1. The bottom pane shows the raw packet data in hexadecimal and ASCII, with a packet length of 28 bytes.

- Or running: `sudo tshark -i ens33`

```
suyash@localhost:~$ sudo tshark -i ens33
Running as user "root" and group "root". This could be dangerous.
Capturing on 'ens33'
  1 0.000000000 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
  2 1.266003198 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
  3 2.002652693 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
  4 3.012472031 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
  5 4.280067416 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
  6 5.003498202 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
  7 6.011442149 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
  8 7.291393354 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
  9 8.011725661 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
 10 9.004199616 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
 11 10.298562620 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
 12 11.002060039 VMware_c0:00:08 → Broadcast ARP 60 Who has 192.168.49.2? Tell 192.168.49.1
```

6. Apply Filters to Isolate Specific Traffic:

- Use filters to analyze relevant traffic:
 - Filter by IP address: `ip.addr == 192.168.1.1`



The image shows the Wireshark interface with the filter `ip.addr == 192.168.49.1` applied. The packet list shows three DNS packets (Standard query and Standard query response) between 192.168.49.128 and 192.168.49.2.

- Filter by protocol (e.g., HTTP):
 - Filter by port: `tcp.port == 80`
- In tshark, filters can be applied using:
 - `sudo tshark -i ens33 -Y "http"`

7. Analyze Captured Traffic:

- Inspect packet details, protocols and communication patterns.
- Identify potential network issues, security threats or performance bottlenecks.

8. Stop Capture and Save Data:

- Stop the capture using Ctrl+C in the terminal or click “Stop” in wireshark.
- Save the capture file for further analysis.
 - `tshark -i ens33 -w capture_file.pcap`

4. Outcome:

To gain hands-on experience in capturing and analyzing network traffic with Wireshark, enhancing understanding of network packet inspection, filtering and troubleshooting.

5. Conclusion:

By following these steps, we will be able to capture, analyze, and filter network traffic using Wireshark on CentOS, understanding the core aspects of network monitoring and security analysis.

Lab 4

Title: Design and implement VLANs using Cisco Packet Tracer and troubleshooting.

1. Objectives:

- To understand the concept and functionality of VLANs.
- To learn process of designing and configuring VLANs using Cisco Packet Tracer.
- To gain the ability to troubleshoot common VLAN configuration issues.

2. Theory:

VLAN (Virtual Local Area Network):

A VLAN is a logical subdivision of a physical network. It allows grouping of devices based on function, department, or application, regardless of their physical location. This segmentation enhances security, improves traffic management and increases network efficiency.

Cisco Packet Tracer:

Cisco Packet Tracer is a powerful network simulation tool used for learning and practicing network configurations. It allows users to design, configure, and troubleshoot network topologies in a virtual environment.

Purpose of VLANs in Networking:

In traditional LANs, all devices belong to the same broadcast domain, which can lead to excessive broadcast traffic. VLANs break up this broadcast domain by logically grouping devices. Each VLAN acts like a separate network. This leads to:

- Reduced broadcast traffic
- Better network performance
- Improved security
- Easier management

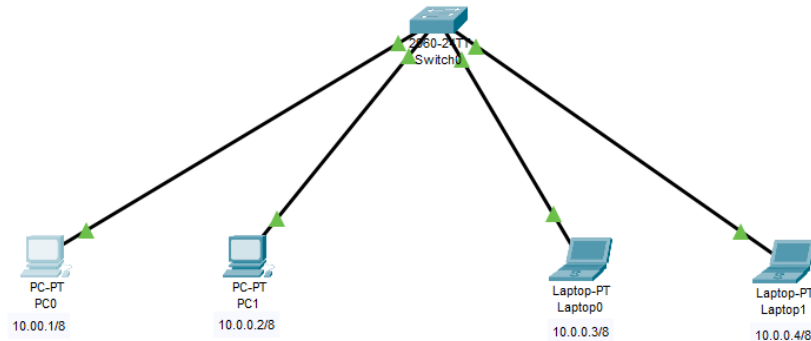
VLAN Troubleshooting:

Common issues in VLAN configuration include:

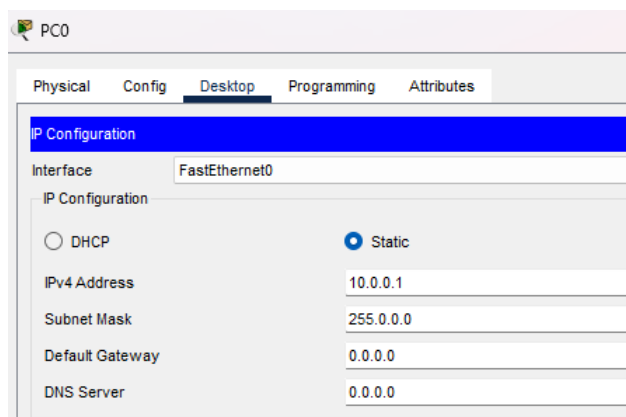
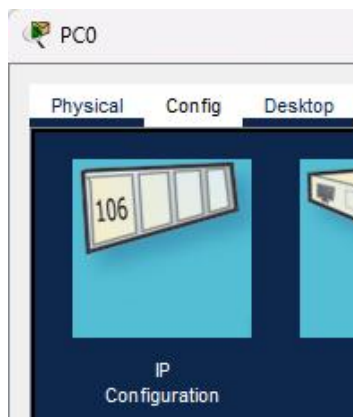
- Incorrect VLAN assignment
- Misconfigured trunk ports
- Inconsistent VLAN IDs
- Interface shutdown or incorrect mode settings

3. Steps:

1. Launch Cisco Packet Tracer and create a new project.
2. Add network devices:
 - a. Add a switch and multiple PCs to the workspace.
 - b. Connect the PCs to the switch using copper straight-through cables.



3. Assign IP Addresses:
 - a. Click on each PC.
 - b. Go to the Desktop tab > IP Configuration.
 - c. Assign IP addresses and subnet masks based on VLAN grouping.



4. Verify Connectivity:
 - a. Use the ping command to ensure basic connectivity between PCs before VLAN configuration.:

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data:

Reply from 10.0.0.3: bytes=32 time=1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time=6ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 6ms, Average = 1ms
```

5. Configure VLANs on the Switch:

- a. Click on the switch and go to the CLI tab.
- b. Enter privileged EXEC mode:
 - i. enable
- c. Enter global configuration mode:
 - i. conf t
- d. Create the VLANs:

```
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#vlan 2
Switch(config-vlan)#name office
Switch(config-vlan)#exit
Switch(config)#vlan 3
Switch(config-vlan)#name home
Switch(config-vlan)#exit
Switch(config)#
```

- e. VLAN 1 is the default VLAN, so VLANs (2 and 3) are created for office and home.

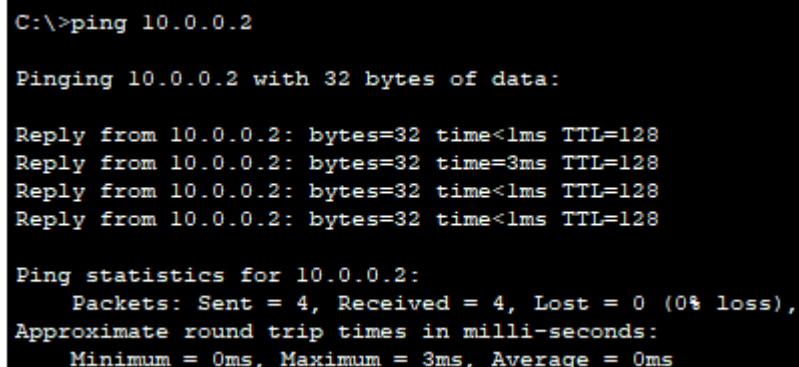
6. Assign VLANs to Interfaces:

- a. Assign switch ports to specific VLANs.

```
Switch(config)#interface fastEthernet 0/1
Switch(config-if)#switchport access vlan 2
Switch(config-if)#exit
Switch(config)#interface fastEthernet 0/2
Switch(config-if)#switchport access vlan 2
Switch(config-if)#exit
Switch(config)#interface fastEthernet 0/3
Switch(config-if)#switchport access vlan 3
Switch(config-if)#exit
Switch(config)#interface fastEthernet 0/4
Switch(config-if)#switchport access vlan 3
Switch(config-if)#exit
Switch(config)#
```

7. Test VLAN Connectivity:

- a. Use the ping command to verify communication:
 - i. PCs within the same VLAN (e.g., PC0 and PC2 in VLAN 2) should be able to communicate.:



```
C:\>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:

Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time=3ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 3ms, Average = 0ms
```


- ii. PCs in different VLANs, PC0 in VLAN 2 and Laptop0 in VLAN 3, should not be able to communicate unless inter-VLAN routing is configured.

```
C:\>ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.0.0.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

8. Troubleshoot Connectivity Issues:

- a. If issues arise:
 - i. Ensure each switch port is correctly assigned to the intended VLAN.
 - ii. Verify trunk port configurations if applicable.
 - iii. Confirm that all devices have correct IP addresses and subnet masks.
 - iv. Use commands for identifying and resolving configuration issues like:
 - show vlan brief
 - show interfaces trunk
 - show running-config

4. Outcome:

Successfully designed and implemented VLANs using Cisco Packet Tracer. Verified inter-VLAN isolation and intra-VLAN communication. Demonstrated the ability to identify and troubleshoot VLAN configuration errors in a simulated network.

5. Conclusion:

By completing this lab, we gained practical experience in creating and managing VLANs within Cisco Packet Tracer. The implementation of VLANs demonstrated enhanced network segmentation and security. Troubleshooting steps reinforced the importance of accurate configuration and verification commands for maintaining network integrity.

Lab 5

Title: Create and manage user accounts, groups and file permissions in Linux.

1. Objectives:

- To understand user and group management in Linux.
- To learn how to create, modify and delete user accounts and groups.
- To manage file permissions for users and groups effectively.

2. Theory:

User and Group management in Linux:

Linux is a multi-user operating system where users and groups help in managing access and security. Each user has a unique user ID (UID) and belongs to at least one group. Groups allow administrators to define access control more efficiently. User management commands such as `useradd`, `usermod`, and `userdel` help create, modify, and remove users, while `groupadd` and `groupdel` are used to manage groups. Proper user and group management is crucial for maintaining system security and organization.

File permissions in Linux:

File permissions are core to the security model used by Linux systems. They determine who can access files and directories on a system and how. This article provides an overview of Linux file permissions, how they work, and how to change them.

3. Steps:

1. Create Groups:

- To create new groups, use:
 - i. `sudo groupadd staff`
 - ii. `sudo groupadd sales`
 - iii. `sudo groupadd marketing`

2. View Existing Groups:

- To check all groups available in the system:
 - i. `cat /etc/group`
`staff:x:1001:`
`sales:x:1002:`
`marketing:x:1003:`

3. Create a New User:

- To create a new user named Suyash:

- i. `sudo useradd suyashshrestha`

- Set a password for the user:

- i. `sudo passwd suyashshrestha`

```
suyash@localhost:~$ sudo useradd suyashshrestha
suyash@localhost:~$ sudo passwd suyashshrestha
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: password updated successfully
```

- Switch to the new user:

- i. `su suyashshrestha`

- Verify the current working directory:

1. `pwd`

```
suyashshrestha@localhost:/home/suyash$ pwd
/home/suyash
```

4. Add User to Groups:

- Add a user to a primary group and multiple secondary groups during creation:

- i. `sudo useradd -g staff -G sales,marketing newuser`

- To add an existing user to additional groups:

- i. `sudo usermod -aG sales,marketing newuser`

- Verify the user's group membership:

- i. `grep newuser /etc/passwd`

```
suyash@localhost:~$ grep newuser /etc/passwd
newuser:x:1002:1001::/home/newuser:/bin/bash
```

- ii. `grep sales /etc/group`

```
suyash@localhost:~$ grep sales /etc/group
sales:x:1002:newuser
```

5. Modify User Details:

- To change the primary group of a user:

- i. `sudo usermod -g sales suyashshrestha`

- To check the updated group:

- i. `id suyashshrestha`

```
suyash@localhost:~$ id suyashshrestha
uid=1001(suyashshrestha) gid=1002(sales) groups=1002(sales)
```

6. Delete a User:

- To remove a user and their home directory:

- i. `sudo userdel -r suyashshrestha`

7. File permissions management:

- Check file permissions:
 - i. To view permissions of file:
`ls -l suyash.txt`
- Change file permissions:
 - i. Grant read, write and execute permissions to the owner:
`chmod u+rwX suyash.txt`
 - ii. Remove write permissions for the group:
`Chmod g-w suyash.txt`
- Change file ownership:
 - i. Change the owner of a file:
`sudo chown newuser suyash.txt`
 - ii. Change both owner and group:
`sudo chown newuser:staff suyash.txt`
- Change group ownership of a file:
 - i. To assign a file to a group:
`sudo chgrp marketing suyash.txt`

4. Outcome:

To gain hands-on experience in creating and managing user accounts, groups and file permissions in Linux, improving system administration skills.

5. Conclusion:

By following these steps, we will be able to efficiently manage user accounts and groups in Linux, ensuring proper access control and system security.

Lab 6

Title: Configure network interfaces, assign static and dynamic IPs, enable/disable network services and troubleshoot connectivity issues

1. Objectives:

- To understand the configuration of network interfaces in Linux.
- To learn how to assign static and dynamic IP addresses.
- To enable and disable network services as needed.
- To troubleshoot connectivity issues using network diagnostic tools.

2. Theory:

Network Configuration in Linux:

Linux provides various tools and commands to configure network interfaces, assign IP addresses, and manage network connectivity. The `ifconfig` or `ip` command is used to view and modify network interfaces, while `ping`, `traceroute` and `netstat` help diagnose connectivity problems. Network services such as DHCP and DNS can be enabled or disabled depending on the network requirements. Configuring network interfaces correctly is crucial for establishing reliable communication between devices.

3. Steps:

1. View Network Interface Information:

- Check network interfaces and assigned IP addresses:

i. `ifconfig`

```
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.49.128 netmask 255.255.255.0 broadcast 192.168.49.255
    inet6 fe80::20c:29ff:feef:8637 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:ef:86:37 txqueuelen 1000 (Ethernet)
    RX packets 66285 bytes 85793174 (81.8 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 16817 bytes 1244952 (1.1 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 159 bytes 1433772 (1.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 159 bytes 1433772 (1.3 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

2. Assign a Static IP Address:

- Open the network configuration file:
 - i. `sudo nano /etc/sysconfig/network-scripts/ifcfg-ens33`
- Modify the file to include the following:
 - i. `BOOTPROTO=static`
`IPADDR=192.168.1.100`
`NETMASK=255.255.255.0`
`GATEWAY=192.168.1.1`
`DNS1=8.8.8.8`
- Save the file and restart the network service:
 - i. `sudo systemctl restart network`

3. Assign a Dynamic IP Address (Using DHCP):

- Modify the network configuration file:
 - i. `sudo nano /etc/sysconfig/network-scripts/ifcfg-ens33`
- Set `BOOTPROTO=dhcp` and save the file.
- Restart the network service:
 - i. `sudo systemctl restart network`

4. Enable and Disable Network Services:

- To enable the network service at startup:
 - i. `sudo systemctl enable NetworkManager`
- To disable a network service:
 - i. `sudo systemctl stop NetworkManager`

5. Troubleshooting Network Connectivity Issues:

- Check connectivity with a remote host:
 - i. `ping -c 4 8.8.8.8`


```
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.  
64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=34.8 ms  
64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=33.2 ms  
64 bytes from 8.8.8.8: icmp_seq=3 ttl=128 time=32.2 ms  
64 bytes from 8.8.8.8: icmp_seq=4 ttl=128 time=31.9 ms  
  
--- 8.8.8.8 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3006ms  
rtt min/avg/max/mdev = 31.889/33.012/34.760/1.121 ms
```
- Trace the route to a destination:
 - i. `tracert google.com`

```
Tracing route to google.com [2404:6800:4002:80c::200e]  
over a maximum of 30 hops:  
  0  0 ms  0 ms  0 ms  2400:1a00:4b27:99ec::1  
  1  36 ms  6 ms  24 ms  2400:1a00:4b02:ip6.wlink.com.np [2400:1a00:4b02::1]  
  2  *  *  *  Request timed out.  
  3  16 ms  8 ms  8 ms  2400:1a00:0:41::136  
  4  9 ms  7 ms  8 ms  2400:1a00:0:41::139  
  5  14 ms  17 ms  6 ms  2400:1a00:dccc:1:72:9:128:67  
  6  *  *  *  Request timed out.  
  7  32 ms  26 ms  24 ms  2001:4860:1:1::2b5a  
  8  28 ms  25 ms  33 ms  2001:4860:0:1::78d3  
  9  *  *  *  Request timed out.  
 10  30 ms  23 ms  29 ms  del03s17-in-x0e.1e100.net [2404:6800:4002:80c::200e]  
Trace complete.
```

- Check open network connections and listening ports:

- i. `netstat -tulnp`

```
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:53              0.0.0.0:*               LISTEN      -
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 :::1:631                :::*                    LISTEN      -
tcp6       0      0 :::53                   :::*                    LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
tcp6       0      0 :::80                   :::*                    LISTEN      -
tcp6       0      0 :::9090                  :::*                    LISTEN      -
udp        0      0 0.0.0.0:5353            0.0.0.0:*               -           -
udp        0      0 0.0.0.0:53              0.0.0.0:*               -           -
udp        0      0 0.0.0.0:67              0.0.0.0:*               -           -
udp        0      0 127.0.0.1:323           0.0.0.0:*               -           -
udp6       0      0 :::5353                  :::*                    -           -
udp6       0      0 :::53                   :::*                    -           -
udp6       0      0 :::1:323                 :::*                    -           -
```

- Look up DNS resolution for a domain:

- i. `nslookup google.com`

```
Server:          192.168.49.2
Address:         192.168.49.2#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.194.110
Name:   google.com
Address: 2404:6800:4002:824::200e
```

- View the current routing table:

- i. `route -n`

```
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0        192.168.49.2   0.0.0.0         UG    100    0      0 ens33
192.168.49.0   0.0.0.0        255.255.255.0   U     100    0      0 ens33
```

4. Outcome:

To gain hands-on experience in configuring network interfaces, assigning static and dynamic IP addresses, managing network services and troubleshooting connectivity issues in Linux.

5. Conclusion:

By following these steps, we will be able to effectively configure network settings, enable and disable network services and diagnose connectivity issues, ensuring stable and secure network communication in a Linux environment.

Lab 7

Title: Implement firewall rules to control inbound and outbound traffic and test policies.

1. Objectives:

- To understand the importance of firewall rules in securing network traffic.
- To learn how to configure firewall rules using firewalld on CentOS.
- To test firewall policies by allowing and denying specific services.

2. Theory:

Firewall:

A firewall is a network security system that monitors and controls incoming and outgoing network traffic, acting as a barrier between a private network and the internet, blocking unauthorized access and malicious traffic.

Firewall Configuration Using Firewalld:

Firewalld is a dynamic firewall management tool used in CentOS to control inbound and outbound network traffic. It allows users to define rules that permit or restrict access to network services, enhancing system security. Firewalld operates with zones, where different trust levels can be assigned to interfaces, and rules can be configured to allow or block specific services, ports, and protocols. Implementing firewall rules helps protect systems from unauthorized access while allowing legitimate traffic.

3. Steps:

1. Install and Start Firewalld:

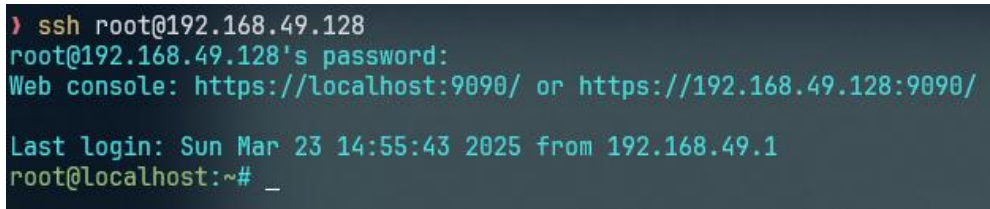
- Open the terminal and install Firewalld:
 - i. `sudo yum install firewalld -y`
- Start and enable the firewall service:
 - i. `sudo systemctl start firewalld`
 - ii. `sudo systemctl enable firewalld`

2. Allow Common Services (SSH, HTTP, HTTPS):

- Allow SSH (Port 22):
 - i. `sudo firewall-cmd --permanent --add-service=ssh`
- Allow HTTP (Port 80):
 - i. `sudo firewall-cmd --permanent --add-service=http`

- Allow HTTPS (Port 443):
 - i. `sudo firewall-cmd --permanent --add-service=https`
- 3. Apply Changes and Reload Firewall:
 - Reload the firewall to apply the new rules:
 - i. `sudo firewall-cmd --reload`
- 4. Verify Configured Firewall Rules:
 - List all allowed services:
 - i. `sudo firewall-cmd --list-services`

```
-----
suyash@localhost:~$ sudo firewall-cmd --list-services
cockpit dhcpv6-client http https ssh
```
 - List open ports:
 - i. `sudo firewall-cmd --list-ports`

```
suyash@localhost:~$ sudo firewall-cmd --list-ports
80/tcp
```
- 5. Test Firewall Rules:
 - Test SSH access (should be allowed):
 - i. `ssh 192.168.49.128`
 - Test HTTP and HTTPS access by opening a web browser and navigating to:
 - i. `http://192.168.126.133`
 - ii. `https://192.168.126.133`
 - Test FTP access (should be denied):
 - i. `ftp 192.168.126.133`

4. Outcome:

To gain hands-on experience in configuring and testing firewall rules using Firewalld on CentOS, enhancing understanding of network security and traffic control.

5. Conclusion:

By following these steps, we will be able to implement firewall rules to control inbound and outbound traffic on CentOS, ensuring secure network access while restricting unauthorized connections.

Lab 8

Title: Setup DNS servers, configure domain name resolution and troubleshoot common issues

1. Objectives:

- To understand DNS server setup and configuration.
- To configure domain name resolution using local and external DNS servers.
- To troubleshoot common DNS issues and ensure proper name resolution.

2. Theory:

Domain Name System (DNS) in Linux:

The DNS is responsible for translating human-readable domain names into IP addresses. Linux systems rely on `/etc/resolv.conf` for DNS resolution and can use `bind` to set up a local DNS server. `NetworkManager` can be used to configure persistent DNS settings. Common troubleshooting tools like `nslookup`, `dig` and `ping` help diagnose DNS-related issues. Proper DNS configuration ensures stable and efficient network communication.

3. Steps:

1. Check Current DNS Configuration:

- View the active DNS servers:
 - i. `cat /etc/resolv.conf`

```
# Generated by NetworkManager
search localdomain
nameserver 192.168.49.2
```

2. Configure DNS Servers (Client-Side):

- Temporarily Modify DNS Settings:
 - i. `sudo nano /etc/resolv.conf`
- Add the following lines:
 - i. `nameserver 8.8.8.8`
`nameserver 1.1.1.1`
- This change is temporary and will be lost after a reboot.
- Set Permanent DNS Using `NetworkManager`:
 - i. `sudo nmcli connection modify ens33 ipv4.dns "8.8.8.8 1.1.1.1"`
 - ii. `sudo nmcli connection up ens33.`

```
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/3)
```

- iii. `sudo systemctl restart NetworkManager`

3. Set Up a Local DNS Server (Using BIND):

- Install the BIND package:
 - i. `sudo dnf install -y bind bind-utils`
- Modify the DNS configuration file:
 - i. `sudo nano /etc/named.conf`
- Update the file with:
 - i. `listen-on port 53 { 127.0.0.1; 192.168.1.1; };`
 - ii. `allow-query { localhost; 192.168.1.0/24; };`
- Start and enable the DNS service:
 - i. `sudo systemctl enable --now named`

4. Test and Troubleshoot DNS Issues:

- Check if DNS resolution is working:
 - i. `nslookup google.com`

```
Server:      8.8.8.8
Address:     8.8.8.8#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.194.14
Name:   google.com
Address: 2404:6800:4002:824::200e
```
 - ii. `dig google.com`

```
; <<>> DiG 9.18.33 <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 58231
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;google.com.                IN      A

;; ANSWER SECTION:
google.com.                 109     IN      A      142.250.206.142

;; Query time: 22 msec
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Sun Mar 30 21:48:11 +0545 2025
;; MSG SIZE rcvd: 55
```
- Verify if the DNS service is running:
 - i. `sudo systemctl status named`
- Restart the DNS service if necessary:
 - i. `sudo systemctl restart named`

- Test network connectivity:
 - i. `ping -c 4 8.8.8.8`

```
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.  
64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=34.8 ms  
64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=33.2 ms  
64 bytes from 8.8.8.8: icmp_seq=3 ttl=128 time=32.2 ms  
64 bytes from 8.8.8.8: icmp_seq=4 ttl=128 time=31.9 ms  
  
--- 8.8.8.8 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3006ms  
rtt min/avg/max/mdev = 31.889/33.012/34.760/1.121 ms
```
 - Prevent /etc/resolv.conf from being overwritten:
 - i. `sudo chattr +i /etc/resolv.conf`
 - ii. `sudo chattr -i /etc/resolv.conf`

4. Outcome:

To gain hands-on experience in setting up DNS servers, configuring domain name resolution and troubleshooting common DNS issues, ensuring a stable network environment.

5. Conclusion:

By following these steps, we will be able to configure and manage DNS services effectively, ensuring proper name resolution and network connectivity in a Linux environment.

Lab 9

Title: Setup DHCP servers and troubleshoot common networking issues.

1. Objectives:

- To understand the purpose and function of a DHCP server.
- To learn how to install, configure, and manage a DHCP server using dnsmasq on CentOS 10.
- To troubleshoot common networking issues related to IP address assignment.

2. Theory:

DHCP Server:

A DHCP (Dynamic Host Configuration Protocol) server automatically assigns IP addresses and network configurations to client devices within a network. This eliminates the need for manual IP assignment and helps manage large networks efficiently.

Key Functions of DHCP:

- Assigning dynamic IP addresses to clients.
- Providing additional network information, such as subnet mask, default gateway and DNS servers.
- Reducing IP conflicts and simplifying network management.

Common Networking Issues:

- IP Address Conflicts: When two devices have the same IP address, network connectivity issues occur.
- Incorrect Network Configuration: Wrong subnet mask, gateway, or DNS settings can cause connectivity failures.
- DHCP Server Unavailability: If the DHCP server is down, clients won't receive IP addresses, leading to network failures.

3. Steps:

1. Install dnsmasq for DHCP:

Since CentOS 10 doesnot provide dhcp-server, dnsmasq is used which functions as both a DHCP and DNS Server.

- a. Install dnsmasq:

```
sudo yum install dnsmasq -y
```

2. Configure the DHCP Server (dnsmasq).

- a. Open the dnsmasq configuration file:

```
sudo nano /etc/dnsmasq.conf
```

- b. Add the following configuration:

```
interface=ens33
dhcp-range=192.168.1.50,192.168.1.100,12h # IP range for clients
dhcp-option=3,192.168.1.1 # Gateway IP address
dhcp-option=6,8.8.8.8,8.8.4.4 # DNS servers
```

- c. Save and exit the file.

3. Start and Enable the DHCP Service.

```
sudo systemctl enable --now dnsmasq
```

4. Restart Network Services.

- a. Restart the network service to apply the new configuration:

```
sudo systemctl restart NetworkManager
```

- b. Verify network interface configuration:

```
ip addr show
suyash@localhost:~$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:ef:86:37 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    altname enx000c29ef8637
    inet 192.168.49.128/24 brd 192.168.49.255 scope global dynamic noprefixroute ens33
        valid_lft 1779sec preferred_lft 1779sec
    inet6 fe80::20c:29ff:feef:8637/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

4. Outcome:

To gain hands-on practical knowledge in installing and configuring a DHCP server using dnsmasq on CentOS 10, while troubleshooting common networking issues related to IP address assignment. This will help establish network connectivity and ensure efficient IP management across a network.

5. Conclusion:

By following these steps, we were able to successfully install and configure a dnsmasq DHCP server, resolving networking issues related to dynamic IP assignments. This practical experience enhances our understanding of DHCP and its critical role in managing network configurations in virtualized environments.

Lab 10

Title: Install and configure Apache/Nginx web servers.

1. Objectives:

- To learn how to install and configure Apache and Nginx web servers.
- To understand the basic differences between Apache and Nginx.
- To deploy and manage a basic web server environment.

2. Theory:

Web Servers:

Web servers are software applications that serve web pages to users over the internet or a local network. They process HTTP requests and deliver HTML, CSS, JavaScript and other files to web browsers.

Apache Web Server:

Apache is one of the most widely used open-source web servers. It follows a process-driven architecture, handling multiple connections using a multi-threaded approach. Apache is known for its extensive module support, flexibility, and compatibility with various operating systems.

Nginx Web Server:

Nginx is a high-performance web server designed to handle a large number of simultaneous connections efficiently. It follows an event-driven architecture, making it ideal for handling static content and acting as a reverse proxy for load balancing. Nginx is often preferred for modern web applications due to its speed and scalability.

3. Steps:

1. For Apache Web Server:

- a. Install Apache.

```
sudo yum install httpd
```
- b. Start Apache:

```
sudo systemctl start httpd
```
- c. Enable Apache to start on boot:

```
sudo systemctl enable httpd
```
- d. Configure Apache (if required):

```
sudo nano /etc/httpd/conf/httpd.conf
```

- e. Modify the web page:

```
sudo nano /var/www/html/index.html
```

Insert html content like:

```
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to Suyash's site</title>
</head>
<body>
  <h1>I am Suyash Shrestha</h1>
  <p>Hello!!! </p>
</body>
</html>
```

- f. Set permissions for Apache:

```
sudo chown -R apache:apache /var/www/html
sudo chmod -R 555 /var/www/html
```

- g. Restart Apache:

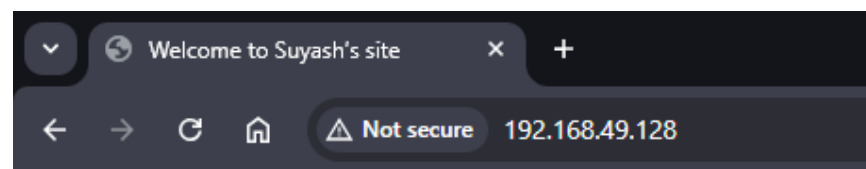
```
sudo systemctl restart httpd
```

- h. Allow Apache through the VM's firewall:

```
sudo firewall-cmd -zone=public -add-service=http -permanent
sudo firewall-cmd --reload
```

- i. Open browser outside the VM and Test Apache (enter the ip of the VM):

http://192.168.49.128 or http://localhost # for inside VM



I am Suyash Shrestha

Hello!!!

2. For Nginx Web Servers:

- a. Install Nginx:

```
sudo yum install nginx
```

- b. Start Nginx:

```
sudo systemctl start nginx
```

- c. Enable Nginx to start on boot:

```
sudo systemctl enable nginx
```

- d. Configure Nginx: (make necessary changes like ports, if needed)

```
sudo nano /etc/nginx/nginx.conf
```


- e. Modify the web page:

```
sudo nano /usr/share/nginx/html/index.html
```

Insert HTML content:

```
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to Suyash's site</title>
</head>
<body>
  <h1>I am Suyash Shrestha</h1>
  <p>Hello from nginx!!! </p>
</body>
</html>
```

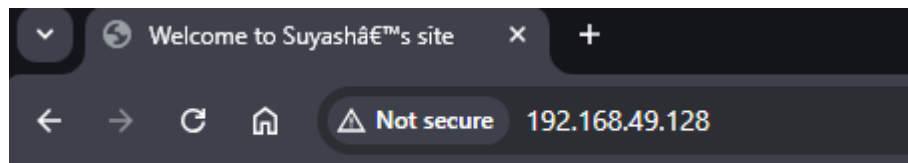
- f. Set permissions for Nginx:

```
sudo chown -R nginx:nginx /usr/share/nginx/html
sudo chmod -R 755 /usr/share/nginx/html
```

- g. Restart Nginx:

```
sudo systemctl restart nginx
```

- h. Open browser and test nginx:



I am Suyash Shrestha

Hello from nginx!!!

4. Outcome:

To gain hands-on experience in installing, configuring and running Apache and Nginx web servers along with understanding how web servers' function, their differences and how to deploy a basic website.

5. Conclusion:

This lab demonstrated the installation and basic configuration of both Apache and Nginx web servers. By completing this lab, we will be able to set up and manage web servers, making us ready for more advanced configurations like virtual hosts, SSL setup and load balancing.

Lab 11

Title: Install and configure Squid proxy servers

1. Objectives:

- To install, configure and understand the role and functionality of Squid Proxy.
- To learn how to use Squid for caching and access control.
- To monitor and troubleshoot Squid operations.

2. Theory:

Proxy servers:

A proxy server is a system or router that provides a gateway between users and the internet. Therefore, it helps prevent cyber attackers from entering a private network. It is a server, referred to as an “intermediary” because it goes between end-users and the web pages they visit online.

Squid proxy servers:

Squid is a caching and forwarding proxy that improves web performance, optimizes bandwidth usage and enhances security by filtering and controlling traffic. It supports HTTP, HTTPS and FTP protocols and is widely used by enterprises, ISPs and educational institutions. Key features include caching frequently accessed content, implementing access control policies, load balancing and blocking malicious websites.

3. Steps:

1. Install Squid Proxy:

- Update system packages:
 - i. `sudo yum update -y`
- Install Squid:
 - i. `sudo yum install squid -y`
- Enable Squid to start on boot:
 - i. `sudo systemctl enable squid`
- Start Squid service:
 - i. `sudo systemctl start squid`
- Verify Squid status:
 - i. `sudo systemctl status squid`

```
● squid.service - Squid caching proxy
   Loaded: loaded (/usr/lib/systemd/system/squid.service; enabled; preset: disabled)
   Active: active (running) since Mon 2025-03-31 06:53:49 +0545; 7s ago
     Invocation: 8336c343ef8643fc878f0bc101d92731
```

2. Configure Squid Proxy:

- Backup the default configuration file:
 - i. `sudo cp /etc/squid/squid.conf /etc/squid/squid.conf.bak`
- Edit the Squid configuration file:
 - i. `sudo nano /etc/squid/squid.conf`
- Allow specific IP ranges (Example: 192.168.49.0/24 network):

Add the following lines before the `http_access deny all` directive:

- i.

```
acl localnet src 192.168.49.0/24
http_access allow localnet
```

- Modify the Squid listening port (if needed):

Find this line in `squid.conf`:

- i. `http_port 3128`

Modify it to another port if required, such as:

- ii. `http_port 8080`

Port changes help avoid conflicts with other services.

- Restart Squid to apply configuration changes:
 - i. `sudo systemctl restart squid`
- Check if Squid is listening on the specified port:

- i. `sudo netstat -tulnp | grep squid`

```
tcp6      0      0 :::3128          :::*              LISTEN      4535/(squid-1
udp       0      0 0.0.0.0:39462    0.0.0.0:*         4535/(squid-1
udp6      0      0 :::40389         :::*              4535/(squid-1
```

3. Using the Squid Service:

- Monitor Squid logs for requests:
 - i. `sudo tail -f /var/log/squid/access.log`
- Clear Squid cache (if needed):
 - i.

```
sudo squid -k shutdown
sudo rm -rf /var/spool/squid/
sudo squid -z
sudo systemctl start squid
```
- Test Squid Proxy:
 - i. Configure a web browser or system proxy settings to `http://<Squid_Server_IP>:3128`
 - ii. Browse the web and verify requests in Squid logs.

4. Troubleshooting Squid Proxy:

- Check if Squid is running:
 - i. `sudo systemctl status squid`

- Validate Squid configuration syntax:
 - i. `sudo squid -k check`
- Restart Squid if necessary:
 - i. `sudo systemctl restart squid`
- Analyze Squid logs for troubleshooting:
 - i. `sudo cat /var/log/squid/access.log`

4. Outcome:

To gain hands-on experience in installing and configuring Squid Proxy, enabling caching and access control and troubleshooting its operation for optimized network performance.

5. Conclusion:

By following these steps, we will be able to deploy and configure Squid Proxy, understanding its key functionalities such as caching, security and traffic control. Proper maintenance and troubleshooting of Squid ensure efficient, secure network management.

Lab 12

Title: Install, Configure and Test MySQL Server.

1. Objectives:

- To learn how to install and configure MySQL Server on a Linux-based system.
- To understand how to create and manage databases and tables in MySQL.
- To execute basic SQL queries to store and retrieve data.

2. Theory:

MySQL:

MySQL is a widely used open-source relational database management system that utilizes Structured Query Language for managing and manipulating data. It has grown in popularity due to its reliability, speed and ease of use. MySQL supports various OS making it a versatile choice for many applications. It offers the software to fit user's specific needs, while also offering enterprise versions for larger-scale deployments.

3. Steps:

1. Update the system:

```
sudo yum update
```

2. Install MySQL Server:

```
sudo yum install mysql-server
```

```
suyashshrestha@localhost:~$ sudo yum install mysql-server
[sudo] password for suyashshrestha:
Updating Subscription Management repositories.
Unable to read consumer identity
```

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

Last metadata expiration check: 0:19:04 ago on Sun 02 Mar 2025 06:56:49 AM +0545.
Dependencies resolved.

Package	Arch	Version	Repository	Size
Installing:				
mysql-server	x86_64	8.4.2-5.el10	appstream	18 M
Installing dependencies:				
mariadb-connector-c-config	noarch	3.4.4-1.el10	baseos	9.9 k
mecab	x86_64	0.996-9.el10	appstream	363 k
mysql	x86_64	8.4.2-5.el10	appstream	2.4 M
mysql-common	noarch	8.4.2-5.el10	appstream	75 k
mysql-errmsg	noarch	8.4.2-5.el10	appstream	529 k
mysql-selinux	noarch	1.0.13-2.el10	appstream	38 k
protobuf-lite	x86_64	3.19.6-11.el10	appstream	262 k

Transaction Summary

Install 8 Packages

```

Total download size: 22 M
Installed size: 164 M
Is this ok [y/N]: y
Downloading Packages:
(1/8): mariadb-connector-c-config-3.4.4-1.el10. 16 kB/s | 9.9 kB 00:00
(2/8): mysql-common-8.4.2-5.el10.noarch.rpm 31 kB/s | 75 kB 00:02
(3/8): mecab-0.996-9.el10.x86_64.rpm 84 kB/s | 363 kB 00:04
(4/8): mysql-selinux-1.0.13-2.el10.noarch.rpm 23 kB/s | 38 kB 00:01
(5/8): mysql-errmsg-8.4.2-5.el10.noarch.rpm 30 kB/s | 529 kB 00:17
(6/8): protobuf-lite-3.19.6-11.el10.x86_64.rpm 73 kB/s | 262 kB 00:03
(7/8): mysql-8.4.2-5.el10.x86_64.rpm 24 kB/s | 2.4 MB 01:42
(8/8): mysql-server-8.4.2-5.el10.x86_64.rpm 163 kB/s | 18 MB 01:54
-----
Total 181 kB/s | 22 MB 02:03

Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing : 1/1
  Installing : mariadb-connector-c-config-3.4.4-1.el10.noarch 1/8
  Installing : mysql-common-8.4.2-5.el10.noarch 2/8
  Installing : mysql-8.4.2-5.el10.x86_64 3/8
  Installing : mysql-errmsg-8.4.2-5.el10.noarch 4/8
  Installing : protobuf-lite-3.19.6-11.el10.x86_64 5/8
  Running scriptlet: mysql-selinux-1.0.13-2.el10.noarch 6/8
  Installing : mysql-selinux-1.0.13-2.el10.noarch 6/8
  Running scriptlet: mysql-selinux-1.0.13-2.el10.noarch 6/8
libsemanage.semanage_direct_install_info: Overriding mysql module at lower priority 100 with module at priority 200.

  Installing : mecab-0.996-9.el10.x86_64 7/8
  Running scriptlet: mysql-server-8.4.2-5.el10.x86_64 8/8
  Installing : mysql-server-8.4.2-5.el10.x86_64 8/8
  Running scriptlet: mysql-server-8.4.2-5.el10.x86_64 8/8
  Running scriptlet: mysql-selinux-1.0.13-2.el10.noarch 8/8
  Running scriptlet: mysql-server-8.4.2-5.el10.x86_64 8/8
Installed products updated.

Installed:
mariadb-connector-c-config-3.4.4-1.el10.noarch
mecab-0.996-9.el10.x86_64
mysql-8.4.2-5.el10.x86_64
mysql-common-8.4.2-5.el10.noarch
mysql-errmsg-8.4.2-5.el10.noarch
mysql-selinux-1.0.13-2.el10.noarch
mysql-server-8.4.2-5.el10.x86_64
protobuf-lite-3.19.6-11.el10.x86_64

Complete!
suyashshrestha@localhost:~$ █

```

3. Start the MySQL service:

```
sudo systemctl start mysqld
```

4. Enable MySQL to start on boot:

```
sudo systemctl enable mysqld
```

```
suyashshrestha@localhost:~$ sudo systemctl enable mysqld
Created symlink '/etc/systemd/system/multi-user.target.wants/mysqld.service' → '/usr/lib/systemd/system/mysqld.service'.
```

5. Secure the MySQL installation:

```
sudo mysql_secure_installation
```

Follow the prompts to set up a root password, remove anonymous users, disable remote root login, and remove the test database.

```

suyashshrestha@localhost:~$ sudo mysql_secure_installation
[sudo] password for suyashshrestha:

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.
The 'validate_password' component is installed on the server.
The subsequent steps will run with the existing configuration
of the component.
Please set the password for root here.

New password:

Re-enter new password:

Estimated strength of the password: 50
Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) : Y
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : n

... skipping.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : n

... skipping.
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!

```

6. Log in to MySQL:

```
mysql -u root -p
```

```

suyashshrestha@localhost:~/Desktop$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 13
Server version: 8.4.2 Source distribution

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

```

7. Create a new database:

```
CREATE DATABASE suyashprime;
```

```

mysql> CREATE DATABASE suyashprime;
Query OK, 1 row affected (0.01 sec)

```

8. Use a newly created database and create a table:

```
USE suyashprime;

CREATE TABLE students ( id INT PRIMARY KEY, name VARCHAR(255));

mysql> USE suyashprime;
Database changed
mysql> CREATE TABLE students ( id INT PRIMARY KEY, name VARCHAR(255) );
Query OK, 0 rows affected (0.19 sec)
```

9. Insert a record into the table:

```
INSERT INTO students VALUES (1, 'Suyash');

mysql> INSERT INTO students VALUES (1, 'Suyash');
Query OK, 1 row affected (0.03 sec)
```

10. Retrieve and display the data:

```
SELECT * FROM students;

mysql> SELECT * FROM students;
+----+-----+
| id | name  |
+----+-----+
|  1 | Suyash |
+----+-----+
1 row in set (0.01 sec)
```

11. Exit MySQL:

```
exit

mysql> exit
Bye
```

4. Outcome:

To successfully install and configure MySQL Server within CentOS, understanding how to create a database, tables and various other operations in MySQL like INSERT and SELECT.

5. Conclusion:

By following these instructions, installation, configuration and basic usage of MySQL Server on a Linux system was performed successfully. We successfully set up a MySQL database, created a table, and performed simple operations such as inserting and retrieving data.

Lab 13

Title: Install and Configure mail servers (Postfix/Sendmail), enable SMTP, IMAP/POP3 services

1. Objectives:

- To understand the basic functioning of email communication over a network.
- To configure a mail server using Cisco Packet Tracer.
- To enable SMTP and POP3 services for email exchange.
- To verify email delivery and reception between client PCs.

2. Theory:

Mail Server:

A mail server is a software system that sends, receives, and stores email. It typically includes a Mail Transfer Agent (MTA), a Mail Delivery Agent (MDA), and protocols like SMTP, IMAP, and POP3 for communication between clients and servers.

Postfix and Sendmail:

Postfix is a fast and secure open-source Mail Transfer Agent (MTA) widely used in Linux environments. Sendmail is an older but powerful MTA that supports robust configuration and mail routing. Both MTAs handle sending and routing of emails between systems over the internet or internal networks.

SMTP (Simple Mail Transfer Protocol):

It is used for sending emails from clients to servers or between servers and uses port 25.

IMAP (Internet Message Access Protocol):

It allows users to access and manage their email on the server without downloading it.

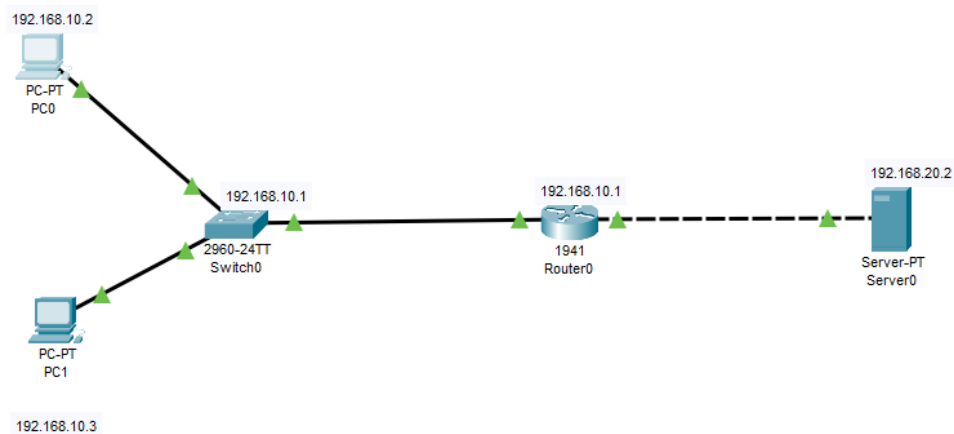
POP3 (Post Office Protocol version 3):

It allows users to download emails from the server to their local device and usually deletes it from the server afterward and uses port 110.

3. Steps:

1. Open Cisco Packet Tracer and create a new workspace.
2. Drag and drop the following devices into the workspace:
 - a. 1 Generic Server
 - b. 2 PCs (PC0 and PC1)
 - c. 1 Switch and 1 Router.

3. Connect devices using copper straight-through cables:

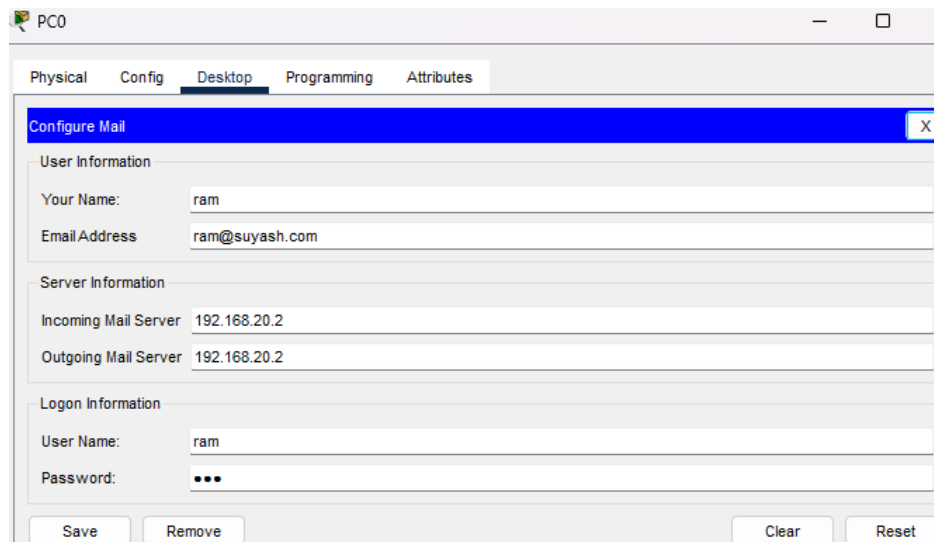


4. Assign IPv4 Address (from config tab) to the Router and turn on Port Status:
- Set GigabitEthernet0/0 IPv4 address to 192.168.10.1
 - Set GigabitEthernet0/1 IPv4 address to 192.168.20.1
5. Assign static IP addresses to each device (PCs and Server):
- Server: 192.168.20.2
 - PC0: 192.168.1.2
 - PC1: 192.168.1.3
 - Subnet Mask for all: 255.255.255.0
 - Default Gateway: 192.168.20.1
6. Configure the Mail Server:
- Click on the Server → Go to Services tab → Click on Email.
 - Turn ON the SMTP and POP3 services.
 - Enter the Domain name and click set.
 - Add email users:
 - User: ram, Password: ram
 - User2: sita, Password: sita

The screenshot shows the Mail Server configuration interface. The 'Services' tab is selected, and the 'EMAIL' section is visible. The 'SMTP Service' and 'POP3 Service' are both turned ON. The 'Domain Name' is set to 'suyash.com'. The 'User Setup' section shows two users: 'ram' and 'sita'.

7. Configure Email Client on PC0:

- a. Go to Desktop → Email
- b. Set the following details:

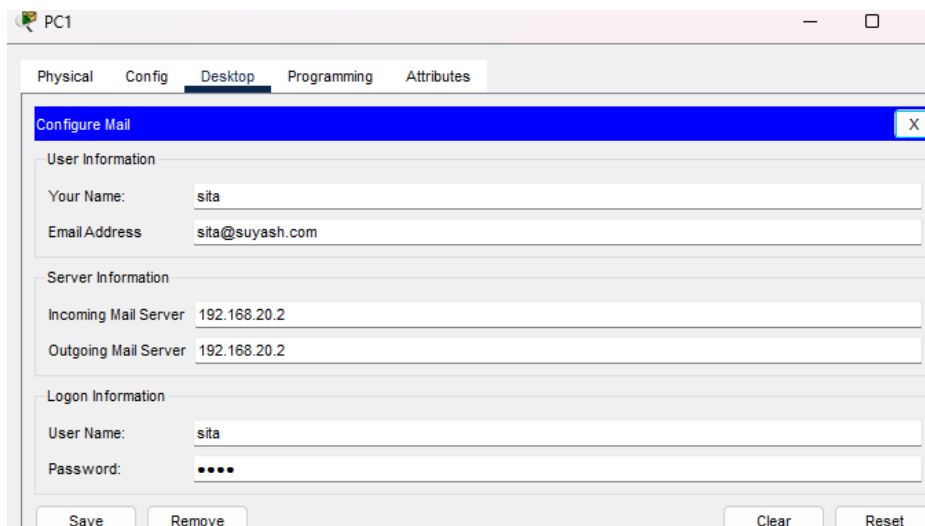


The screenshot shows the 'PC0' window with the 'Desktop' tab selected. A 'Configure Mail' dialog box is open, containing the following fields:

- User Information:**
 - Your Name: ram
 - Email Address: ram@suyash.com
- Server Information:**
 - Incoming Mail Server: 192.168.20.2
 - Outgoing Mail Server: 192.168.20.2
- Logon Information:**
 - User Name: ram
 - Password: (masked with dots)

At the bottom of the dialog are buttons for 'Save', 'Remove', 'Clear', and 'Reset'.

8. Configure Email Client on PC1 (for User2) similarly.



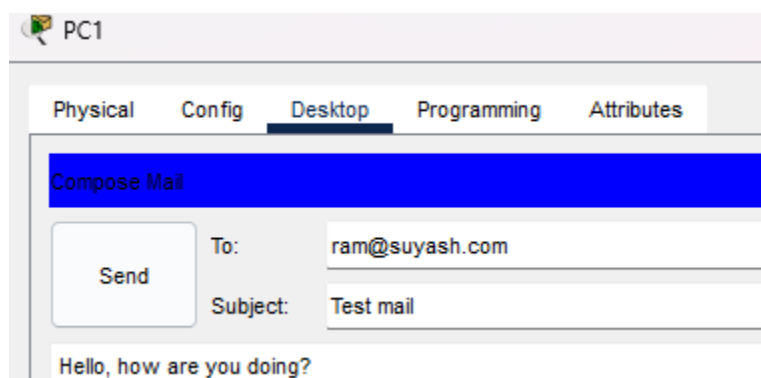
The screenshot shows the 'PC1' window with the 'Desktop' tab selected. A 'Configure Mail' dialog box is open, containing the following fields:

- User Information:**
 - Your Name: sita
 - Email Address: sita@suyash.com
- Server Information:**
 - Incoming Mail Server: 192.168.20.2
 - Outgoing Mail Server: 192.168.20.2
- Logon Information:**
 - User Name: sita
 - Password: (masked with dots)

At the bottom of the dialog are buttons for 'Save', 'Remove', 'Clear', and 'Reset'.

9. Send an Email from PC1 to PC0:

- a. Go to PC1's Email app
- b. Compose a new email:



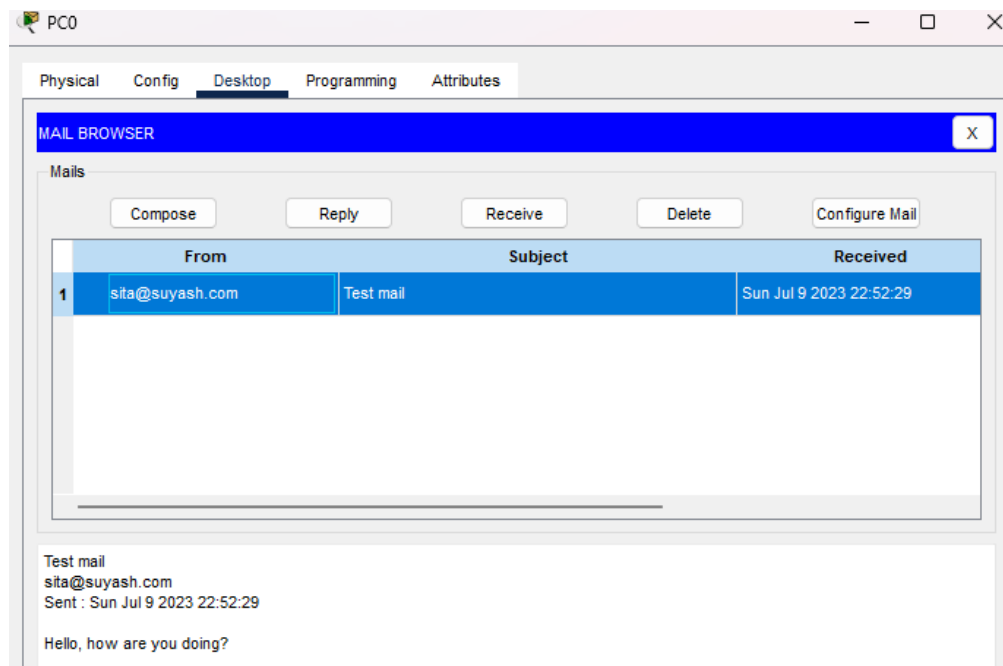
The screenshot shows the 'PC1' window with the 'Desktop' tab selected. A 'Compose Mail' dialog box is open, containing the following fields:

- To:** ram@suyash.com
- Subject:** Test mail
- Body:** Hello, how are you doing?

There is a 'Send' button on the left side of the dialog.

10. Receive Email on PC0:

- a. Open Email app → Click Receive
- b. Check if the message appears



4. Outcome:

Successfully simulated a mail server environment in Cisco Packet Tracer. SMTP and POP3 services were configured and email communication was established between two client PCs. This helped in understanding basic email flow over a network.

5. Conclusion:

This lab demonstrated how SMTP and POP3 protocols function within a network using Cisco Packet Tracer. Although simplified, the simulation effectively illustrated the basic principles of sending and receiving emails in a client-server environment. It provided foundational knowledge applicable to real-world server configurations.

Lab 14

Title: Configure file-sharing protocols

(SAMBA for Windows-Linux, NFS for Linux-Linux sharing)

1. Objectives:

- To implement file sharing using SAMBA for Windows-Linux communication.
- To configure NFS for Linux-Linux file sharing.
- To manage permissions and secure access to shared directories.
- To test the file-sharing configurations between different operating systems.

2. Theory:

File Sharing protocols:

File sharing protocols are network protocols that define how computers can access and share files, with common examples including FTP, HTTP, SFTP and SMB.

SAMBA for Windows-Linux sharing:

SAMBA is an open-source software suite that allows file and print sharing between Linux/Unix and Windows systems. It implements the SMB/CIFS protocol, which allows Linux systems to share directories and files with Windows clients.

NFS for Linux-Linux sharing:

NFS is a protocol that allows the sharing of directories and files between Linux systems. It works by allowing one system (the server) to export directories that other systems (clients) can mount and access as if they were local directories.

3. Steps:

A. Configure SAMBA for Windows-Linux File Sharing

1. Install and Configure SAMBA for Windows-Linux Sharing

- Install SAMBA and configure:
 - i. `sudo yum install samba -y`
 - ii. `sudo nano /etc/samba/smb.conf`

Modify the following configuration:

```
[prime]
  path = /home/prime
  writable = yes
  read only = no
  valid users = prime
```

2. Add a User for SAMBA

- Create a new user (prime):
 - i. `sudo adduser prime`
- Set a password for the user:
 - i. `sudo smbpasswd -a prime`

3. Configure the Firewall

- Allow SAMBA service through the firewall:
 - i. `sudo firewall-cmd --permanent --add-service=samba`
 - ii. `sudo firewall-cmd --reload`

4. Start and Enable SAMBA Service

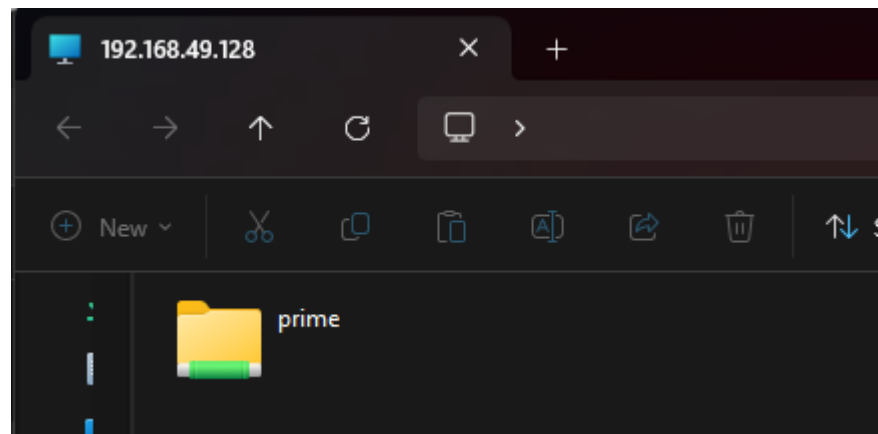
- Restart and enable SAMBA to start at boot:
 - i. `sudo systemctl restart smb`
 - ii. `sudo systemctl enable smb`
 - iii. `sudo systemctl status smb`

5. Test SAMBA Configuration

- i. `testparm`

6. Access the SAMBA Share from Windows

- On a Windows machine, open the File Explorer and type the following:
 - i. `\\192.168.49.128`



B. Configure NFS for Linux-Linux File Sharing

1. Install and Configure NFS Server

- Install NFS server packages:
 - i. `sudo yum install nfs-utils -y`
- Start and enable the NFS server:
 - i. `sudo systemctl start nfs-server`
 - ii. `sudo systemctl enable nfs-server`

- Create a directory to share:
 - i. `sudo mkdir -p /srv/nfs/prime`
 - ii. `sudo chown -R nobody:nobody /srv/nfs/prime`
 - iii. `sudo chmod 755 /srv/nfs/prime`
 - Edit the NFS export file to define shared directories:
 - i. `sudo nano /etc/exports`
 - Add the following line to export the directory:
`/srv/nfs/prime 192.168.49.0/24(rw,sync,no_root_squash,no_subtree_check)`
Replace 192.168.49.0/24 with the appropriate network range.
 - Export the shared directory:
 - i. `sudo exportfs -r`
2. Configure the Firewall on the NFS Server
- Allow NFS service through the firewall:
 - i. `sudo firewall-cmd --permanent --add-service=nfs`
 - ii. `sudo firewall-cmd --reload`
3. Configure the NFS Client (Another Linux Machine)
- Install the NFS client package:
 - i. `sudo yum install nfs-utils -y`
 - Create a mount point:
 - i. `sudo mkdir -p /mnt/nfs/prime`
 - Mount the NFS shared directory:
 - i. `sudo mount 192.168.49.128:/srv/nfs/prime /mnt/nfs/prime`Replace 192.168.49.128 with the IP address of the NFS server.
 - Verify the mount:
 - i. `df -h | grep nfs`
4. Make the Mount Persistent (Optional)
- Edit the `/etc/fstab` file:
 - i. `sudo nano /etc/fstab`
 - Add the following entry:
 - i. `192.168.49.128:/srv/nfs/prime /mnt/nfs/prime nfs defaults 0 0`Now the NFS share will be automatically mounted at boot.

4. Outcome:

To gain hands-on experience in configuring file-sharing protocols (SAMBA and NFS) for file access between Linux and Windows systems, and between Linux systems. This lab will help enhance understanding of networked file systems and secure access control for shared resources.

5. Conclusion:

By following these steps, we successfully set up SAMBA for sharing files between Linux and Windows, and NFS for Linux-Linux sharing. This configuration ensures smooth file sharing while maintaining proper access control and security policies. Proper management of file shares and permissions enhances the efficiency of networked systems.

Lab 15

Title: Configure and Setup CUPS for network printing

1. Objectives:

- To understand the CUPS and its role in network printing.
- To learn how to install and configure CUPS on CentOS.
- To set up printer sharing and allow network clients to access printers.
- To manage print jobs and troubleshoot printing issues.

2. Theory:

CUPS:

CUPS (Common UNIX Printing System) is an open-source printing system used in Linux and UNIX-based systems. It allows a computer to act as a print server, managing print jobs for local and network printers.

How CUPS Works:

1. **Print Server:** CUPS runs as a background service to manage printers and print jobs.
2. **Print Queue:** When a user sends a print job, it gets stored in queue before processing.
3. **Driver Support:** CUPS supports various printer drivers for compatibility with different printers.
4. **Network Printing:** CUPS allows remote devices to send print jobs over the network.
5. **Web-Based Management:** It provides a web-based interface for managing printers.

3. Steps:

1. Install CUPS

- Update the system and install CUPS:
 - i. `sudo yum update -y`
 - ii. `sudo yum install cups -y`

2. Start and Enable CUPS Service

- Enable and start the CUPS service so it runs at startup:
 - i. `sudo systemctl start cups`
 - ii. `sudo systemctl enable cups`

3. Allow CUPS Through the Firewall

- If the firewall is active, allow printing services:
 - i. `sudo firewall-cmd --permanent --add-service=ipp`
 - ii. `sudo firewall-cmd --reload`
- ipp (Internet Printing Protocol) is the primary protocol used by CUPS.

4. Configure CUPS for Remote Access

- Edit the CUPS configuration file:
 - i. `sudo nano /etc/cups/cupsd.conf`
- Modify or add the following lines to allow remote access:

```
Listen 192.168.49.168:631 # Replace with your server's IP
Listen localhost:631
Browsing On
BrowseOrder allow,deny
BrowseAllow all
<Location />
    Order allow,deny
    Allow @LOCAL
</Location>
```
- Save and exit (CTRL+X, then Y, then ENTER).
- Restart CUPS to apply changes:
 - i. `sudo systemctl restart cups`

5. Access CUPS Web Interface

- Open a web browser and go to:
 - i. `http://localhost:631`
 - or
 - ii. `http://server-ip:631`
- Here, you can add, manage, and configure printers.

6. Add a Printer:

- i. In the CUPS web interface, click Administration > Add Printer.
- ii. Select Local Printer or Network Printer (depending on your setup).
- iii. Provide a printer name, location, and description.
- iv. Choose the correct printer driver.
- v. Set default printer options and save.

7. Test Printing

- Send a test print job:
 - i. `lp -d Printer_Name /etc/hosts`
- Replace Printer_Name with the name of the printer you added.
- Check the print queue:
 - i. `lpstat -o`
- Cancel a print job if needed:
 - i. `cancel Job_ID`

4. Outcome:

To gain hands-on practical knowledge in installing and configuring CUPS for network printing, enabling remote printing, verifying printer connectivity and managing print jobs efficiently using CUPS.

5. Conclusion:

By setting up CUPS, we have configured a centralized printing system that supports both local and network printing. The web-based management interface makes it easy to add and manage printers, while the firewall and remote access settings ensure secure printing over the network.

Lab 16

Title: Install and Configure FTP server for file transfers

1. Objectives:

- To understand the purpose and functionality of an FTP server.
- To install and configure vsftpd (FTP server) on CentOS.
- To configure users and manage FTP access.
- To test file transfers using FTP clients.

2. Theory:

FTP (File Transfer Protocol) Servers:

FTP is a standard network protocol used to transfer files between a client and a server over a network. The vsftpd (Very Secure FTP Daemon) is one of the most secure and fast FTP servers available for Linux-based systems. FTP servers are widely used for hosting and sharing files in enterprise environments.

FTP (File Transfer Protocol) Servers:

- Supports anonymous and authenticated user access.
- Allows file uploads and downloads.
- Can be secured using firewall rules and access restrictions.

3. Steps:

1. Install and Configure FTP Server (vsftpd)

- Update system packages:
 - i. `sudo dnf update -y`
- Install vsftpd (FTP Server):
 - i. `sudo dnf install vsftpd -y`
- Edit the vsftpd configuration file:
 - i. `sudo nano /etc/vsftpd/vsftpd.conf`
- Update/add the following lines:
 - i. `anonymous_enable=NO`
 - `local_enable=YES`
 - `write_enable=YES`
- Restart vsftpd to apply changes:
 - i. `sudo systemctl restart vsftpd`

2. Configure Firewall (if enabled)

- Allow FTP traffic through the firewall:
 - i. `sudo firewall-cmd --permanent --add-service=ftp`
 - ii. `sudo firewall-cmd --reload`

3. Create a New FTP User

- Add a new user for FTP access:
 - i. `sudo adduser suyashftp`
- Set a password for the user:
 - i. `sudo passwd suyashftp`
- Restart vsftpd to apply user access settings:
 - i. `sudo systemctl restart vsftpd`

4. Test FTP File Transfers

- Create a test file:
 - i. `echo "Hi, I am suyash shrestha" > suyash.txt`
- Install FTP client:
 - i. `sudo dnf install ftp -y`
- Connect to the FTP server:
 - i. `ftp 192.168.49.128`
- Upload the test file to the FTP server:
 - i. `put suyash.txt`
- Download the test file from the FTP server:
 - i. `get suyash.txt`
- Exit the FTP session:
 - i. `quit`

4. Outcome:

To gain hands-on experience in setting up and configuring an FTP server, managing users and performing file transfers using FTP clients.

5. Conclusion:

By following these steps, we will be able to successfully install and configure an FTP server, allowing secure file transfers between clients and servers. Proper configuration of users, firewall settings and testing ensure an efficient and secure FTP environment.