Network and System Administration

New Notes

Provided By Dilbar Yadav

Website: www.arjun00.com.np

Unit 1

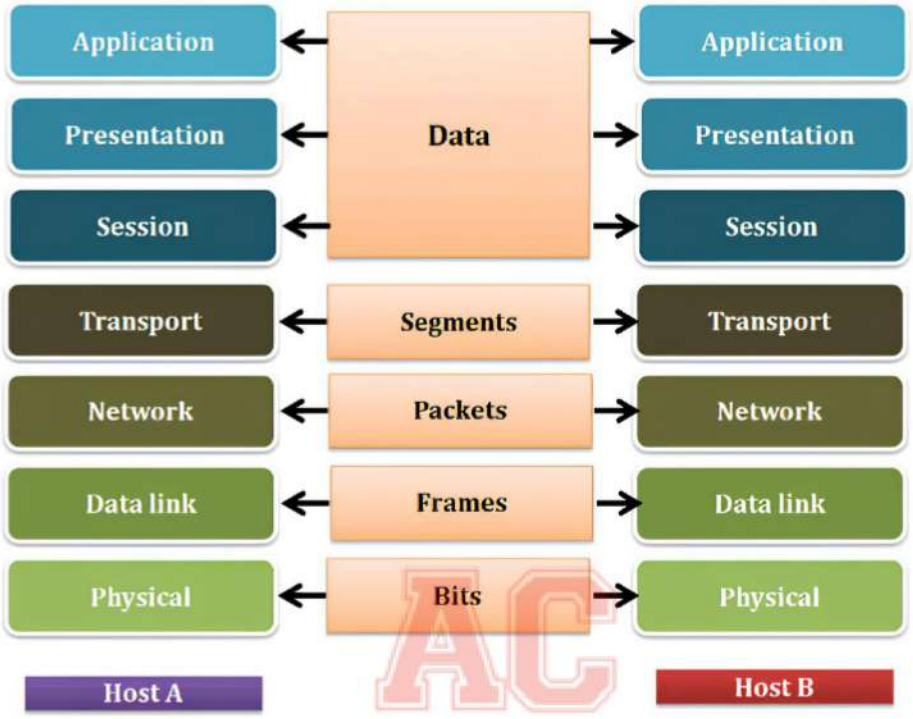
Network protocol:

In computer networks, protocols are a set of rules and procedures that govern the communication between devices. These protocols define how data is formatted, transmitted, received, and interpreted. They ensure that devices can communicate and understand each other in a networked environment. Here are some important protocols used in computer networks:

1. **Transmission Control Protocol/Internet Protocol (TCP/IP):** TCP/IP is the fundamental protocol suite used for communication on the internet. TCP provides reliable, connection-oriented communication between devices, while IP handles the addressing and routing of data packets across networks.
2. **Hypertext Transfer Protocol (HTTP):** HTTP is the protocol used for transferring hypertext documents on the World Wide Web. It defines how web browsers and web servers communicate and exchange information.
3. **File Transfer Protocol (FTP):** FTP is a protocol used for transferring files between computers on a network. It allows users to upload and download files from remote servers.
4. **Simple Mail Transfer Protocol (SMTP):** SMTP is an email protocol used for sending and receiving email messages. It defines how email clients and mail servers communicate to transmit messages.
5. **Post Office Protocol (POP) and Internet Message Access Protocol (IMAP):** POP and IMAP are protocols used for retrieving email from a mail server. They allow email clients to access and download messages from the server.
6. **Domain Name System (DNS):** DNS is a protocol that translates domain names (e.g. www. googgle.com) into IP addresses. It enables users to access websites using human-readable domain names instead of numeric IP addresses.
7. **Dynamic Host Configuration Protocol (DHCP):** DHCP is a protocol used for dynamically assigning IP addresses to devices on a network. It simplifies the process of network configuration by automatically providing IP addresses and related network settings.
8. **Secure Shell (SSH):** SSH is a cryptographic network protocol that provides secure remote access to devices over an unsecured network. It enables encrypted communication and secure command execution.
9. **Internet Protocol Security (IPsec):** IPsec is a suite of protocols used to secure IP communication by encrypting and authenticating data packets. It ensures the confidentiality, integrity, and authenticity of network traffic.
10. **Simple Network Management Protocol (SNMP):** SNMP is a protocol used for managing and monitoring network devices. It allows administrators to gather information, configure settings, and receive notifications from network devices.

OSI model:

The OSI (Open Systems Interconnection) reference model is a conceptual framework that standardizes the functions of a communication system into seven distinct layers. It was developed by the International Organization for Standardization (ISO) in the late 1970s and early 1980s to facilitate interoperability between different computer systems and network protocols. The OSI model serves as a guide for designing, implementing, and troubleshooting network architectures.

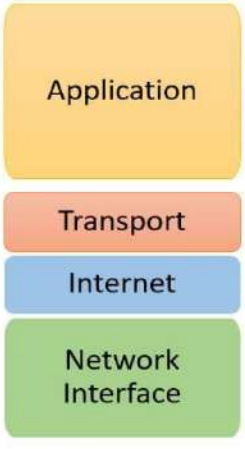


1. **Application Layer:** This layer represents the interface between the network and the user. It provides services such as email, file transfer, and web browsing. Protocols like HTTP, SMTP, FTP, and DNS operate at this layer.
2. **Presentation Layer:** This layer is responsible for data representation and encryption. It ensures that data is formatted, compressed, encrypted, or decrypted according to the needs of the application layer. Examples of protocols that operate at this layer include JPEG, MPEG, SSL, and ASCII.
3. **Session Layer:** The session layer establishes, manages, and terminates communication sessions between applications. It handles functions such as session setup, coordination, and synchronization. Remote Procedure Call (RPC) and Session Control Protocol (SCP) are examples of session layer protocols.
4. **Transport Layer**: This layer ensures reliable and error-free data delivery between end systems. It segments data into smaller units, manages flow control, and provides mechanisms for error recovery and retransmission. TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) operate at this layer.

5. **Network Layer**: The network layer deals with the logical addressing and routing of data across networks. It determines the optimal path for data transmission and handles the addressing and forwarding of packets. IP (Internet Protocol) is the primary protocol of this layer.
6. **Data Link Layer**: This layer provides error-free transmission of data frames between adjacent network nodes. It organizes raw data into frames, performs error detection, and may include flow control mechanisms. Ethernet, Wi-Fi, and PPP (Point-to-Point Protocol) are examples of data link layer protocols.
7. **Physical Layer**: The physical layer defines the physical characteristics of the transmission medium, such as cables, connectors, and signaling. It handles the transmission and reception of raw bit streams over the physical medium.

TCP/IP Model:

- **TCP/IP** was designed and developed by the Department of Defense (DoD) in the 1960s and is based on standard protocols. It stands for Transmission Control Protocol/Internet Protocol. It contains four layers, unlike the seven layers in the OSI model. **TCP/IP Model** helps you to determine how a specific computer should be connected to the internet and how data should be transmitted between them. It helps you to create a virtual network when multiple computer networks are connected together. The purpose of TCP/IP model is to allow communication over large distances.



TCP/IP Conceptual Layers

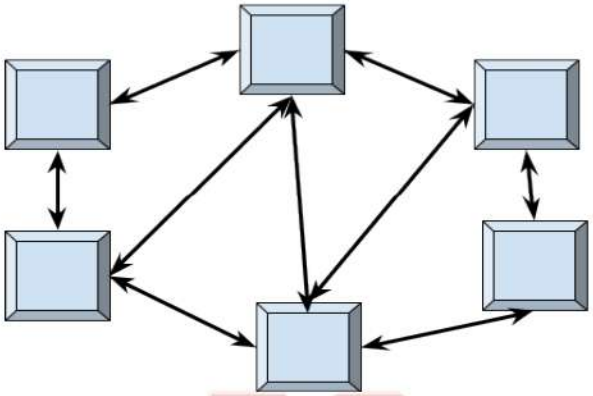
1. **Application Layer:** This is the topmost layer and defines the interface of host programs with the transport layer services. This layer includes all high-level protocols like Telnet, DNS (Domain Name System), HTTP (Hyper Text Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), etc.
2. **Transport Layer:** It is responsible for error-free end-to-end delivery of data. The protocols defined here are Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

3. **Internet Layer:** It defines the protocols for logical transmission of data over the network. The main protocol in this layer is Internet Protocol (IP) and it is supported by the protocols ICMP (Internet Control Massage Protocol), IGMP (Internet Group Management Protocol), RARP (Reverse Address Resolution Protocol), and ARP (Address Resolution Protocol).
4. **Network access layer :** It helps you to defines details of how data should be sent using the network . It also includes how bits should optically be signaled by hardware devices which directly interfaces with a network medium, like coaxial, optical, fiber, or twisted-pair cables.

OSI Model	TCP/IP model
It is developed by ISO (*International Standard Organization*)	It is developed by ARPANET (*Advanced Research Project Agency Network*).
OSI model provides a clear distinction between interfaces, services, and protocols.	TCP/IP doesn't have any clear distinguishing points between services, interfaces, and protocols.
OSI refers to Open Systems Interconnection.	TCP refers to Transmission Control Protocol.
OSI uses the network layer to define routing standards and protocols.	TCP/IP uses only the Internet layer.
OSI model use two separate layers physical and data link to define the functionality of the bottom layers.	TCP/IP uses only one layer (link).
OSI layers have seven layers.	TCP/IP has four layers.
OSI model, the transport layer is only connection-oriented.	A layer of the TCP/IP model is both connection-oriented and connectionless.
In the OSI model, the data link layer and physical are separate layers.	In TCP, physical and data link are both combined as a single host- to-network layer.
Session and presentation layers are a part of the TCP model.	There is no session and presentation layer in TCP model.
It is defined after the advent of the Internet.	It is defined before the advent of the internet.
The minimum size of the OSI header is 5 bytes.	Minimum header size is 20 bytes.

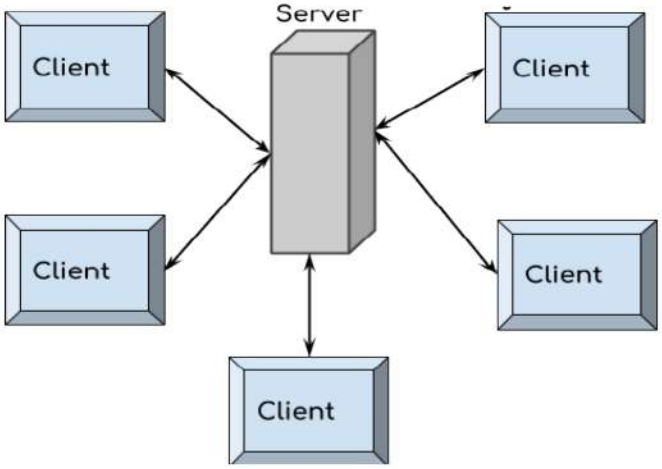
Peer to peer Architecture:

- In peer-to-peer architecture all the computers in a computer network are connected with every computer in the network. Every computer in the network use the same resources as other computers. All computers are considered equal and all have the same abilities to use the resources available on this network. Peer-to-peer (P2P) network architecture is a decentralized model in which computers, referred to as peers, communicate and share resources directly with each other without the need for a central server. In a P2P network, each peer can act as both a client and a server, enabling them to request and provide resources or services.



Client server Architecture:

- In Client Server architecture a central computer acts as a hub and serves all the requests from client computers. All the shared data is stored in the server computer which is shared with the client computer when a request is made by the client computer.
- All the communication takes place through the server computer, for example if a client computer wants to share the data with other client computer, then it has to send the data to server first and then the server will send the data to other client.



FACTORS	CLIENT-SERVER	PEER-TO-PEER
Basic	There is a specific server and specific clients connected to the server.	Clients and server are not distinguished; each node act as client and server.
Service	The client request for service and server respond with the service.	Each node can request for services and can also provide the services.
Focus	Sharing the information.	Connectivity.
Data	The data is stored in a centralized server.	Each peer has its own data.
Server	When several clients request for the services simultaneously, a server can get bottlenecked.	As the services are provided by several servers distributed in the peer-to-peer system, a server in not bottlenecked.
Expense	The client-server are expensive to implement.	Peer-to-peer are less expensive to implement.
Stability	Client-Server is more stable and scalable.	Peer-to-Peer suffers if the number of peers increases in the system.

Unit 2

Linux:

Linux is a powerful free and open-source software that operates on its own operating system. The term 'Linux' stands for GNU + Linux. Initially developed by Linus Torvalds in 1991. It acts as the basis for a variety of devices, such embedded systems, cell phones, servers, and personal computers.

Linux Components:

There are a number of moving parts that come together to ensure a Linux OS functions properly. Below are some of the main components to know:

Kernel: kernel is the lowest level of system software which is in direct contact with the hardware. It is the program that is responsible for allocating the computer's hardware resources and scheduling the user's jobs in a fair manner. It is called the heart of Linux which controls all the major functions of the hardware.

Shell: It is an interface to access Linux system. It hides the complexity of Linux kernel. It takes input from user and executes a program based on the input . After the program has finished it displays output of that program. In other words, shell is an interface which can run commands, programs and shell scripts.

Applications: these are the software developed for performing specific task Linux has variety of application software which can be installed similarly like window mac. These are installed over the Linux kernel or operating system

Init system: The init system is the first process the kernel runs after booting. It allows other processes to run and manages daemons.

Daemons: Daemons are services that run in the background like printing and scheduling.
They start either during the booting process or when a user logs into the desktop.

System Libraries:

- Standard libraries that applications use for functionality.
- **Example: GNU C Library (glibc)**

System Utilities:

Basic commands for managing files, processes, and the system.

Example : ls, cp , mv, pwd etc.

Advantages of Linux

- The main advantage of Linux is it is an open-source operating system. This means the source code is easily available for everyone and you are allowed to contribute, modify and distribute the code to anyone without any permissions.
- In terms of security, Linux is more secure than any other operating system. It does not mean that Linux is 100 percent secure, it has some malware for it but is less vulnerable than any other operating system. So, it does not require any anti-virus software.
- The software updates in Linux are easy and frequent.
- Various Linux distributions are available so that you can use them according to your requirements or according to your taste.
- Linux is freely available to use on the internet.
- It has large community support to help each other over issues.
- It provides high stability. It rarely slows down or freezes and there is no need to reboot it after a short time.
- It has its own software repository which are free to use.
- It maintains the privacy of the user.
- The performance of the Linux system is much higher than other operating systems. It allows a large number of people to work at the same time and it handles them efficiently.

- It is network friendly.
- The flexibility of Linux is high. There is no need to install a complete Linux suite; you are allowed to install only the required components.
- Linux is compatible with a large number of file formats.
- It is fast and easy to install from the web. It can also install it on any hardware even on your old computer system.
- It performs all tasks properly even if it has limited space on the hard disk.

Disadvantages of Linux

- It is not very user-friendly. So, it may be confusing for beginners.
- It has small peripheral hardware drivers as compared to windows.

- **ls:** The ls command is used to list all directories and files in the Linux terminal.

  Syntax of Ls: ls[option][file/directory]

- **pwd:** The 'pwd,' which stands for "print working directory. "It prints the path of the working directory, starting from the root.

  Syntax: pwd[option]

- **mkdir:** It allows you to create one or multiple fresh directories at once as well as set the permissions for the directories in the terminal itself.

  Syntax: mkdir <directory name>

- **cd:** The **'cd' command** allows users to change their current working directory within the file system.

  Syntax: cd[directory]

- **rmdir:** It is used to delete permanently an empty directory. To perform this command the user running this command must be having **sudo** privileges in the parent directory.

  Syntax: Rmdir[option]...[directory]...

- **cp:** The cp command of Linux is equivalent to copy-paste and paste in Windows.

  Syntax: cp source_file destination

- **mv:** The **mv command** is generally used for renaming the files in Linux.

  Syntax: mv[source_file] [destination_file]

- **touch:** it is used to create a empty file.

  syntax: touch filename.ext.

- **cat>filename**:create a new file and write contents and press ctl+D to save content.
- **cat:** display content of file.
- **cat file1 file2>file3**: creates new file and content of both files will be copied to new file.
- **head:** display first 10 line contents of file
- **tail:** display last 10 line contents of file
- **tac:** to display line of content of file in reverse order.
- **More:** similar to cat and here we can display large content by using ENTER or SPACEBAR
- **Id:** Display id of User/ Group.
- **Clear:** clear the content of screen.
- **Vi:** it is a text editor to write program or text.
- **grep:** filter to search given pattern from file content
  Syntax grep pattern filename
- **ping:** check the network connectivity status of network and server
- **history**: Review all the commands which you have used or typed.
- **hostname:** Display the host name
- **hostname –i :** it is used to display host ip.
- **chmod:** it is used to set the permission of user/group to access file
- **wc:** display numbers of line, words and characters available in the file content. unique: Remove duplicate of file content/ it can remove only continuous duplicate. rm: remove file and directory. Dose not matter either empty or not

User and group management:

User database file:

/etc/passwd: user list **/etc/shadow:** password list

Users UID System users: 0-999 Normal Users 1000-60000

User:

In Linux, a user is an individual who interacts with the system. Each user has a unique username and a user ID (UID). User accounts are used to log in, run processes, and access files and directories. Linux systems typically have several user accounts, including the root user, which has superuser privileges and can perform administrative tasks.

Groups

Groups are collections of users. They are used to simplify access control and permissions management. Users within the same group share common permissions to files and directories. A group also has a unique group ID (GID). When a user creates a file, the file's group ownership is set to the user's primary group by default.

Creating Users

To create a new user, use the **useadd** command followed by the username

eg. sudo useradd linuxuser

Setting User Password

Use the **passwd** command to set the password for the user.

e.g: sudo passwd linuxuser

modifying User Attributes

To change user attributes like the username or home directory, use the **usermod** command **Sudo usermod –l new_user old_username Sudo usermod –d**

/new/home/directory username Deleting Users

Deleting Users

To remove a user account, including their home directory and files, use the **userdel** command

eg sudo userdel – r username

User Database files

/etc/passwd : This file contains all user details as a list.

/etc/shadow :This file contains all user password details as a list

Creating Groups:

You can create a group using the **groupadd** command

e.g. Sudo groupadd mygroup Adding Users to Groups:

to add users to a group, use the **usermod** command with the –aG flag:

sudo usermod –aG mygroup linuxuser

Changing Group Ownership of Files:

To change the group ownership of a file or directory, use the chown command

eg: sudo chown :mygroup file_or_directory

Deleting Groups:

To delete a group, use the **groupdel** command.

e.g. Sudo groupdel mygroup

Group Database files

/etc/group : This file contains all group details as a list.

/etc/gshadow : This file contains all group members details as a list.

1: Creating a User and Group

Let's create a user named "chandra" and

a group named "developers":

e.g: sudo useradd chandra sudo groupadd developers

2: Adding User to a Group

Add "chandra" to the "developers"

group:

e.g: sudo usermod -aG developers Chandra

3: Changing File Ownership

Change ownership of a file to the
"developers"
group: **e.g: sudo chown :developers file.txt**

4: Deleting User and Group

Remove "chandra" and the "developers"
group: **e.g: sudo userdel -r chandra sudo groupdel developers**

File System:

The Linux file system is a hierarchical structure that organizes and manages file on a Linux-based **operating system**. It defines how to store, access, and retrieve data on the disk. Files are arranged in a hierarchical directory tree starting with the root directory (/) which branches into subdirectories and files. Unlike Windows, Linux does not use drive letters and makes all devices and partitions part of the unified file system.

Linux supports a variety of file system types, each suitable for different use cases, ranging from general-purpose desktop environments to large-scale enterprise storage solutions. Using the appropriate file system optimizes performance, reliability, and data management.

The table below outlines **key Linux file systems**, their main features, advantages, and common applications:

File System	Use Case	Features	Main Pros/Cons
ext4	Default on most Linux distributions. Suitable for general-purpose systems, including desktops and servers.	Journaling for crash recovery. Support for volumes up to 1 TiB and files up to 16 TiB. Fast mount times and delayed allocation for better performance.	Mature and stable with extensive community support. Lacks advanced features like snapshots.
XFS	Ideal for high-performance environments that handle large files, such as media servers or **databases**.	High throughput with large files and scalable to massive storage. Good support for parallel I/O operations. Online **defragmentation** and resizing.	Poor performance with many small files. Complex to tune and manage effectively.
Btrfs	Used for advanced use cases that require snapshots, subvolumes, or integrated volume management; often found in modern servers and storage appliances.	Snapshotting, **RAID** support, self- healing via **checksums**. Support for **compression** and **deduplication**. Online resizing and defragmentation.	Less mature than ext4. Stability issues in certain use cases.

File System	Use Case	Features	Main Pros/Cons
ZFS	Suitable for environments that require extreme data integrity, such as enterprise servers, **backups**, and **NAS devices**.	Combines file system and volume management. RAID-Z, data deduplication, and end-to-end data integrity verification. Supports pools with immense storage capacities.	Requires substantial **memory** for optimal performance (8 GB+). Licensing restrictions (CDDL vs. **GPL**).
exFAT & NTFS (via ntfs-3g)	Typical choice for dual-boot setups or external drives shared between Linux and Windows.	Read/write support for NTFS and exFAT partitions on Linux. exFAT is optimized for flash drives and SD cards. NTFS supports advanced Windows-specific features.	ntfs-3g performance can be slower compared to native NTFS on Windows.exFAT does not support journaling, which limits data recovery options.

Process:

A process is an instance of a program currently running on a computer system. In Linux, processes are managed by the operating system's kernel, which allocates system resources and schedules processes to run on the CPU. In Linux, ***processes can be categorized into two types:***

Foreground Processes :

Foreground processes are the kinds of processes that require input from the user and are characterized by their interactivity. For instance, a foreground process would be like you are running an Office application on the Linux system.

Background Processes:

Background processes are non-interactive operations carried out in the background and do not call for any participation from the user. Antivirus software is an example of a Background Process.

Additionally, processes can be system processes or user processes. ***System processes*** are initiated by the kernel, while users initiate ***User processes.***

Process Management:

Process management is the task of controlling and monitoring the processes that are running on a Linux system. It involves managing process resources, scheduling processes to run on the CPU, and terminating processes when required.

In Linux, a process can be in one of five states:

1. **Running:**
   The process is currently executing on the CPU.
2. **Sleeping:**
   The process is waiting for a resource to become available.
3. **Stopped:**
   The process has been terminated by a user
4. **Zombie:**
   The process has completed execution but has not yet been cleaned by the system.
5. **Orphan:**
   The parent process of the current process has been terminated.

Commands	Description
ps	Displays information about the processes running currently.
top	Provides real-time information about system processes and their resource usage.
kill	Terminates a process by sending a signal to it.
nice	Adjusts the priority of a process.
renice	Changes the priority of a running process.
ps PID	Shows the state of an exact process.
pidof	Shows the Process ID of a process.
df	Shows Disk Management of your system.
free	Shows the status of your RAM.
bg	For sending a running process to the background.
fg	For running a stopped process in the foreground.

Jobs: will show active jobs

Bg: Resume jobs to background

Fg: Resume job to the foreground

- Sudo apt update
- Sudo apt install mysql-server

To check the status of mysql server

- Sudo systemctl staus mysql To enabled/disable

  Sudo systemctl enable/disable mysql

To start/stop service

Sudo systemctl start /stop mysql Set security related configuration

Sudo mysql-secure-installation

Sudo mysql

To set password

Mysql> ALTER USER
'root'@'localhost' IDENTIFIED WITH mysql_native_password BY "your password"

Exit

- Sudo mysql –u username –p "yourpassword"

Listout databasese.

- mysql> show databases;

To choose databse.

- mysql>use mysql;

To listout tables of selected database.

- Mysql> show tables;

To create database:

mysql> create database nmss;

mysql> use nmss;

To create table mysql> use nmss;

```
mysql>CREATE table student (
Id INT AUTO_INCREMENT PRIMARY KEY,
Name VARCHAR(25), Email VARCHAR(50),
Created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP);
```

Unit 3

Mail server:

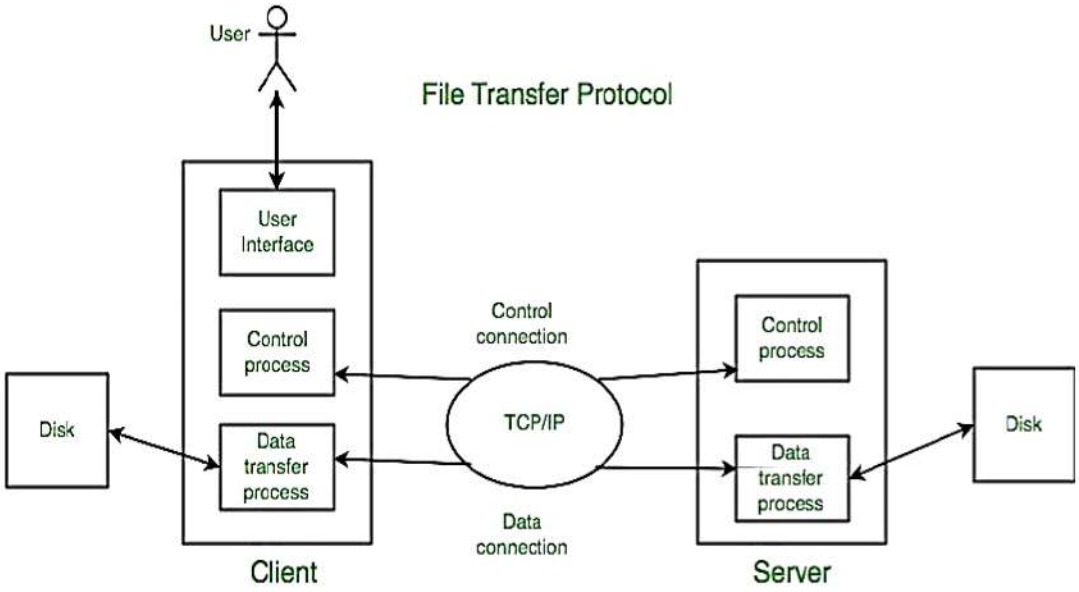
- A mail server is a computerized system that is dedicated to sending, receiving, storing, and forwarding electronic mail(email) message over a network. It uses various network protocols to handle the exchange of emails between users or between email clients and servers. The primary function of a mail server is to facilitate the efficient and reliable transfer of email messages.

- ***The key components and functionalities associated with a mail server:***

1. **Mail transfer Agent(MTA):** The mail Transfer Agent is responsible for routing and transferring emails between different mail servers. It uses standard email protocol such as Simple Mail Transfer Protocol (SMTP) for sending emails and post office protocol version 3 (POP3) or internet Message Access Protocol (IMAP) for receiving emails.
2. **Mail Delivery Agent(MDA): T**he mail Delivery Agent is responsible for delivering emails to the recipient's mailbox. It works in conjunction with the MTA to ensure that emails are placed in the appropriate mailbox on the recipient's server.
3. **Mail User Agent(MUA): T**he mail User Agent is the email client used by end- users to compose, send, receive, and manage emails. Popular MUAs include Microsoft Outlook, Mozilla thunderbird, Apple mail , and web-based clients like Gmail or Yahoo Mail.
4. **Mailbox:** The mailbox is a storage area on the mail server where a user's received emails are stored until they are retrieved by the user's email client. The messages remain on the server until the user decides to download or access them.
5. **Spam filtering and Security:** mail servers often incorporate spam filters and security features to protect users from unwanted emails and potential threats , such as malware or phishing attempts.
6. **Authentication and Encryption:** Authentication mechanisms ensure that only authorized users can send or access emails on the server. Encryption , such as Transport Layer Security(TLS) for SMTP, is used to secure the communication between mail servers and clients.

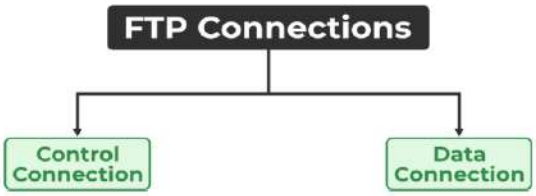
File Transfer Protocol:

- FTP is a standard network protocol for transferring data between a client and a server via a computer network, most often the internet. It was created in the early 1970s and has undergone various revisions since then. FTP allows users to upload and receive files from a remote server, as well as execute many file and directory operations.



- The above figure shows the basic model of the FTP. The FTP client has three components the user interface, control process, and data transfer process. The server has two components: the server control process and the server data transfer process.

There are two types of connections in FTP:



a) **Control Connection:** The control connection uses very simple rules for communication. Through control connection, we can transfer a line of command or line of response at a time. The control connection is made between the control processes. The control connection remains connected during the entire interactive FTP session.

b) **Data Connection:** The Data Connection uses very complex rules as data types may vary. The data connection is made between data transfer processes. The data connection opens when a command comes for transferring the files and closes when the file is transferred.

Advantages of FTP:

- **Speed:** One of the biggest advantages of FTP is speed. The FTP is one of the fastest ways to transfer the files from one computer to another computer.
- **Efficient:** It is more efficient as we do not need to complete all the operations to get the entire file.
- **Security:** To access the FTP server, we need to login with the username and password. Therefore, we can say that FTP is more secure.
- **Back & forth movement:** FTP allows us to transfer the files back and forth. Suppose you are a manager of the company, you send some information to all the employees, and they all send information back on the same server.

Disadvantages of FTP:

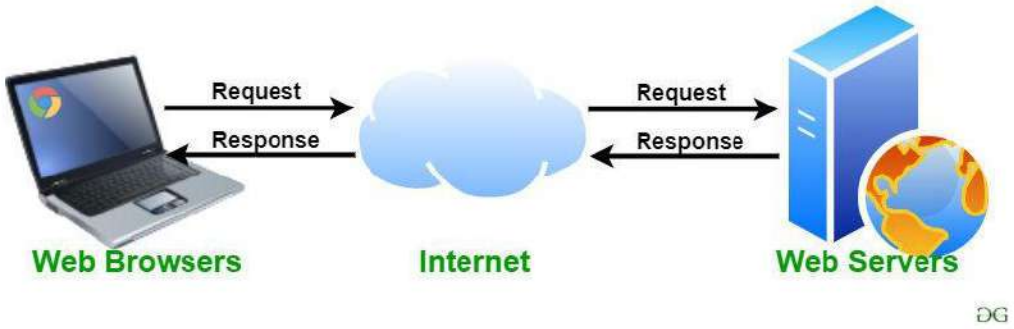
- FTP Lacks Security
- Not All Vendors Are Created Equal
- Encryption isn't a Given
- FTP can be Vulnerable to Attack
- Compliance is an Issue
- It's Difficult to Monitor Activity

Web Server:

- A web server is software and hardware that uses HTTP and other protocol to respond to client requests made over the World wide web. The main job of a web server is to display website content through storing, processing and delivering web pages to users. Besides HTTP, web servers also support SMTP used for email & FTP used for file transfer and storage.
- Web server hardware is connected to the internet and allows data to be exchanged with other connected devices, while web server software controls how a user accesses hosted files. The web server process is an example of the client/ server model. All computers that host website must have web server software.
- Webserver are used in web hosting, or the hosting of data for websites and web-based applications or web applications.

How Does a Web Server Work?

- When a user accesses a website by entering a URL in their web browser, the browser sends an HTTP request to the web server hosting the website. The web server processes this request and returns the necessary resources to display the page on the user's browser.

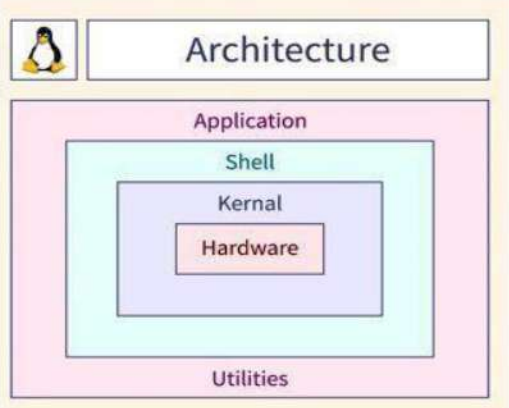


1. **Client Request:** In the web browser(https://www.google.com) the user enters a URL.
2. **DNS Resolution:** To get the IP address of the requested domain, the browser contacts a Domain Name System (DNS) server.
3. **Connecting to the Web Server:** Using the obtained IP address the browser establishes
   a connection with the web server.
4. **Processing Request:** The web server receives the request and processes it.
5. **Serving the Response:** The requested files (HTML, CSS, JavaScript, images) are sent back to the client's browser by the web server.
6. **Rendering the Web Page:** Based on the received data the browser displays the web page to the user.

Unit 4

Shell:

- The shell can be defined as a command interpreter within an operating system like Linux/GNU or Unix. It is a program that runs other programs. The shell facilitates every user of the computer as an interface to the Unix/GNU Linux system. Hence, the user can execute different tools/utilities or commands with a few input data .The shell sends the result to the user over the screen when it has completed running a program which is the common output device. That's why it is known as "**command interpreter**". The shell is not just a command interpreter. Also, the shell is a programming language with complete constructs of a programming language such as **functions, variables, loops, conditional execution**, and many others.



1. Shell is responsible to read command provided by user
2. Shell will check whether the command is valid or not.
3. Shell will check whether the command is properly used or not
4. If every thing is proper then shell interprets (convert) that command into command understandable form and handover that converted command to kernel.
5. Shell acts as interface between user and kernel .shell +kernel is nothing but operating system.
6. Kernel is responsible to execute that command with the help of hardware .

Bourne shell:

It is developed by Stephen Bourne. It is a first shell which is developed for **UNIX.**

By using **sh** command we can access this shell.

1. Command full-path name is /bin/sh and /sbin/sh,
2. Non-root user default prompt is $,
3. Root user default prompt is #.

Bash shell (Bourne Again Shell)

It is advanced version of Bourne shell. This is default shell for most of Linux Flavors.

By using **bash** command we can access this shell.

Korn Shell:

1. It is developed by David Korn.
2. Mostly this shell used in IBM AIX operating system.
3. By using **ksh** command, we can access this shell.
4. Command full-path name is /bin/ksh .

Tshell :

1. T means Terminal
2. It is advanced version of Cshell
3. It is most commonly used in Hp UNIX systems
4. By using **tcsh** command we can access Tshell.
5. Command full-path name is /bin/tcsh.

Zshell :

1. Z Shell was created by Paul Falstad in 1990 while he was a student at Princeton University. Z Shell is an extended version of the Bourne-Again Shell (bash), with additional features and capabilities.
2. By using **zsh command we can acess Zshell**
3. Command full-path name is /bin/zsh,

Note:

- **Usually owner or root user can change permission of shell**
- **The most commonly used shell in Linux environment is Bash**
- **How to check default shell in our system**
  - **echo $0** or **echo $SHELL**
  - **We can check default shell information inside /etc/passwd file also.**
- **How to check all available shell in our system**
  - **cat /etc/shells**

What is shell Script?

- A sequence of command saved to a file and this file is nothing but shell script. It can contains programming features like control statements, loops, functions, array ,if-else, switch /case etc. **Shell Scripting** is a way of writing scripts of programs that are executed in a terminal or shell. Basically, it is a program or script which is written with the help of variables mentioned in it. It is powerful because it can automate tasks, and in this one can use programming constructs that are available in shell, such as loops, conditionals and functions.

Shell Variable is used in shell scripts for many functionalities like storing data and information, taking input from users, printing values that are stored. They are also used for storing data temporarily and storing output of commands.

Shell Variables are used to store data and information within a shell (terminal), and they are also used for controlling the behavior of program and scripts.
Some of the common uses are:

1. Setting environment variables.
2. Storing configuration data.
3. Storing temporary data.
4. Passing arguments to scripts.

Rules for variable definition

A variable name could contain any alphabet (a-z, A-Z), any digits (0-9), and an underscore (_). However, a variable name must start with an alphabet or underscore. It can never start with a number. Following are some examples of valid and invalid variable names:-

Valid variable:

ABC, _AV_3, AV232

Invalid variable:

2_AN, !ABD, $ABC, &QAID

Accessing variable

Variable data could be accessed by appending the variable name with '$' as follows:

```
#!/bin/bash
VAR_1="Devil" VAR_2="OWL"
echo "$VAR_1$VAR_2"
```

Unsetting Variables

The unset command directs a shell to delete a variable and its stored data from list of variables. It can be used as follows:

```
#!/bin/bash var1="Devil" var2=23
echo $var1 $var2 unset var1
 echo $var1  $var2
```

Note: The unset command could not be used to unset read-only variables.

By using sha-bang, we can specify the interpreter(command) which is responsible to execute the script.

Variable Types:

1. **Local Variables:**

   Variables declared inside a function or a script block using local keyword are local to that scope and are not accessible outside.

   function
   my_function() { local
   localVar="Hello"
   echo $localVar
   }

2. **Global Variables:**

   Variables declared outside any function or block are global and can be accessed throughout the script

   globalVar="Wo
   rld“ echo
   $globalVar

3. **Environment Variable:**

   These variables are commonly used to configure the behavior script and programs that are run
   by shell. Environment variables are only created once, after which they can be used by any user.

For example:

`export PATH=/usr/local/bin:$PATH` would add `/usr/local/bin` to the beginning of the shell's search path for executable programs.

4. **Shell Variables:/System Variables:**

   These are predefined variables by the shell or the system, and their values are set by the shell or the operating system.

For example:

$HOME: Home directory of the user.
$USER: Current username.
$PWD: Present working directory.
$SHELL : Stores the path to the shell program that is being used.

5. **Special Variables:**

   These are variables with special meanings in the shell.

   Examples:

$?: Exit status of the last executed command.
$$: Process ID of the current script.
$!: Process ID of the last background command.

6. Array Variables:

Arrays are used to store multiple values under a single variable name.

```
fruits=("apple" "banana" "orange")
echo ${fruits[0]}
var1=23
echo $var1 $var2
```

7. Read only Variables.

These variables are read only i.e. their values could not be modified later in the script.

Following is an example:

```
#!/bin/bash
var1="Devil"
var2=23
read only var1
echo $var1 $var2
var1=23
echo $var1 $var2
```

Calculate area of rectangle

```
#!/bin/bash
echo "Enter the length of the rectangle"
read length
echo "Enter the width of the rectangle"
read width
area=$((length * width))
echo "The are of the rectangle is:
$area"
```

Working with files and directories :

One of the most important features of the Unix file system is its support for symbolic links, which are pointers to other files or directories. This allows for flexible organization of files and directories without having to physically move them around.

1. **/:** The slash / character alone denotes the root of the file system tree.
2. **/bin:** Stands for "binaries" and contains certain fundamental utilities, such as ls or cp, which are generally needed by all users.
3. **/boot :** Contains all the files that are required for successful booting process.
4. **/dev :**Stands for "devices". Contains file representations of peripheral devices and pseudo-devices.

5. **/etc :** Contains system-wide configuration files and system databases. Originally also contained "dangerous maintenance utilities" such as init, but these have typically been moved to /sbin or elsewhere.
6. **/home:** Contains the home directories for the users.
7. **/lib:** Contains system libraries, and some critical files such as kernel modules or device drivers.
8. **/media:** Default mount point for removable devices, such as USB sticks, media players, etc.
9. **/mnt :** Stands for "mount". Contains filesystem mount points. These are used, for example, if the system uses multiple hard disks or hard disk partitions. It is also often used for remote (network) filesystems, CD-ROM/DVD drives, and so on.
10. **/proc:** procfs virtual filesystem showing information about processes as files.
11. **/root:** The home directory for the superuser "root" – that is, the system administrator. This account's home directory is usually on the initial filesystem, and hence not in/home (which may be a mount point for another filesystem) in case specific maintenance needs to be performed, during which other filesystems are not available. Such a case could occur, for example, if a hard disk drive suffers physical failures and cannot be properly mounted.
12. **/tmp:** A place for temporary files. Many systems clear this directory upon startup; it might have tmpfs mounted atop it, in which case its contents do not survive a reboot, or it might be explicitly cleared by a startup script at boot time.
13. **/usr:** Originally the directory holding user home directories,its use has changed. It now holds executables, libraries, and shared resources that are not system critical, like the X Window System, KDE, Perl, etc. However, on some Unix systems, some user accounts may still have a home directory that is a direct subdirectory of /usr, such as the default as in Minix. (on modern systems, these user accounts are often related to server or system use, and not directly used by a person).
14. **/usr/bin:** This directory stores all binary programs distributed with the operating system not residing in /bin, /sbin or (rarely) /etc.
15. **/usr/include:** Stores the development headers used throughout the system. Header files are mostly used by the **#include** directive in C/C++ programming language.
16. **/usr/lib:** Stores the required libraries and data files for programs stored within /usr or elsewhere.
17. **/var:** A short for "variable." A place for files that may change often – especially in size, for example e-mail sent to users on the system, or process-ID lock files.
18. **/var/log:** Contains system log files.

19. **/var/mail:** The place where all the incoming mails are stored. Users (other than root) can access their own mail only. Often, this directory is a symbolic link to /var/spool/mail.
20. **/var/spool:** Spool directory. Contains print jobs, mail spools and other queued tasks.
21. **/var/tmp:** A place for temporary files which should be preserved between system reboots.

Unit 5

Shared Resources:

Shared resources also known as network resources, refer to computer data, information, or hardware devices that can be easily accessed from a remote computer through a local area network (LAN) or enterprise intranet.

Successful shared resources access allows users to operate as if the shared resources were on their own computer. The most frequently used shared network environment objects are files , data , multimedia and hardware resources like printers, fax machines and scanners. File and printer sharing occurs via two network communication mechanisms: Peer-to-peer (p2p) sharing and the client-server network model.

Sharing network resources requires:

- **Security :**Organization present ongoing opportunities for unauthorized shared resources. Security mechanism should be implemented to provide efficient parameters.
- **Compatibility:** Various client –server operating systems may be installed, but the client must have a compatible OS or application to access shared resources. Otherwise, the client may encounter issues that create communication delays and requires troubleshooting.
- **Mapping :** Any shared OS hardware drive , file or resource may be accessed via mapping ,which requires a shred destination address and naming convention.
- **File transfer protocol(FTP) and file sharing:** FTP is not affected by shared resources because the internet is FTP's backbone. File sharing is a LAN concept.

Network File System(NFS):

- ***Network File System (NFS)*** is a type of file system mechanism that enables the storage and retrieval of data from multiple disks and directories across a shared network. A network file system enables local users to access remote data and files in the same way they are accessed locally . NFS was initially developed by sun Microsystems.
- NFS is derived from the distributed files system mechanism .it is generally implemented in computing environments where the centralized management of data and resources is critical . Network file system works on all IP-based networks. It uses TCP and UDP for data access and delivery, depending on the version in use.
- Network File System is implemented in a client/server computing model where an NFS server manages the authentication, authorization and management of clients, as well as all the data shared within a specific file system . Once authorized, clients can view and access the data through their local systems much like they'd access it from an internal disk drive.

Advantages of NFS:

- **Easy to Set Up:** NFS is relatively easy to setup and manage, making it accessible for users with varying levels of technical expertise.
- **High Performance :** NFS is designed for high throughput and low latency , making it suitable for environments where performance is priority.
- **Cross-platform:** While most commonly used on unix and linux systems, NFS clients exist for other operating Systems, including Windows.
- **Scalability:** NFS can easily scale to accommodate growing storage and user demands.
- **File Locking:** NFS supports file locking , which is essential for multi-user collaboration.

Disadvantages Of NFS:

- **Security Concerns:** Earlier version of NFS had limited security features , making them susceptible to unauthorized access. While newer versions have improved security , it remains a concern.
- **Network Dependency :** Being a network file system, the performance and availability of NFS are highly dependent on the network's reliability.
- **No Native Encryption:** NFS does not provide native encryption for data in transit, although this can be mitigated using external solutions.
- **Complexity in Large Environments:** while NFS is easy to setup for small networks, it can become complex to manage in larger, more heterogeneous environments

Samba:

- **Samba** is an ***open-source*** software package that allows file and print services between Linux and windows machines. It is an open-source implementation of the SMB/CIFS protocol.
- Samba is a widely adopted solution for enabling file and print services in heterogenous network environments. It provides a bridge between windows and Unix-like systems, facilitating collaboration and resources sharing across platforms.
- Samba was originally developed in 1991 for and fast and secure file and print share for all clients using the SMB protocol. Since then it has evolved and added more capabilities.
- Samba server in Linux is specially designed to facilitate the communication between the operating systems and several resources. It takes the usage or advantage of the most important protocol called server message block in order to ensure the communication between various systems.

The main uses of samba:

- **File Sharing:** samba allows Linux/Unix systems to share files and directories with windows clients using the SMB/CIFS protocol.
- **Print Sharing:** samba enables Linux/Unix systems to share printers with windows clients, allowing centralized printer management.
- **Domain Controller:** samba can act as a domain controller, enabling Linux/Unix systems to join and participate in windows domains.
- **Integration with windows services:** samba integrates with windows services like WINS and supports Windows domain authentication.
- **Interoperability:** samba promotes seamless collaboration and data exchange between Windows and Linux/Unix systems.
- **Security:** Samba provides user authentication, access control, encryption , and signing mechanism for secure sharing

Print services:

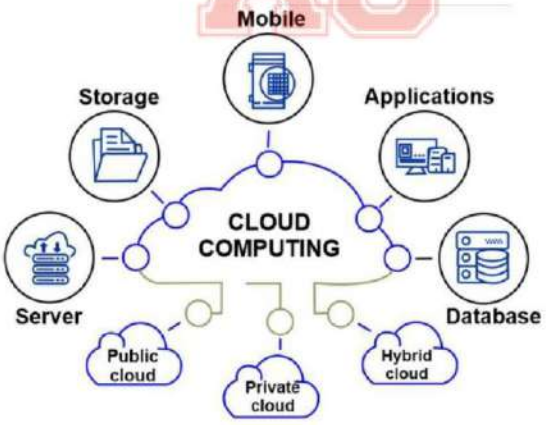
- ***Printer sharing*** is the process of allowing multiple computers and devices connected to the same network to access one or more printers. Each node or device on the network can print to any shared printer and . To some extent, make changes to the printer setting , depending on the permission set by administrator for each user.

- If a printer is attached to a computer that supports printer sharing the computer can share that printer with other computers on the same network. It does not matter whether the shared printer is old or new, as long as it is properly installed in one computer it can be shared by that computer.

- The sharing is facilitated by the OS, which handles the communication between computers and devices within the network and the printer itself. When a print request is sent from a networked computer, this is received by the computer where the shared printer is attached , this host computer initializes the printer then sends the print job to it . Printers can also be shared through a LAN cable if the printer supports the LAN cable port without connecting printer to any of host computers.

Cloud Computing:

- Cloud computing is an emerging model of Business Computing. It distributes computing tasks in a resource pool which consists of many computers, so that various applications can access the cloud as they need. For example computing ability, storage space and a variety of software services. Cloud computing is the product of grid computing, distributed computing, parallel computing, utility computing, network storage and load balancing traditional product development of computer technology and network technology.



- Cloud computing uses the internet as a bridge for digital services. It allows computers to send and receive data to run processes without relying on the local computer. Cloud computing offers faster innovation , flexible resources, and economies of scale. It's also easier to maintain cloud computing applications because they don't need to be installed on each user's computer

Software-as-a-Service (SaaS):

- Software-as-a-Service (SaaS) is a way of delivering services and applications over the Internet. Instead of installing and maintaining software, we simply access it via the Internet, SaaS provides a complete software solution that you purchase on a **pay-as-you-go** basis from a cloud service provider. Most SaaS applications can be run directly from a web browser without any downloads or installations required. The SaaS applications are sometimes called **Web-based software, on-demand software, or hosted software.**

Platform as a Service:

- PaaS is a category of cloud computing that provides a platform and environment to allow developers to build applications and services over the internet. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

Infrastructure as a Service:

- Infrastructure as a service (IaaS) is a service model that delivers computer infrastructure on an outsourced basis to support various operations. Typically IaaS is a service where infrastructure is provided as outsourcing to enterprises such as networking equipment, devices, database, and web servers. It is also known as **Hardware as a Service (HaaS).**

Unit 6

Proxy Server:

A proxy server is a type of internet intermediate server that operates as a bridge between a client, like a computer or smartphone, and the destination server. A resource request is made by a client, which the proxy server intercepts, passes to the target server, and then relays back to the client the response from the destination server.

How Does a Proxy Server Work?

All devices connected to the internet have an internet protocol (IP) address. This address is how a device is recognized on the internet, and it plays a role in how proxy servers work. Proxies can have different ways of working, but the following steps are common among all proxy servers:

When a device makes a request to the internet through a proxy, the proxy server reads and interprets the request.

1. That request is then forwarded to the right internet server.
2. The internet server reads the IP of the proxy and sends the requested data to the IP of that proxy.
3. The proxy server receives the data, extracts it, and checks it for possible malware.
4. Once marked safe, the data is forwarded to the requesting device.

Benefits of proxy Server:

As a proxy server filters out malicious data from the internet before it reaches the company's servers, it can act as an additional layer of security. A proxy server alone might not save the company's network from all hacking attempts, but it can add to the security of the system and lower the risk of cyberattacks.

1. **Anonymity :** Since proxies sit between company networks and internet servers, the internet is unable to know the company IP that generated the request. A company's research and development process, part of its intellectual property, is crucial for its success and must be protected. When an additional layer of security is present between the unfiltered internet and the company servers, it protects sensitive company data from being stolen.

2. **Faster Speed:** Caching is another important function performed by proxy servers. More frequently visited sites can be cached by the proxy, thereby eliminating the need for the proxy to send a request to the internet servers whenever a request is made for those pages, More than that, proxy servers also compress traffic and remove ads from websites, thereby making the internet faster than usual.

3. **Control Internet Usage:**

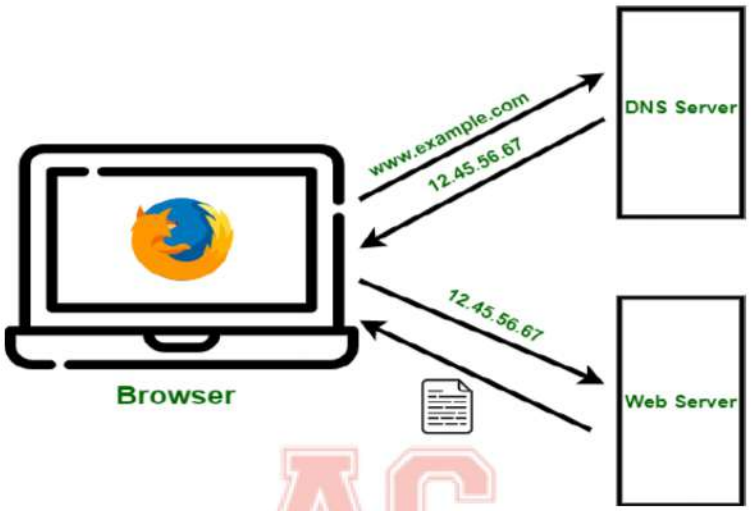
   Proxies can be used to block undesirable content. For example, some companies might want to block certain social media sites so their employees aren't distracted from their work. A proxy server also lets network administrators monitor the requests sent to the internet to ensure no illegal or improper activities are being carried out.

4. **Bypassing Restrictions:**

   Some websites only allow access to IPs from a certain location. This can be a problem when a business needs to access a geo-restricted website, but when a company uses a proxy server, the IP is masked and employees can access the content they need.

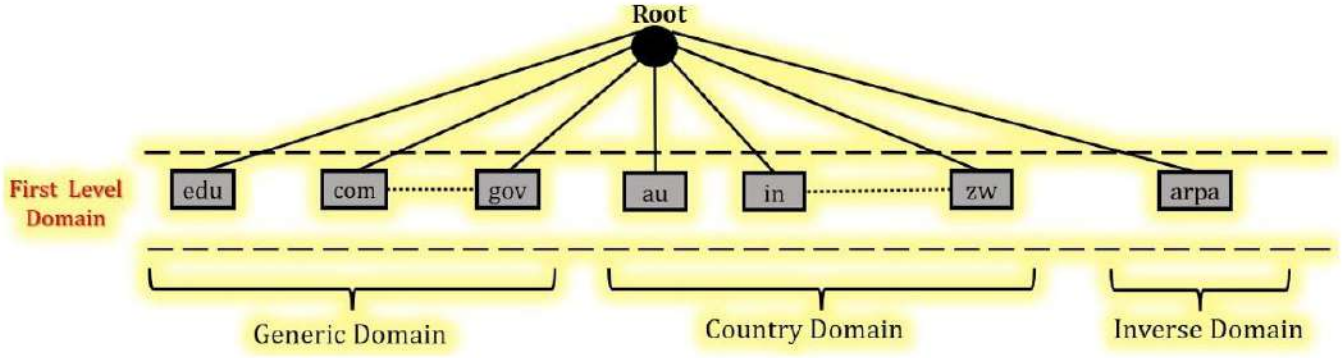
DNS(Domain Name System):

Similar to the phone book on the internet is the *Domain Name System* (DNS). By converting memorable names (like www.dnsexample.com) into the numerical IP addresses (192.100.2.10) that computers use to find one another on the internet, it makes it easier for you to find websites. To access your favorite websites without DNS, you would need to memorize lengthy String of string . A hostname utilized for IP address translation services is the Domain Name System (DNS). A hierarchy of name servers implements DNS, which is a distributed database. It is an application layer protocol that allows clients and servers to exchange messages. It is necessary for the Internet to operate.



Types of Domain

There are various kinds of domains:



1. **Generic Domains:** .com(commercial),.edu(educational), .mil(military), .org(nonprofit organization), .net(similar to commercial) all these are generic domains.
2. **Country Domain:** .np (Nepal) like (arjun00.com.np) .us .uk ,.in(India) etc.
3. **Inverse Domain:** if we want to know what is the domain name of the website. IP to domain name mapping. So DNS can provide both the mapping for example to find the IP addresses of geeksforgeeks.org then we have to type.

Dynamic Host Configuration Protocol (DHCP):

- A network protocol called Dynamic Host Configuration Protocol is used to automate the process of configuring devices (such PCs, printers, and smartphones) on a network by assigning IP addresses and other configuration data. DHCP enables devices to connect to a network and automatically obtain all required network information, such as IP address, subnet mask, default gateway, and DNS server addresses, from a DHCP server, eliminating the need for each device to be individually configured with an IP address.

  It is an application layer protocol which is used to provide:

  1. Subnet Mask (Option 1 – e.g., 255.255.255.0)
  2. Router Address (Option 3 – e.g., 192.168.1.1) **Default Gateway**
  3. DNS Address (Option 6 – e.g., 8.8.8.8)

DORA is the process that is used by DHCP. DORA helps in providing an IP address to hosts or client machines. DORA is the process that follows some steps between the server and client. It gets the IP address from the centralized server. It consists of four-stage:

1. **Discover**
2. **Offer**
3. **Request**
4. **Acknowledge**

Step 1: DHCP Discover Message

This is the first message in the DORA process which helps in finding the DHCP server of the network. DHCP client will find the server by sending DHCP discover message. The broadcast message is sent to the network. As the DHCP client doesn't know the IP address of the server so the message is broadcast with a destination IP is 255.255.255.255. And the source IP will be 0.0.0.0 as the client does not have any IP address. Here the DHCP discover message in the data link layer and network layer is always broadcast.

Source IP address: 0.0.0.0

Destination IP address: 255.255.255.255

Source MAC address: MAC address of DHCP clients

Destination MAC address: FF:FF:FF:FF:FF:FF

Step 2: DHCP Offer Message

DHCP server receives the discover message and it replays the DHCP client with the DHCP offer request. The server sends a DHCP offer message with filled information. It has information about the IP address and duration of time that a host can use. Here destination IP address will be 255.255.255.255 as the DHCP client still does not have its IP address. But this DHCP offer message is broadcast in the network layer and unicast in the data link layer.

Source IP address: IP Address of DHCP Server Destination IP address: 255.255.255.255

Source MAC address: MAC address of DHCP Server

Destination MAC address: MAC address of DHCP clients

Step 3: DHCP Request Message

DHCP clients send the request message to the server when it receives a DHCP offer message from the server. This message tells the server that it accepts the IP address given by the server. Here destination address will be 255.255.255.255 means it's again broadcast. The reason for this is there might be many DHCP servers in the network so the client may receive multiple offer messages and it will accept the request that reaches him first and send a broadcast message to eliminate other DHCP servers. Here source IP address will be 0.0.0.0 as the DHCP server hasn't yet assigned an IP address to the client. DHCP Request Message is also a broadcast message.

Source IP address: 0.0.0.0

Destination IP address: 255.255.255.255

Source MAC address: MAC address of DHCP clients

Destination MAC address: MAC address of DHCP server

Step 4: DHCP Acknowledge Message

This is the last step or message in the DORA process. The DHCP server sends Acknowledge Message to the client when it receives the request message from the DHCP client. This message will contain the IP address and subnet mask that the server assigns to the client. Source IP address will be the IP address of the server. This will be again broadcast message as the destination IP address is 255.255.255.255. But it is unicast in the case of the data link layer.

Source IP address: IP Address of DHCP Server
Destination IP address: 255.255.255.255
Source MAC address: MAC address of DHCP server
Destination MAC address: MAC address of DHCP clients

IPv6:

The goal of IPv6, the next generation of Internet Protocol (IP) address standard, is to complement IPv4, which is still widely used today, and eventually replace it. A computer, smartphone, Internet of Things sensor, home automation component, or any other device connected to the Internet requires a numerical IP address in order to communicate with other devices. The original IP address scheme, known as IPv4, is running out of addresses due to the widespread use of linked devices.

This new IP address version is being deployed to fulfil the need for more Internet addresses. With 128-bit address space, it allows 340 undecillion unique address space.

IPv6 support a theoretical maximum of 340, 282, 366, 920, 938, 463, 463, 374, 607, 431, 768, 211, 456. To keep it straightforward, we will never run out of IP addresses again.

Internet Protocol version 6 (IPv6) is the next version of the IP standard.
While IPv4 and IPv6 will coexist for some time, IPv6 is designed to function in conjunction with IPv4 before fi nally replacing it.

In order to move forward and continue adding new devices and services to the Internet, IPv6 must be implemented.

1. **Internet of Things (IoT):** With a flood of IoT devices ranging from smart home appliances to industrial sensors, a vast and easily scalable addressing system is required. IPv6 provides a solution by providing an almost infinite supply of addresses, allowing for effective communication and administration of these devices.
2. **Simplified Header format:** When compared to IPv4, IPv6 delivers a simpler and more efficient header format. This simplified architecture increases routing efficiency and decreases processing overhead on networking devices, resulting in improved network performance.
3. **Security Enhancements:** IPv6 has built-in support for IPsec (Internet Protocol Security), a set of protocols that provides authentication and encryption for data transported over the internet. While IPsec was optional in IPv4, its presence in IPv6 facilitates secure communication with no additional setups.
4. **Address Configuration:** IPv6 provides better address configuration techniques, making it easier for devices to automatically get and configure their IP addresses. This is especially critical in cases where devices change networks often or must be setup dynamically.

5. **Global Reachability:** IPv6 is designed to provide end-to-end connection without the requirement for NAT, which in IPv4 frequently results in devices hidden behind a single IP address. This worldwide accessibility facilitates peer-to-peer communication and contributes to the development of a more decentralized and efficient network.

Introduction to IPv6 and its necessity

- ***Internet Protocol version 6*** (IPv6) is the most recent version of the Internet Protocol, designed to replace IPv4. Because of the rapid development of devices and internet users, IPv4, which has been in use since the early days of the internet, has a restricted address space that is nearly depleted. IPv6 was created to solve the limitations of IPv4 and to the coming address exhaustion situation.

Here are some key aspects of IPv6 and why it's necessary:

1. **Address Space:** One of the key reasons for the development of IPv6 was the expiration of accessible IPv4 addresses. IPv4 addresses are 32 bits long, providing for approximately 4.3 billion distinct addresses. IPv6, on the other hand, employs 128-bit addresses, allowing for a far bigger pool of addresses—over 340 undecillion (3.4 x 1038) unique addresses. This numerous of addresses assures that every device, service, and object that requires internet access has its own unique IP address.
2. **Scalability:** As the number of devices and users connected to the internet has grown, IPv6's vast address space enables for seamless growth without the need for sophisticated address management mechanisms like IPv4's Network Address Translation (NAT). This makes network administration and routing easier.

Rules to represent IPv6:

1. If at least two blocks(segment) contain consecutive zeros, omit them all and replace with double colon sign(::)

   FFFF:A890:CDEF:0000:0000:A001:00AB:AD00

   can be written as FFFF:A890:CDEF::A001:00AB:AD00

2. (::) must be used to represent the largest number of 16 bits

   sets of zero as possible

   FFAB:0000:0000:ABDC:0000:0000:0000:ABAA

   can be written as FFAB:0000:0000:ABDC::ABAA

3. Remove leading zeros

FFFF:ABCD:00CD:A789:0000:0000:00AB:0A79
can be written as FFFF:ABCD:CD:A789::AB:A79

4. If there are multiple places where(::) can be used and the numbers of zeros are the same ,use(::) on the left most set of zeros

FFFF:0000:0000:AB00:000A:0000:0000:A978
can be written as FFFF::AB00:A:0:0:A978

5 .(::) cannot be used to shorten a single 16 bit set of zero
FFFF:0000:ABCD:EFAB:1000:0011:A983:8977
can be written as FFFF:0:ABCD:EFAB:1000:11:A983:8977

IPv4 VS IPv6 :

Feature	IPv4	IPv6
Address Length	32-bit address scheme	128-bit address scheme
Address Notation	Decimal, e.g., 192.168.1.1	Hexadecimal, e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334
Number of Addresses	Approximately 4.3 billion	Approximately 340 undecillion (3.4 x 10^38)
Header Complexity	More complex with 12 fields	Simpler with fewer fields
Subnetting	Supports subnetting	More flexible and easier to manage
Network Address Translation (NAT)	Commonly used to extend address space	Generally not used; direct end-to-end communication
Broadcasting	Supports broadcasting	No broadcasting; uses multicast and anycast
Security	Optional, provided through IPsec and other protocols	Mandatory, built into the protocol (IPsec)
Configuration	Manual or via DHCP	Automatic via SLAAC or manual via DHCPv6

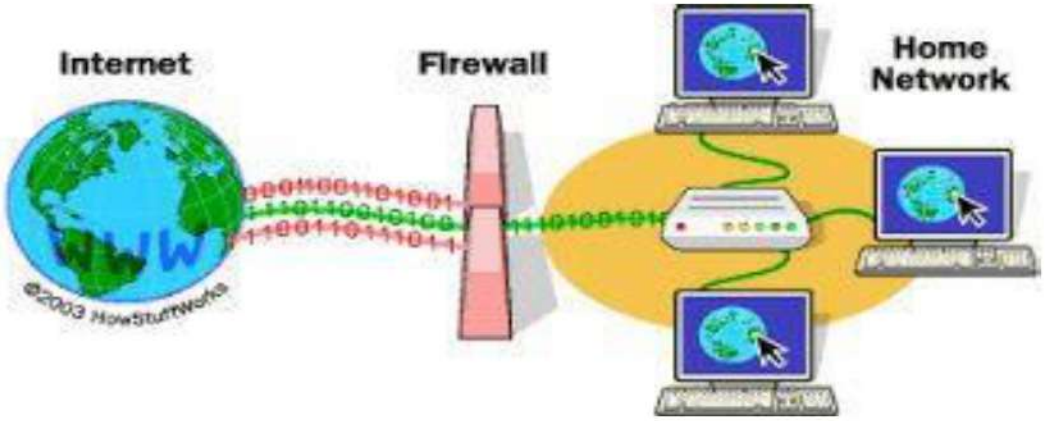
Address Types	Unicast, Broadcast, Multicast	Unicast, Multicast, Anycast
ARP (Address Resolution Protocol)	Used to map IP addresses to MAC addresses	Neighbor Discovery Protocol (NDP) performs this function
Fields	IPv4 is a numeric address that consists of 4 fields which are separated by dot (.).	IPv6 is an alphanumeric address that consists of 8 fields, which are separated by colon.

IPv6 Autoconfiguration :

- Every node in the network requires a unique IP address to communicate and exchange data with other nodes. There are multiple ways to configure IP addresses on nodes. One such way is the address autoconfiguration. The address autoconfiguration is a feature of IPv6. It allows nodes to automatically configure IPv6 addresses for them.
- The IPv6 address consists of 128 binary bits. These bits are divided into two equal portions. The first 64 bits are known as the **network ID** (*network address*) and the last 64 bits are known as the **interface ID** (*host address*). An interface ID identifies the interface in the subnet. A network ID identifies a group of interfaces in the network.
- **A stateful address assignment** involves a server or other device that keeps track of the state of each assignment. It tracks the address pool availability and resolves duplicated address conflicts. It also logs every assignment and keeps track of the expiration times.
- **Stateless address assignment** means that **no server keeps track** of what addresses have been assigned and what addresses are still available for an assignment. Also in the stateless assignment scenario, nodes are responsible to resolve any duplicated address conflicts following the logic: Generate an IPv6 address, run the Duplicate Address Detection (DAD), if the address happens to be in use, generate another one and run DAD again, etc.

Firewall:

- A firewall is a piece of hardware or software for network security that is intended to keep an eye on, filter out, and regulate incoming and outgoing network traffic in accordance with pre-established security standards. A firewall's main objective is to create a wall between a trusted internal network and unreliable external networks, such the internet, in order to safeguard the internal network from intruders, hackers, and other security threats.
- Firewalls can be implemented in various ways, including as hardware appliances, software applications, or a combination of both.



1. **Packet Filtering Firewall:** This is the most basic type of firewall. It examines individual packets of data as they pass through the network and allows or blocks them based on predetermined rules. Packet filtering firewalls use information like source and destination IP addresses, port numbers, and protocols to make filtering decisions.
2. **Stateful Inspection Firewall (Stateful Firewall):** It is also a type of packet filtering that is used to control how data packets move through a firewall. It is also called dynamic packet filtering. These firewalls can inspect that if the packet belongs to a particular session or not. It only permits communication if and only if, the session is perfectly established between two endpoints else it will block the communication.
3. **Proxy Firewall (Application-Level Firewall):** Proxy firewalls act as intermediaries between a user's device and the destination server. They receive requests from the user, make the request to the server on behalf of the user, receive the response, and then forward it to the user. This process can hide the user's true IP address and provide additional security by isolating internal network details from external entities.
4. **Next-Generation Firewall (NGFW):** NGFWs combine traditional firewall functionality with advanced features such as intrusion detection and prevention, deep packet inspection, application-aware filtering, and more. They offer more sophisticated security mechanisms to deal with modern threats and often provide better visibility into network traffic.
5. **Deep Packet Inspection (DPI) Firewall:** DPI firewalls inspect the actual content of packets, looking beyond just header information. They analyze the data within packets to identify specific applications, protocols, or even malware patterns. This allows them to make more informed filtering decisions based on the actual content being transmitted

Administering TCP/IP Networks:

- Administering TCP/IP (Transmission Control Protocol/Internet Protocol) networks involves the management, configuration, and troubleshooting of devices and communication protocols that operate within a network infrastructure. Here are some key areas to consider when administering TCP/IP networks:

1. IP Addressing

- **IPv4 & IPv6 Addressing:** Assigning IP addresses to devices is fundamental. IPv4 uses a 32-bit address space, while IPv6 uses 128-bit.
- **Subnetting:** Dividing an IP network into subnets helps manage network traffic and improve performance. This includes understanding subnet masks, CIDR notation, and calculating subnets.
- **DHCP (Dynamic Host Configuration Protocol):** Automates the assignment of IP addresses and other network configuration details to devices.
- **NAT (Network Address Translation):** Translates private IP addresses to a public IP address to enable internet access for devices within a network.

2. Routing and Switching :

- **Routing Protocols:** TCP/IP networks rely on routing protocols like OSPF (Open Shortest Path First), BGP (Border Gateway Protocol), and RIP (Routing Information Protocol) to direct traffic between networks.
- **Switching:** At Layer 2 of the OSI model, switches help in efficiently forwarding data frames within a local area network (LAN). VLANs (Virtual Local Area Networks) can be configured to segment network traffic.
- **Default Gateways:** Devices use the default gateway to route traffic outside of the local network. The default gateway is often the router's IP address.

3. TCP/IP Protocols and Services :

- **TCP vs. UDP:** TCP provides reliable, connection-oriented communication, while UDP is a faster, connectionless protocol used for streaming and DNS.
- **DNS (Domain Name System):** Translates domain names to IP addresses.
- **FTP, HTTP, HTTPS, SMTP:** Common application layer protocols used for transferring files, web browsing, and sending emails, respectively.
- **ICMP (Internet Control Message Protocol):** Used for diagnostics and error reporting, e.g., "ping" commands to check network connectivity.

4. Security :

- **Firewalls:** Filter incoming and outgoing traffic based on predefined security rules.
- **VPN (Virtual Private Networks):** Securely connects remote users to the network over the internet.
- **Encryption:** Securing data with SSL/TLS (for HTTPS) or IPSec for VPNs.
- **Access Control Lists (ACLs):** Used on routers and firewalls to control which traffic is allowed into or out of the network.

5. Monitoring and Management Tools :

- **SNMP (Simple Network Management Protocol):** Used to monitor and manage network devices like routers and switches.
- **Syslog:** Centralized logging for network devices.
- **Network Monitoring Tools:** Applications like Nagios, SolarWinds, or Wireshark are used to monitor network performance, detect anomalies, and troubleshoot issues

6. Troubleshooting TCP/IP Networks

- **Ping and Traceroute:** Basic tools for checking connectivity and tracing the path of packets through the network.
- **Packet Capture:** Using tools like Wireshark to analyze network traffic at a granular level.
- **Network Diagnostics:** Testing cable connections, checking routing tables, verifying IP configurations, and using command-line tools like ipconfig ,ifconfig , netstat.

7. Quality of Service (QoS)

- **Traffic Prioritization:** QoS settings help ensure that critical services like VoIP or video conferencing
  have the required bandwidth and low latency.

8. Network Redundancy and High Availability

- **Load Balancing:** Distributes network traffic across multiple servers to prevent overload.
- **Redundant Links:** Ensuring multiple paths are available for critical network segments.
- **Failover Systems:** Standby systems that take over if primary systems fail.

9.IPv6 Considerations

- **Migration from IPv4 to IPv6:** As IPv4 addresses are exhausted, IPv6 adoption is growing. Admins need to handle dual-stack configurations and ensure compatibility.
- **IPv6 Addressing and Stateless Address Auto-configuration (SLAAC):** New addressing schemes and auto-configuration options available in IPv6.

Unit 7

Switch :

- A network switch connects devices in a network to each other, enabling them to talk by exchanging data packets. Switches can be hardware devices that manage physical networks or software-based virtual devices.
- A network switch operates on the data-link layer, or Layer 2, of the Open Systems Interconnection (OSI) model. In a local area network (LAN) using Ethernet, a network switch determines where to send each incoming message frame by looking at the media access control (MAC) address. Switches maintain tables that match each MAC address to the port receiving the MAC address.

Types of Switch

1. **Unmanaged switches:** These switches have a simple plug-and-play design and do not offer advanced configuration options. They are suitable for small networks or for use as an expansion to a larger network.
2. **Managed switches:** These switches offer advanced configuration options such as VLANs, QoS, and link aggregation. They are suitable for larger, more complex networks and allow for centralized management.
3. **PoE switches:** These switches have Power over Ethernet capabilities, which allows them to supply power to network devices over the same cable that carries data.
4. **Gigabit switches:** These switches support Gigabit Ethernet speeds, which are faster than traditional Ethernet speeds.
5. **Rack-mounted switches:** These switches are designed to be mounted in a server rack and are suitable for use in data centers or other large networks.
6. **LAN Switch** – : Local Area Network (LAN) switches connects devices in the internal LAN of an organization. They are also referred as Ethernet switches or data switches. These switches are particularly helpful in reducing network congestion or bottlenecks. They allocate bandwidth in a manner so that there is no overlapping of data packets in a network.

Routing and its necessity:

- Routing is a process which is performed by layer 3 (or network layer) devices in order to deliver the packet by choosing an optimal path from one network to another.
- A Routing is a process of selecting path along which the data can be transferred from source to the destination. Routing is performed by a special device known as a router.
- Routing is process of establishing the routes that data packets must follow to reach the destination. In this process, a routing table is created which contains information regarding routes which data packets follow.
- A Router works at the network layer in the OSI model and internet layer in TCP/IP model. A router is a networking device that forwards the packet based on the information available in the packet header and forwarding table.
- In internetworking, the process of moving a packet of data from source to destination. Routing is usually performed by a dedicated device called a router.
- Routing is a key feature of the Internet because it enables messages to pass from one computer to another and eventually reach the target machine

Types of Routing:

1. **Static Routing:** In static routing, network administrators manually configure the routing tables of routers. The paths that data will take are predetermined and do not change automatically, even if the network topology changes. While simple and easy to implement, static routing is not suitable for large, complex networks that undergo frequent changes.

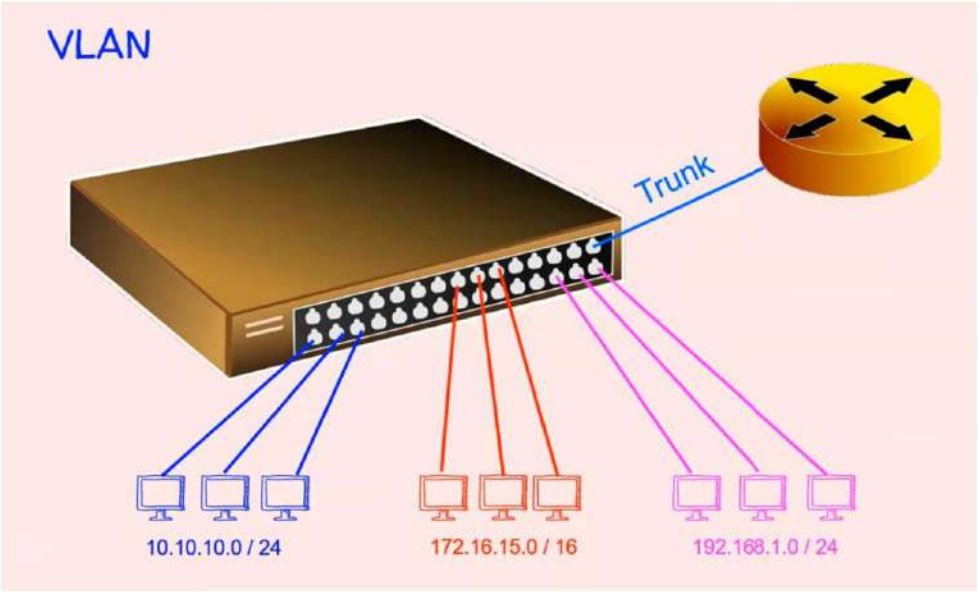
2. **Dynamic Routing:** Dynamic routing protocols allow routers to exchange information about the network's current state, enabling them to dynamically update their routing tables based on real-time information. This adaptive nature allows dynamic routing to respond to changes in network topology, link failures, and traffic conditions. Common dynamic routing protocols include OSPF (Open Shortest Path First) and RIP (Routing Information Protocol).

Static Routing VS Dynamic Routing

Static Routing	Dynamic Routing
In static routing routes are user-defined.	In dynamic routing, routes are updated according to the topology.
Static routing does not use complex routing algorithms.	Dynamic routing uses complex routing algorithms.
Static routing provides high or more security.	Dynamic routing provides less security.
Static routing is manual.	Dynamic routing is automated.
Static routing is implemented in small networks.	Dynamic routing is implemented in large networks.
Another name for static routing is non-adaptive routing.	Another name for dynamic routing is adaptive routing.

VLAN (Virtual Local Area Network):

VLAN is a method of partitioning a single physical network into multiple logical networks. It enables network administrators to group devices together virtually, regardless of their physical location. This grouping is based on factors such as department, function, or security requirements . VLANs help improve network performance, security, and manageability.



Here's how VLANs work:

1. **Logical Segmentation**: Instead of creating separate physical networks for different groups of devices, VLANs provide a way to create isolated broadcast domains within a single physical network.

2. **Grouping Devices**: Devices are grouped into VLANs based on certain characteristics, such as department, function, or security requirements. For example, all devices belonging to the HR department can be placed in one VLAN, while devices from the IT department are placed in another.

3. **Communication between VLANs**: By default, devices within a VLAN can communicate with each other, but they are isolated from devices in other VLANs. If communication between VLANs is required, a router or a Layer 3 switch is used to route traffic between them.

4. **Security**: VLANs improve security by segregating sensitive data or critical systems from general network traffic. This way, unauthorized devices have a harder time accessing sensitive information.

5. **Management Flexibility**: VLANs allow network administrators to make changes to the logical structure of the network without physically rewiring it. Devices can be moved from one VLAN to another easily through configuration changes.

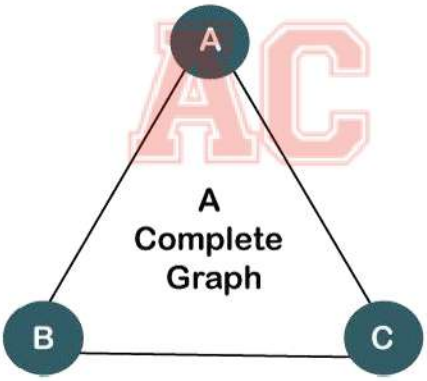
Advantages of VLAN:

1. VLAN allows you to add an additional layer of security. The message broadcast in one group cannot be listened by members of other groups
2. It can make device management simple and easier.
3. You can make a logical grouping of devices by function rather than location.
4. It allows you to create groups of logically connected devices that act like they are on their own network.
5. VLAN removes the physical boundary.
6. It lets you easily segment your network.
7. It helps you to enhance network security.
8. You can keep hosts separated by VLAN.
9. You do not require additional hardware and cabling, which helps you to saves costs.

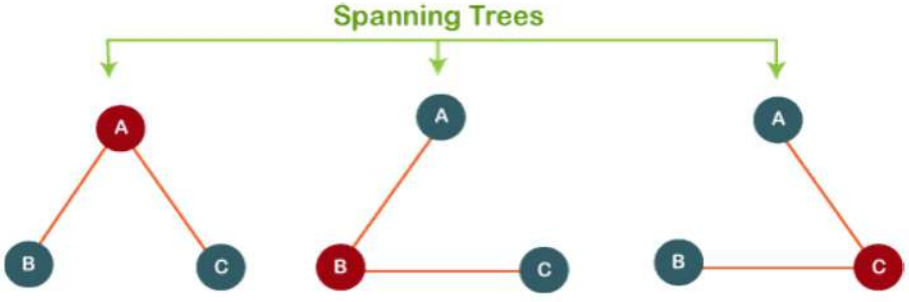
Spanning tree:

- The spanning tree protocol is a layer 2 protocol that tends to solve the problems when the computers use the shared telecommunications paths on a local area network. When they share the common path, if all the computers send the data simultaneously, it affects the overall network performance and brings all the network traffic near a halt.
- The spanning tree protocol (STP) overcomes this situation by using the concept of bridge looping. Bridge looping is used when there are multiple connections between the two endpoints, and messages are sent continuously, which leads to the flooding of the network. To remove the looping, STP divides the LAN network into two or more segments with the help of a device known as bridges. The bridge is used to connect the two segments so when the message is sent, the message is passed through the bridge to reach the intended destination. The bridge determines whether the message is for the same segment or a different segment, and it works accordingly. This network segmentation greatly reduces the chances of a network coming to a halt.

- Suppose there are three points, i.e., A, B, and C. Three lines connect these three points. A line connects every two-point, and we get a complete graph.



The complete graph is formed when a maximum number of lines connects all the points, whereas the spanning tree is formed when a minimum number of lines connects all the points



1. A is directly connected to B and C, while B and C are indirectly connected through A. In this spanning
   tree, A is a central point and all the points are connected without any formation of loops.
2. B is directly connected to A and C, while A and C are connected through B. B is a bridge between A and C, or we can say that B is a central point. In this case, also, all the points are connected without any formation of loops.
3. C is directly connected to both A and B, while A and B are connected through C. Therefore, C is a bridge between A and B, and C is a central point. In this case, all the points are connected without any formation of loops

How spanning tree protocol works?

1. This protocol selects one switch as a root bridge where the root bridge is a central point as when the message is sent; then it always passes through the bridge.
2. It selects the shortest path from a switch to the root bridge.
3. It blocks the links that cause the looping on a network, and all the blocked links are maintained as backups. It can also activate the blocked links whenever the active link fails. Therefore, we can say that it also provides fault tolerance on a network

Unit 8

Webmin :

Webmin is a powerful web-based interface designed to manage Unix-like systems, such as Linux, FreeBSD, Solaris, and so forth. It eliminates the need for system administrators to physically alter configuration files in order to configure and administer a server.

Features:

- **User and Group Management:** Add, remove, and modify users and groups.
- **Service Management:** Manage services like Apache, MySQL, SSH, FTP, and more.
- **Network Configuration:** Configure networking, DNS settings, routing, and more.
- **Package Management:** Install, remove, and update software packages.
- **Disk and Filesystem Management:** Partition management, mount points, and filesystem checks.
- **Security:** SSL management, firewall configuration, and user permissions.
- **Automation:** Webmin offers scheduling and cron job configuration.
- **Modular Design:** Webmin is modular, meaning additional features can be added through modules.

Usermin :

Purpose: Usermin is more user-focused and is designed to give regular users control over personal account settings. It does not provide administrative functionality but focuses on tasks a non-admin user would need, such as:

Managing emails (reading, sending, filtering)

Changing passwords

Managing files and file permissions

Viewing logs related to their own

account Configuring their own SSH

keys

Setting up cron jobs for personal tasks

Key Differences

Audience:

Webmin: For system administrators managing the entire server.

Usermin: For individual users managing their personal settings.

Scope:

Webmin: Full system administration and server management.

Usermin: Limited to user-specific tasks like email,password management, etc.

TELNET:

- ***TELNET*** stands for Teletype Network. It is a client/server application protocol that provides access to virtual terminals of remote systems on local area networks or the Internet. The local computer uses a telnet client program and the remote computers use a telnet server program.
- TELNET is a type of protocol that enables one computer to connect to the local computer. It is used as a standard TCP/IP protocol for virtual terminal service which is provided by ISO. The computer which starts the connection is known as the local computer. The computer which is being connected to i.e. which accepts the connection known as the remote computer. During telnet operation, whatever is being performed on the remote computer will be displayed by the local computer. Telnet operates on a client/server principle.

SSH:

- The SSH (Secure Shell) is an access credential that is used in the SSH Protocol. In other words, it is a cryptographic network protocol that is used for transferring encrypted data over the network. The port number of SSH is 22.Secure Shell or SSH, is a protocol that allows you to connect securely to another computer over an unsecured network. It developed in 1995. SSH was designed to replace older methods like Telnet, which transmitted data in plain text.

- **Features of SSH:**
  - **Encryption:** Encrypted data is exchanged between the server and client, which ensures confidentiality and prevents unauthorized attacks on the system.
  - **Authentication:** For authentication, SSH uses public and private key pairs which provide more security than traditional password authentication.
  - **Data Integrity:** SSH provides Data Integrity of the message exchanged during the communication.
  - **Tunneling:** Through SSH we can create secure tunnels for forwarding network connections over encrypted channels.

Telnet Vs SSH :

Feature	Telnet	SSH
Full Form	Teletype Network	Secure Shell
Security	No encryption; data is transmitted in plain text	Encrypted; provides secure data transmission
Authentication	Username and password sent in plain text	Uses public key cryptography or encrypted passwords
Encryption	None	Strong encryption (e.g., RSA, AES)
Port Number	Default port 23	Default port 22
Usage	Legacy and less secure remote management	Secure remote management of network devices/servers
Data Confidentiality	None	Ensures confidentiality and integrity of data
Data Integrity	No data integrity checks	Provides data integrity through cryptographic algorithms
Platform	Supported on most platforms but not recommended for security reasons	Supported on all major platforms and highly recommended
Performance	Faster, due to no encryption overhead	Slower than Telnet due to encryption
Common Usage Scenarios	Debugging or managing legacy systems	Secure administration of systems and networks

SCP:

- **scp** (secure copy) command in Linux system is used to copy file(s) between servers in a secure way. The SCP command or secure copy allows the secure transferring of files between the local host and the remote host or between two remote hosts. It uses the same authentication and security as it is used in the Secure Shell (SSH) protocol.

rsync:

- **rsync** or remote synchronization is a software utility for Unix-Like systems that efficiently sync files and directories between two hosts or machines. One is the source or the local- host from which the files will be synced, the other is the remote-host, on which synchronization will take place. There are basically two ways in which *rsync* can copy/sync data:
- Copying/syncing to/from another host over any remote shell like *ssh*, *rsh*.
- Copying/Syncing through rsync daemon using TCP

NFS Vs DFS :

Feature	Network File System (NFS)	Distributed File System (DFS)
Definition	A protocol that allows users to access files over a network as if they were on a local drive.	A file system where files are stored across multiple servers and locations, appearing as one unified system.
Architecture	Centralized: usually involves a single server providing files to multiple clients.	Decentralized: typically involves multiple servers storing and managing files collaboratively.
Scalability	Limited scalability due to reliance on a single server.	Highly scalable, as more servers can be added to handle larger datasets and traffic.
Fault Tolerance	Limited fault tolerance; server failure affects file access for clients.	High fault tolerance, as files are replicated across multiple nodes.
Performance	Can be slower due to the single-server bottleneck, especially under high loads.	Generally faster, as load is distributed across multiple servers, reducing bottlenecks.
Data Replication	Data replication is typically not inherent; relies on external backups.	Built-in data replication, where files are often copied across multiple nodes for redundancy.
Data Consistency	Maintains strong consistency; changes on the server are immediately visible to clients.	Consistency can vary (e.g., eventual consistency), depending on the DFS design.

Use Cases	Suitable for small to medium-sized networks, such as local networks and enterprise file sharing.	Ideal for large-scale applications, like cloud storage, big data processing, and content delivery networks.
Examples	NFS, CIFS (Common Internet File System).	HDFS (Hadoop Distributed File System), Google File System (GFS), Amazon S3.
Management Complexity	Relatively simple to set up and manage, with fewer components.	More complex due to the distributed architecture and data replication requirements.

***