COMMUNICATIONS AND NETWORK SECURITY

Third Stage

INTRODUCTION

DEFINTION:

A computer network is defined as the interconnection of two or more computers. It is done to enable the computers to communicate and share available resources.

APPLICATIONS:

i. Sharing of resources such as printers
ii. Sharing of expensive software's and database
iii. Communication from one computer to another computer
iv. Exchange of data and information among users via
v. network
vi. Sharing of information over geographically wide areas.

COMPONENTS OF COMPUTER NETWORK

- Two or more computers
- Cables as links between the computers
- A network interfacing card (NIC) on each computer
- Switches
- Software called operating system (OS)

NETWORK BENEFITS

- Sharing
- Connectivity
- Increased speed
- Reduced cost
- Improved security
- Centralized software managements
- Electronic mail
- Flexible access

DISDAVATAGES OF NETWORKS

- High cost of installation
- Requires time for administration
- Failure of server
- Cable faults

SHARING RESOURCES

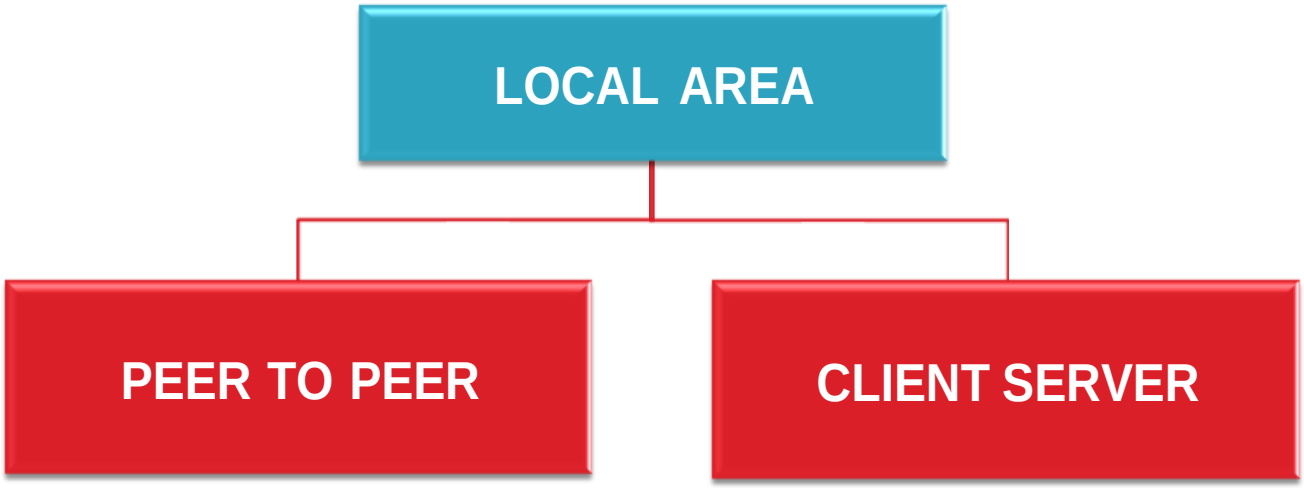
Types of resources are:

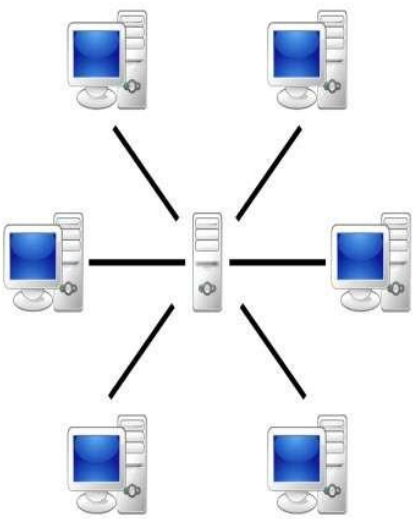
1. **Hardware**: A network allows users to share many hardware devices such as printers, modems, fax machines, CD ROM, players, etc.
2. **Software**: sharing software resources reduces the cost of software installation, saves space on hard disk.

DISTINGUISH BETWEEN LAN, WAN, MAN

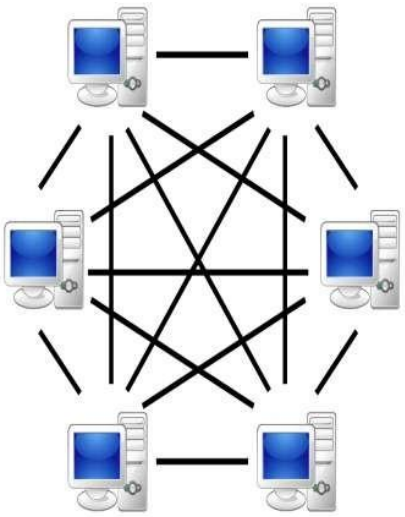
PARAMETERS	LAN	WAN	MAN
Ownership of network	Private	Private or public	Private or public
Geographicalarea covered	Small	Very large	Moderate
Design and maintenance	Easy	Not easy	Not easy
Communication medium	Coaxial cable	PSTN or satellite links	Coaxial cables, PSTN, optical fibre, cables, wireless
Bandwidth	Low	High	moderate
Data rates(speed)	High	Low	moderate

NETWORK CLASSIFICATION BY THEIR COMPONENT ROLE





Server-based



P2P-network

PEER TO PEER NETWORK

- In peer to peer network each computer is responsible for making its own resources available to other computers on the network.
- Each computer is responsible for setting up and maintaining its own security for these resources.
- Also each computer is responsible for accessing the required network resources from peer to peer relationships.
- Peer to peer network is useful for a small network containing
- less than 10 computers on a single LAN .
- In peer to peer network each computer can function as both client and server.
- Peer to peer networks do not have a central control system.
- There are no servers in peer networks.
- Peer networks are amplified into home group.

ADVANTAGES & DISADVANTAGES OF PEER TO PEER NETWORK

Advantages:

- Use less expensive computer hardware
- Easy to administer
- No NOS required
- More built in redundancy
- Easy setup & low cost

Disadvantages:

- Not very secure
- No central point of storage or file archiving
- Additional load on computer because of resource sharing
- Hard to maintain version control

CLIENT/SERVER NETWORK

- In client-server network relationships, certain computers act as server and other act as clients. **A server is** simply a computer, that available the network resources and provides service to other computers when they request it. **A client is** the computer running a program that requests the service from a server.
- Local area network (LAN) is based on client server network relationship.
- A client-server network is one in which all available network resources such as files, directories, applications and shared devices, are centrally managed and hosted and then are accessed by client.
- Client serve network are defined by the presence of servers on a network that provide security and administration of the network.

ADVANTAGES AND DISADVANTAGES OF CLIENT- SERVER NETWORK

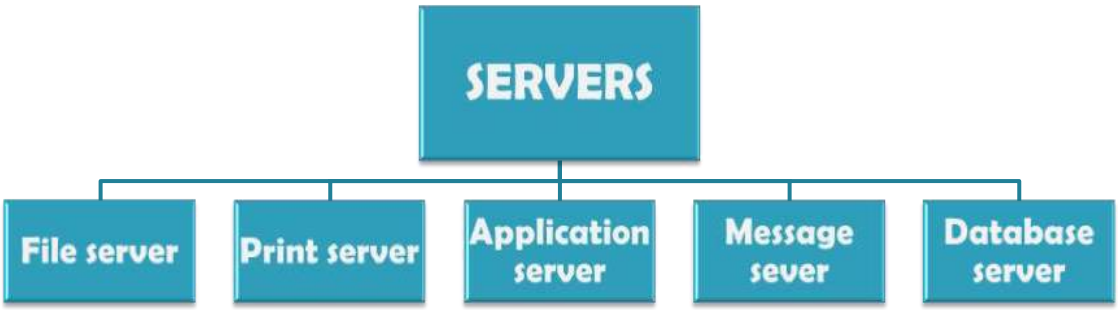
Advantages:

- Very secure
- Better performance
- Centralized backup
- very reliable

Disadvantages:

- requires professionaladministration
- More hardware-intensive
- More softwareintensive
- Expensive dedicatedsoftware

TYPES OF SERVERS



- **File server:** These servers provide the services for storing, retrieving and moving the data. A user can read, write, exchange and manage the files with the help of file servers.
- **Printer server:** The printer server is used for controlling and managing printing on the network. It also offers the fax service to the network users.
- **Application server:** The expensive software and additional computing power can be shared by the computers in a network with he help of application servers.
- **Message server:** It is used to co-ordinate the interaction between users, documents and applications. The data can be used in the for of audio, video, binary, text or graphics.
- **Database server:** It is a type of application server. It allows the uses to access the centralised strong database.

Client-server architecture

- There is always-on host (server). The server services requests from many other hosts (clients). For instance, the Web application, FTP, Telnet, e-mail.
- Clients do not communicate with each other directly.
- The server has a fixed, well-known IP address (since it is always ON).
- One server is incapable of keeping up with all requests from its clients. Therefore, a large cluster of servers (called a datacenter) is used to create a powerful virtual server in client-server architecture.

Peer-to-peer architecture

- The application exploits direct communication between pairs of intermittently connected hosts, called peers.
- No servers involved.
- Examples: Bit Torrent, Skype.

HTTP

- HTTP, hypertext transfer protocol, the Web's application- layer protocol, is at the heart of the Web.
- http is implemented in two programs: a client program and a server program. The client program and the sever program talk to each other by exchanging http messages.
- A Web page (also called a document) consists of objects. An object is simply a file—such as an HTML file, a JPEG image, a Java applet, or a video clip—that is addressable by a single URL (Uniform Resource Locator).
- Most Web pages consist of a base HTML file and several referenced objects. For example, if a Web page contains HTML text and five JPEG images, then the Web page has six objects: the base HTML file plus the five images.
- The base HTML file references the other objects in the page with the objects' URLs. Each URL has two components: the hostname of the server that houses the object and the object's path name. For example, the URL http://www.anbar.edu/EEDept/picture.gif has www.anbar.edu for a hostname and / EEDept /picture.gif for a path name. Web browsers (such as Internet Explorer and Firefox) implement the client side of HTTP. Web servers, which implement the server side of HTTP, house Web objects, each addressable by a URL. Popular Web servers include Apache and Microsoft Internet Information Server.

- When user requests a web page (for example, clicking on a hyperlink), the browser sends http request messages for the objects in the page to the server. The server receives the requests and responds with http response messages that contain the objects.
- http uses TCP. This implies that http messages eventually arrive at their destination intact.
- http is a stateless protocol; the server sends requested files to clients without storing any state information about the client.

Persistent and Non-persistent connections

Non-persistent: each request/response pair is sent over a separate TCP connection.

Persistent: all of the requests and their corresponding responses are sent over the same TCP connection.

Example:

Let's suppose the page consists of a base HTML file and 10 JPEG images, and that all 11 of these objects reside on the same server. Further suppose the URL for the base HTML file is http://www.anbar.edu/EEDept/home.index

Here is what happens:

1. The HTTP client process initiates a TCP connection to the server www. anbar.edu on port number 80, which is the default port number for HTTP. Associated with the TCP connection, there will be a socket at the client and a socket at the server.
2. The HTTP client sends an HTTP request message to the server via its socket. The request message includes the path name /EEDept/home.index.
3. The HTTP server process receives the request message via its socket, retrieves the object / EEDept/home.index from its storage (RAM or disk), encapsulates the object in an HTTP response message, and sends the response message to the client via its socket.

4. The HTTP server process tells TCP to close the TCP connection. (But TCP doesn't actually terminate the connection until it knows for sure that the client has received the response message intact.)
5. The HTTP client receives the response message. The TCP connection terminates. The message indicates that the encapsulated object is an HTML file. The client extracts the file from the response message, examines the HTML file, and finds references to the 10 JPEG objects.
6. The first four steps are then repeated for each of the referenced JPEG objects. As the browser receives the Web page, it displays the page to the user.

The steps above illustrate the use of non-persistent connections, where each TCP connection is closed after the server sends the object—the connection does not persist for other objects. Note that each TCP connection transports exactly one request message and one response message. Thus, in this example, when a user requests the Web page, 11 TCP connections are generated.

Shortcomings:

1. A brand-new connection must be established and maintained for each requested object.
2. Each object suffers a delivery delay of two RTTs, one RTT to establish the TCP connection and one RTT to request and receive an object.

Persistent: When a server receives back-to-back requests, it sends objects back-to-back.

DNS (Domain Name System)

- Hosts can be identified by hostnames such as cnn.com, www.yahoo.com, www.google.com, www.elect-eng.anbar.edu.
- Hostnames are appreciated by humans, but they do not provide much more information about the location of the host within the Internet.
- Also, hostnames can consist of variable-length alphanumeric characters. Therefore, it would be difficult to process by routers.
- For the above two reasons, host are identified by IP addresses as well.
- DNS translates hostnames to IP addresses.

DNS is:

1. A distributed database implemented in a hierarchy of DNS servers.
2. An application-layer protocol that allows hosts to query the distributed database.

- An IP address 121.7.106.83 each period separates one of the bytes expressed in decimal notation from 0 to 255.
- An IP address is hierarchical because as we scan the address from left to right, we obtain more and more specific information about where the host is

located in the Internet (that is, within which network, in the network of networks). Similarly, when we scan a postal address from top to bottom, we obtain more and more specific information about where the addressee is located.

Example:

A browser (that is, an HTTP client like Google chrome), running on some user's host, requests the URL www.anbar.edu/index.html. In order for the user's host to be able to send an HTTP request message to the Web server www. anbar.edu, the user's host must first obtain the IP address of www. anbar.edu. This is done as follows:

1. The same user machine runs the client side of the DNS application.
2. The browser extracts the hostname, www. anbar.edu, from the URL and passes the hostname to the client side of the DNS application.
3. The DNS client sends a query containing the hostname to a DNS server.
4. The DNS client eventually receives a reply, which includes the IP address for the hostname.
5. Once the browser receives the IP address from DNS, it can initiate a TCP connection to the HTTP server process located at port 80 at that IP address.

So DNS adds an additional delay. How can this issue be partially solved or mitigated?

Services provided by DNS

1. Hostname to IP address translation.

2. **Host aliasing:** DNS can be invoked by an application to obtain the canonical hostname for a supplied alias hostname as well as the IP address of the host.

3. Mail server aliasing:

It is highly desirable that e-mail addresses be mnemonic. For example, if Bob has an account with Hotmail, Bob's e-mail address might be as simple as bob@hotmail.com. However, the hostname of the Hotmail mail server is more complicated and much less mnemonic than simply hotmail.com (for example, the canonical hostname might be something like relay1.west-coast.hotmail.com). DNS can be invoked by a mail application to obtain the canonical hostname for a supplied alias hostname as well as the IP address of the host.

4. Load distribution:

DNS is also used to perform load distribution among replicated servers, such as replicated Web servers. Busy sites, such as cnn.com, are replicated over multiple servers, with each server running on a different end system and each having a different IP address. For replicated Web servers, a set of IP addresses is thus associated with one canonical hostname. The DNS database contains this set of IP addresses. When

clients make a DNS query for a name mapped to a set of addresses, the server responds with the entire set of IP addresses, but rotates the ordering of the addresses within each reply. Because a client typically sends its HTTP request message to the IP address that is listed first in the set, DNS rotation distributes the traffic among the replicated servers.

Overview of How DNS Works

DNS works like the following example:

I. Suppose that some application (such as a Web browser or a mail reader) running in a user's host needs to translate a hostname to an IP address.

II. The application will invoke the client side of DNS, specifying the hostname that needs to be translated.

III. DNS in the user's host then takes over, sending a query message into the network.

IV. All DNS query and reply messages are sent within UDP datagrams to port 53.

V. After a delay, ranging from milliseconds to seconds, DNS in the user's host receives a DNS reply message that provides the desired mapping.

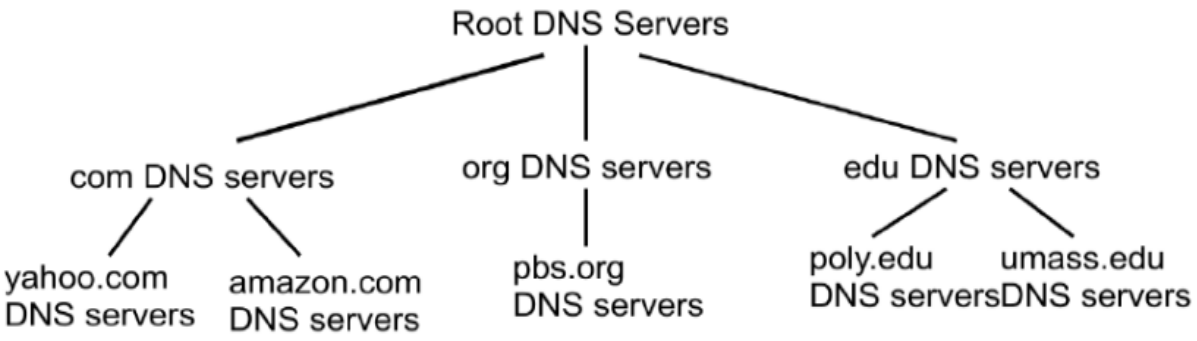
VI. This mapping is then passed to the invoking application.

Why not centralize DNS?

- single point of failure
- traffic volume
- distant centralized database
- maintenance
- doesn’t scale

DNS: a distributed, hierarchical database

1. **Root DNS servers**. In the Internet there are 13 root DNS servers (labeled A through M), most of which are located in North America.
2. **Top-level domain (TLD) servers.** These servers are responsible for top-level domains such as com, org, net, edu, and gov, and all of the country top-level domains such as uk, fr, ca, and jp.
3. **Authoritative DNS servers.** Every organization with publicly accessible hosts (such as Web servers and mail servers) on the Internet must provide publicly accessible DNS records that map the names of those hosts to IP addresses.



client wants IP for www.amazon.com; 1st approx:

- client queries root server to find com DNS server
- client queries .com DNS server to get amazon.com DNS server
- client queries amazon.com DNS server to get IP address for www.amazon.com

DNS name resolution example

- host at cis.poly.edu wants IP address for gaia.cs.umass.edu

iterated query:

- contacted server replies with name of server to contact
- "I don't know this name, but ask this server"