UNIT-6

Application Layer

The application layer is the highest abstraction layer of the TCP/IP model that provides the interfaces and protocols needed by the users. This layer is implemented by network applications. These applications produce the data, which has to be transferred over the network. This layer also serves as a window for the application services to access the network and for displaying the received information to the user.

Examples: Applications, Browsers, Skype, Messenger etc.

Functions of Application Layer:

File transfer → It allows user to access, retrive and manage files in a remote computer.

Mail Services → It provides the basis for email forwarding and storage facilities.

Directory services → It provides ~~database sources~~ access for global information about various services.

⊛ Web & HTTP:

The World Wide Web (WWW), or simply Web, is a global network of servers linked by a common protocol allowing access to all connected hypertext resources. When a client host requests an object, a Web server responds by sending the requested object through browsing tools. The WWW has unique combination of flexibility, portability and user friendly features that distinguish it from other services provided by the Internet. The WWW today is a distributed client-server service, in which a client using a browser can access a service using the server. However, the service provided is distributed over many locations called websites.

HTTP stands for Hyper Text Transfer Protocol that transfers the page browsed by user at the application layer. HTTP uses TCP rather than UDP, since reliability of delivery is important for Web pages with text. In a hypertext environment, information is stored in a set of documents that are linked

together using the concept of pointers. The reader who is browsing through document can move to other documents by clicking the items that are linked to other documents. To use WWW, we need three components: a browser, a web server and a protocal called HTTP.

⑧. HTTP Message Format:

HTTP Message is used to show how data is exchanged between the client and the server. It is based on client-server architecture. HTTP message consists of an initial request line and an initial response line.

Format:

HTTP-Message = Request | Response ; HTTP/1.1 messages

i) Initial Request Line → The initial line is different for the request and the response. A request-line consist of three parts: a method name, requested resource's local path, and the HTTP version being used. All these parts are separated by spaces.

Syntax:

GET /path/to/file/index.html HTTP/1.0

uniform resource identifier

Here,
- → GET is the most common HTTP method.
- → The **path** shows the part of the URL after the host name. It is also called a request URI.
- → The **version** of HTTP always takes the form "HTTP/x.x".

ii) Initial Response Line → The initial response line is also known as the status line. It also has three parts: the HTTP version, a response status code that gives the result of the request and the English reason phrase describing the status code.

Example

HTTP/1.0 200 OK OR HTTP/1.0 404 Not Found.

Here, the HTTP version of the response line and the request line are same as "HTTP/x.x".

⊛ Persistent Versus Nonpersistent Connection:

Persistent Connection → HTTP version 1.1 specifies a persistent connection by default. In a persistent connection, the server leaves the connection open for more requests after sending a response. The server can close the connection at the request of a client or if a time-out has been reached. The sender usually sends the length of the data with each response. However, there are some occasions when the sender does not know the length of the data. This is the case when a document is created dynamically or actively. In these cases, the server informs the client that the length is not known and closes the connection after sending the data so the client knows that the end of the data has been reached.

Nonpersistent Connection → HTTP prior to version 1.1 specifies a non-persistent connection. In a nonpersistent connection, one TCP connection is made for each request/response. The following lists the steps in this strategy:

1. The client opens a TCP connection and sends a request.
2. The server sends the response and closes the connection.
3. The client reads the data until it encounters an end-of-file marker; it then closes the connection.

In this strategy, for N different pictures in different files, the connection must be opend and closed N times. The nonpersistent strategy imposes high overhead on the server because the server needs N different buffers and requires a slow start procedure each time a connection is opened.

⊛ DNS and Query Types:

The Domain Name System (DNS) is a supporting program that is used by the other programs such as e-mail. It is a global system for translating IP addresses to human-readable domain names. When a user tries to access a web address like "example.com", their web browser or application performs a DNS

Query against a DNS server, supplying the host name. The DNS server takes the hostname and resolves it into a numeric IP address, which the web browser can connect to.

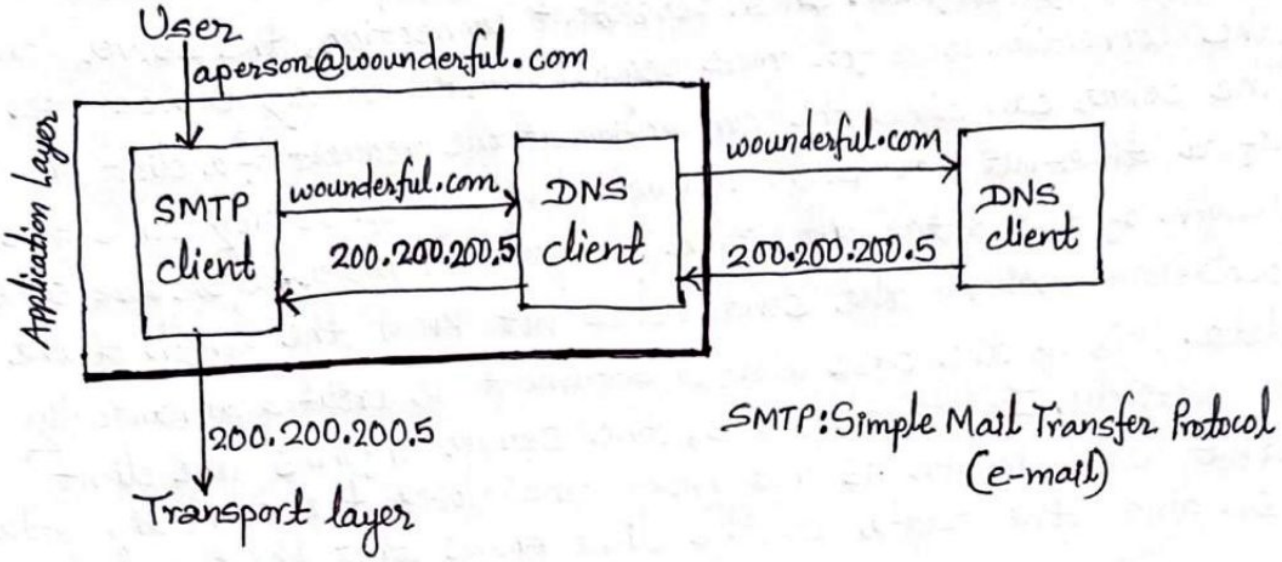


Fig: Example of using DNS service.

Query Types in DNS System:

There are three types of queries in the DNS system:

i) **Recursive Query** → In a recursive query, a DNS client provides a hostname, and the DNS Resolver must provide an answer. DNS Resolver responds with either a relevant resource record, or an error message if it can't be found. The resolver starts a recursive query process, starting from the DNS Root Server, until it finds the Authoritative Name Server that holds the IP address and other information for the requested hostname.

ii) **Iterative Query** → In an iterative query, a DNS client provides a hostname, and the DNS Resolver returns the best answer it can. If the DNS resolver has the relevant DNS records in its cache, it returns them. If not, it refers the DNS client to the Root Server or another Authoritative Name Server which is nearest to the required DNS zone.

iii) **Non-Recursive Query** → In non-recursive query, the DNS Resolver already knows the answer. It either immediately returns a DNS record or queries a DNS Name Server which is authoritative for the record. In both cases, there is no need for additional rounds of queries, rather a response is immediately returned to the client.

Services provided by DNS:

i) **Host Aliasing** → A host with a complicated hostname can have one or more alias names. 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. For example, a hostname such as relay1.west-coast.enterprise.com could have, say, two aliases such as enterprise.com and www.enterprise.com. In this case, the hostname relay1.west-coast.enterprise.com is said to be canonical hostname.

ii) **Mail Server Aliasing** → It is highly desirable that e-mail address be easy to remember. For example, if Bob has an account with Gmail, Bob's e-mail address might be as simple as bob@gmail.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.

iii) **Load Distribution** → DNS is 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 running on a different end system and having a different IP address. For replicated web servers, a set of IP addresses is thus associated with one cannonical hostname.

Overview of How DNS Works:

DNS is a client/server network communication protocol. DNS clients send requests to the server while DNS servers send responses to the client. Client requests contain a name which is converted into an IP address known as a forward DNS lookups while requests containing an IP address which is converted into a name known as a reverse DNS lookups. DNS implements a distributed database to store the name of all the hosts available on the internet.

If a client like a web browser sends a request containing a hostname, then a piece of software such as DNS Resolver sends a request to the DNS server to obtain the

IP address of a hostname. If the DNS server does not contain the IP address associated with a hostname, then it forwards the request to another DNS server. If IP address has arrived at the resolver, which in turn completes the request over the internet protocol.

DNS Records:

DNS records are mainly used to convert domain names into servers IP that host website. The main purpose is that people and applications don't have to remember big numbers to navigate to a domain. For example, www.helloprogrammers.com has an IP of 93.184.220.42, so it is eaiser to remember a friendly name. Following are the different types of DNS records that are used for a webpage, depending on the functions that we need to publish.

A Record → Connects an IP Address to a host name.

CNAME Record → Allows more than one DNS name for a host.

MX Record → Ensures email is delivered to the right location.

NS Record → Contains the name server info.

SRV Record → Finds computers that host specific services.

SPF Record → Used to help prevent against spam.

DNS Messages:

DNS has two types of messages query and response. Both types have the same format. The query message consists of a header and question records. The response message consist of a header, question records, answer records, authoritative records and additional records.

Header → Both query and response messages have the same header format with some fields set to zero for the query messages. The header is 12 bytes.

Question section → This is a section consisting of one or more question records. It is present on both query and response messages.

Answer section → This is a section consisting of one or more resource records. It is present only on response messages. This section includes the answer from the server to the client.

Authoritive Section → This is a section consisting of one or more resource records. It is also present only on response messages. This section gives information (domain name) about one or more authoritative servers for query.

Additional Information Section → This section provides additional information that may help the resolver. For example, a server may give the domain name of an authoritative server to the resolver in the authoritative section and include the IP address of the same authoritative server in the additional information section.

⊛. File Transfer and Email Protocols:-

FTP → File Transfer Protocol (FTP) is a standard Internet protocol for transmitting files between computers on the Internet over TCP/IP connections. FTP is a client-server protocol that relies on two communication channels between client and server: a command channel for controlling the conversation and a data channel for transmitting file content.

FTP differs from other client-server applications in that sense it establishes two connections between the hosts. One connection is used for data transfer and other for control information. Separation of commands and data transfer makes FTP more efficient.

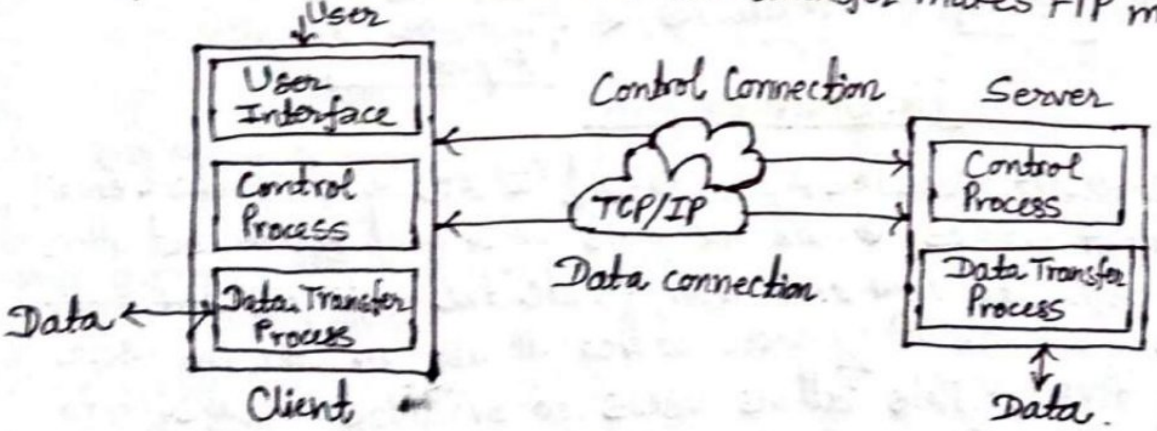


Fig: Control and data connection in FTP.

2) SFTP → SSH File Transfer Protocol (SFTP) is a secure file transfer protocol. It runs over the SSH protocol. It supports the full security and authentication functionality. SFTP also protects against password sniffing and it protects the integrity of the data using encryption and cryptographic hash functions. It also authenticates both the server and the user.

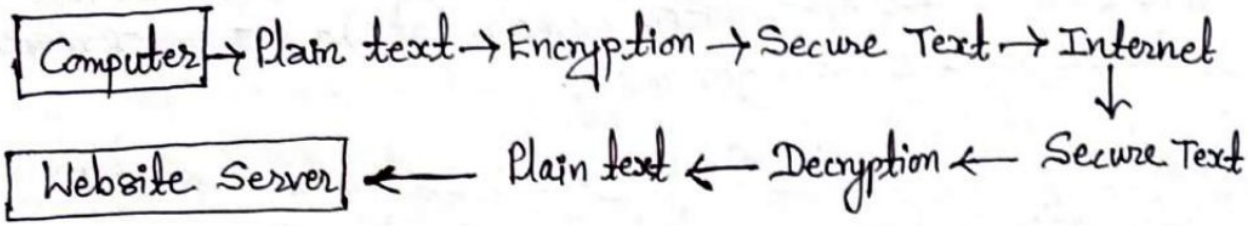


Fig: Process of SFTP

3) SMTP → SMTP stands for "Simple Mail Transfer Protocol". It contains rules for correct communication between computers in a network. SMTP is responsible for feeding and forwarding e-mails from sender to recipient. The SMTP model is of two types: end-to-end method and store-and-forward method.

The end-to-end model is used to communicate between different organizations whereas the store and forward method is used within an organization.

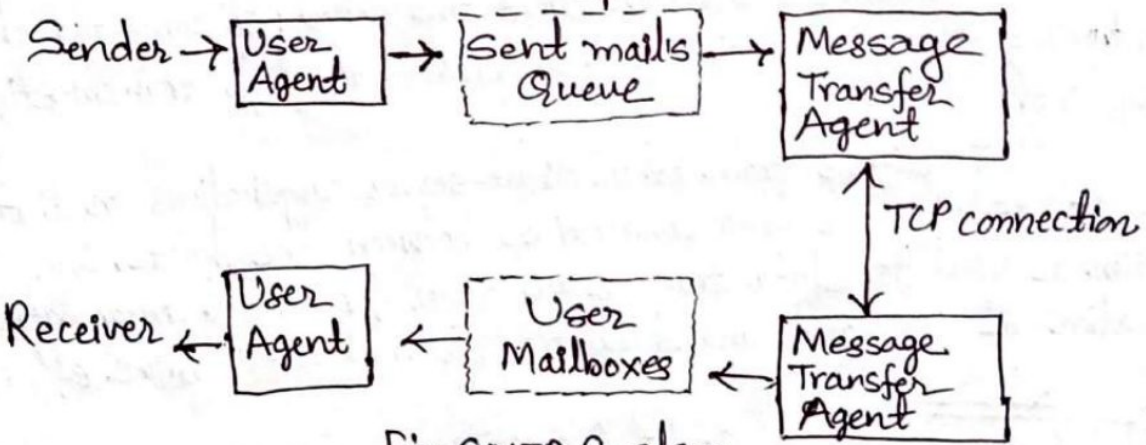


Fig: SMTP System

4) IMAP → Internet Message Access Protocol (IMAP) is a standard email protocol that stores email messages on a mail server, but allows the end user to view and manipulate the messages ~~have been read~~ as though they were stored locally on the end user's computing device. This allows users to organize messages into folders, have multiple client applications know which messages have been read, flag messages for urgency or follow-up and save draft messages on the server.

IMAP supports both on-line and off-line modes of operation. E-mail clients using IMAP generally leave messages on the server until the user explicitly deletes them. To use IMAP, the mail server runs an IMAP server that listenes to port 143.

5). POP(3) → Post Office Protocol version 3 (POP3) is a standard mail protocol used to receive emails from a remote server to a local email client. POP3 allows us to download email messages on our local computer and read them even when we are offline. By default, the POP3 protocol works on two ports:

Port 110: this is a default POP3 non-encrypted port.

Port 995: this is the port we need to use if we want to connect using POP3 securely.

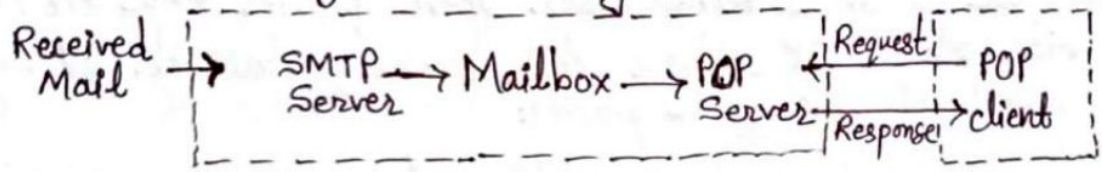


Fig: POP 3

⊛. Overview of Application Server Concepts:

An application server is a component-based product that resides in the middle-tier of a server centric architecture. It provides middleware services for security and state maintenance, along with data access and persistence.

The application server is frequently viewed as a part of three-tier application, consisting of graphical user interface (GUI) server, an application server and a database and transaction server. An application server is a server program in a computer network that provides the business logic for an application program.

1). Proxy Application Server → A proxy server is any machine that translates traffic between networks or protocols. It's an intermediary server separating end-user clients from the destinations that they browse. Proxy servers provide varying levels of functionality, security and privacy depending on use, needs, or company policy.

Proxy servers act as a firewall and web filter, provide shared network connections, and cache data to speed up common requests. Proxy servers can provide a high level of privacy.

Types of Proxy Servers:

i) **Forward Proxies** → A forward proxy server sits between the client and an external network. It evaluates the outbound requests and takes action on them before relaying that request to the external resource.

Most proxy services that we are likely to encounter are forward proxies. Virtual Private Networks (VPN) and Web content filters are both examples of forward proxies.

ii) **Open Proxies** → An open proxy is a proxy server that is accessible by an Internet user. Open proxies helps the clients to hide their IP address while browsing the web. Following are some of the open proxies:

http://www.stayinvisible.com/
http://www.multiproxy.org/
http://www.openproxies.com/

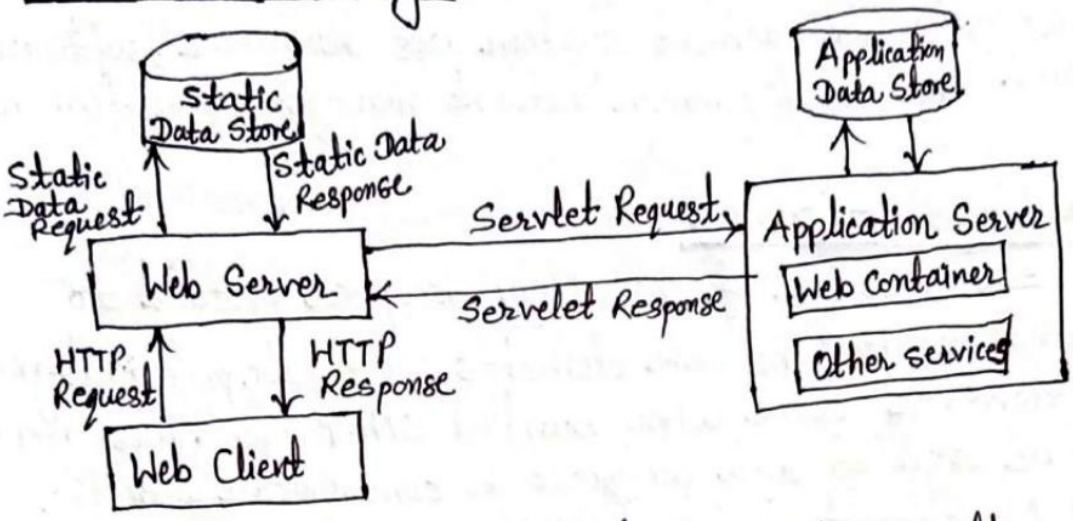
iii) **Reverse Proxies** → A reverse proxy server sits between a network and multiple other internal resources. A large website might have dozens of servers that collectively serve requests from a single domain. To accomplish that, client requests would resolve to a machine that would act as a load balancer. The load balancer would then proxy that traffic back to the individual servers. Varnish and Squid are some popular open source proxies.

⊛. Why should we use a proxy server?

→ To control internet usage of employees and children.
→ For bandwidth savings and improved speeds.
→ For Privacy benfits.
→ For Improved security.
→ To get access to blocked resources.

2). Web Application Servers: Web server is a computer where the web content is stored. Basically web server is used to host the websites but there exists other web servers also such as gaming, storage, FTP, email etc. Website is collection of web pages while web server is a software that respond to the request for web resources.

Web Server Working:



→ When client sends request for a web page, the web server search for the requested page. If requested page is found then it will send it to client with an HTTP response.

→ If the requested web page is not found, web server will send an HTTP response: Error 404 Not found.

→ If client has requested for some other resources then the web server will contact to the application server and data store to construct the HTTP response.

3) Mail Application Server:

A mail server also known as a mail transfer agent or MTA is an application that receives incoming e-mail from local users and forwards outgoing e-mail for delivery. A computer dedicated to running such applications is called a mail server. Microsoft Exchange, Sendmail, gmail etc. are some common mail server programs.

Mail servers can be broken down into two main categories; outgoing mail servers and incoming mail servers. Outgoing mail servers are known as SMTP servers. Incomming mail servers come in two main varieties. POP3 servers are best known for storing sent and received messages. IMAP servers always store copies of messages on servers.

⊛. Network Management:

We define network management as monitoring, testing, configuring, and troubleshooting network components to meet a set of requirements defined by an organization. These requirements include the smooth, efficient operation of the network that provides the predefined quality of services for users. To accomplish this task, a network management system uses hardware, software and humans. The most common network management system is SNMP.

Network Management: SNMP

SNMP is one of the widely accepted protocol to manage and monitor network elements. Most of the professional-grade network elements come with bundled SNMP agent. These agents have to be enabled and configured to communicate with the Network Management System (NMS).

SNMP is a set of protocols for network management and monitoring. These protocols are supported by many typical network devices such as routers, hubs, bridges, switches, servers, workstations, printers etc. SNMP is simple and powerful and has the ability to manage network by providing read/write abilities, collecting information on how much bandwidth is being used, Emailing an alert when computer server is low on disk space and many more.

SNMP Components and their functionalities:

1) SNMP Manager → A manager is a seperate unit that's accountable to communicate with the SNMP agent on network devices. This is usually a software installed on a PC or a server to operate one or more network management systems.

Functions:
- → Queries agents
- → Gets responses from agents
- → Locates variables in agents
- → Accept no synchronous events from agents.

i) **Managed Devices** → A managed device is a part of the network that needs some type of monitoring and management. For example, switches, routers, servers, printers etc.

ii) **SNMP Agent** → The agent is an application packaged inside the network element. The agent gathers the management information database from the device locally and makes it accessible to the SNMP manager, when its asked for. These agents could be specific or standard to a manufacturer.

Functions:
- → Collects management information about its local environment.
- → Stores and retrives management information.
- → Signals an event to the manager.
- → Acts as proxy for some non-SNMP managable network node.

iv) **Management Information Base (MIB):** MIB files are the set of parameters that an SNMP Manager can request the agent. Agent assembles these data locally and stores them as described in the Management Information Base. MIB contains standard set of statistical and control values defined for hardware nodes on a network. The SNMP Manager should be aware of these standard and private questions for every type of agent.