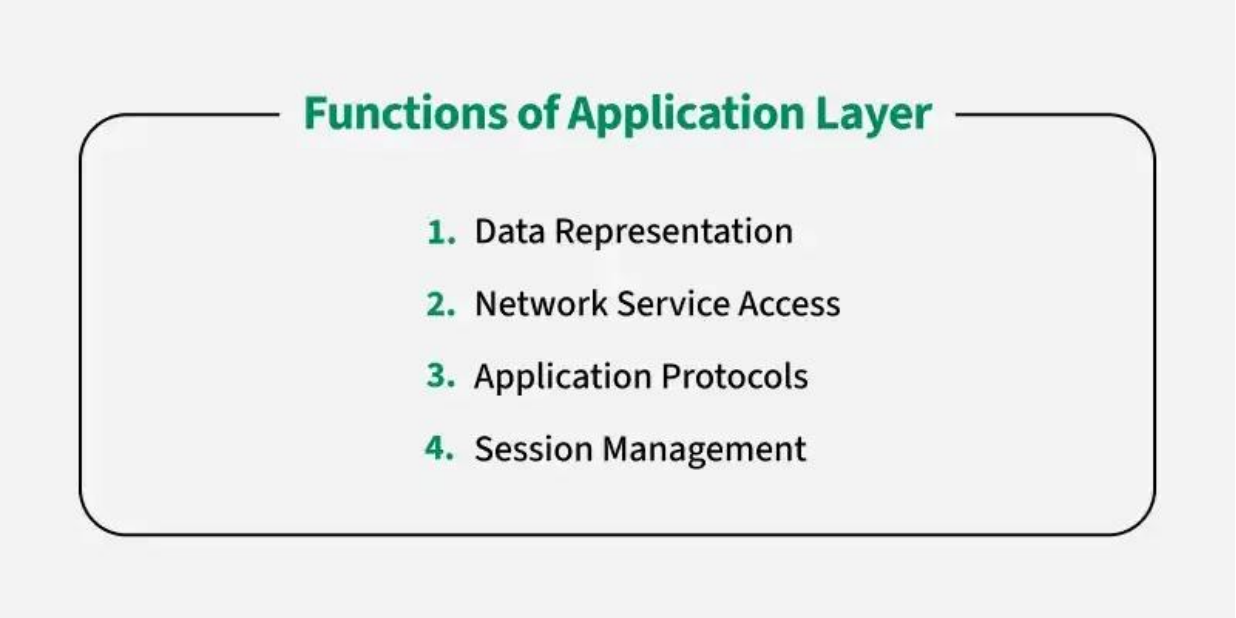
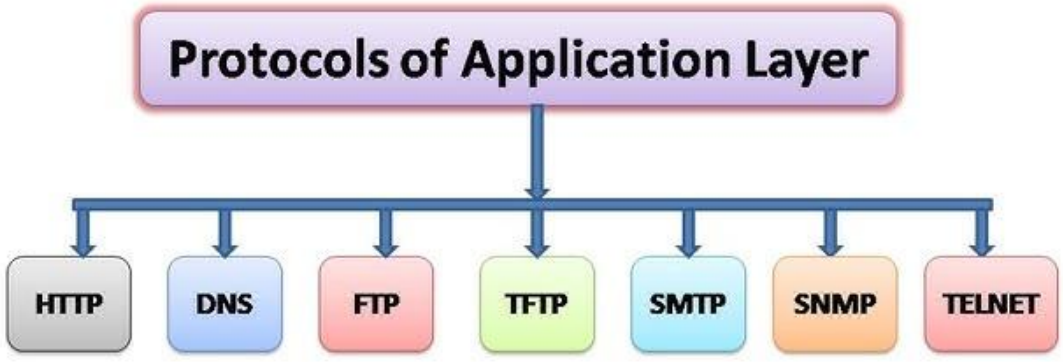
Application Layer



The Application Layer is the topmost layer in the Open System Interconnection (OSI) model. This layer provides several ways for manipulating the data which enables any type of user to access the network with ease. The Application Layer interface directly interacts with the application and provides common web application services. The application layer performs several kinds of functions that are required in any kind of application or communication process. In this article, we will discuss various application layer protocols.

What are Application Layer Protocols?



Application layer protocols are those protocols utilized at the application layer of the OSI (Open Systems Interconnection) and TCP/IP models. They facilitate communication and data sharing between software applications on various network devices. These protocols define the rules and standards that allow applications to interact and communicate quickly and effectively over a network.

Application Layer Protocol in Computer Network

TELNET

Telnet stands for the **TEL**etype **NET**work. It helps in terminal emulation. It allows Telnet clients to access the resources of the Telnet server. It is used for managing files on the Internet. It is used for the initial setup of devices like switches. The telnet command is a command that uses the Telnet protocol to communicate with a remote device or system. The port number of the telnet is 23.

Command

```
telnet [\\RemoteServer]
\\RemoteServer
: Specifies the name of the server
to which you want to connect
```

FTP

FTP stands for File Transfer Protocol. It is the protocol that actually lets us transfer files. It can facilitate this between any two machines using it. But FTP is not just a protocol but it is also a program. FTP promotes sharing of files via remote computers with reliable and efficient data transfer. The Port number for FTP is 20 for data and 21 for control.

Key aspects of FTP:

- **Client-Server Architecture:**

  FTP operates using a client-server model, where a client (user) connects to a server to transfer files.

- **Separate Control and Data Connections:**

  FTP uses two separate connections: one for control (commands) and one for data transfer.

- **Ubiquitous Usage:**

  FTP is widely used for various applications, including website file management, software distribution, and sharing files over the internet.

- **Not Secure:**

  FTP, in its original form, is not inherently secure, as it transmits data in plain text, making it vulnerable to eavesdropping and attacks. Alternatives like SFTP and FTPS offer better security.

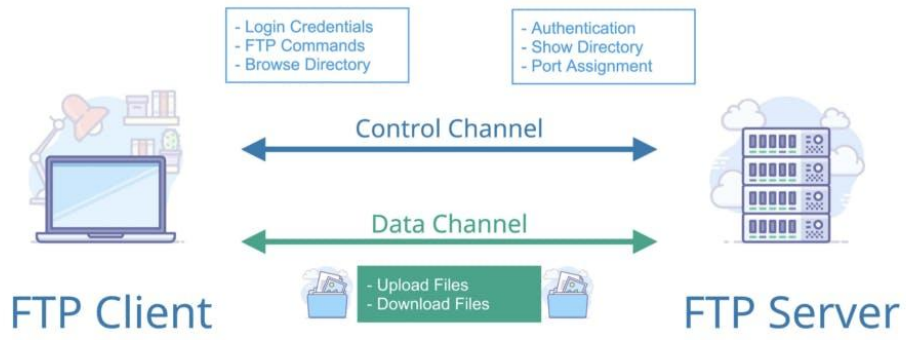
- **TCP/IP based:**
  FTP operates within the TCP/IP suite, making it a fundamental part of the internet infrastructure.

In essence, FTP provides the mechanisms for:

- **Uploading files:** Sending files from a client to a server.
- **Downloading files:** Retrieving files from a server to a client.
- **Managing files:** Listing directories, deleting files, and other file-related operations on a remote server.

Command

```
ftp machinename
```



NFS

It stands for a Network File System. It allows remote hosts to mount file systems over a network and interact with those file systems as though they are mounted locally. This enables system administrators to consolidate resources onto centralized servers on the network. The Port number for NFS is 2049.

Command

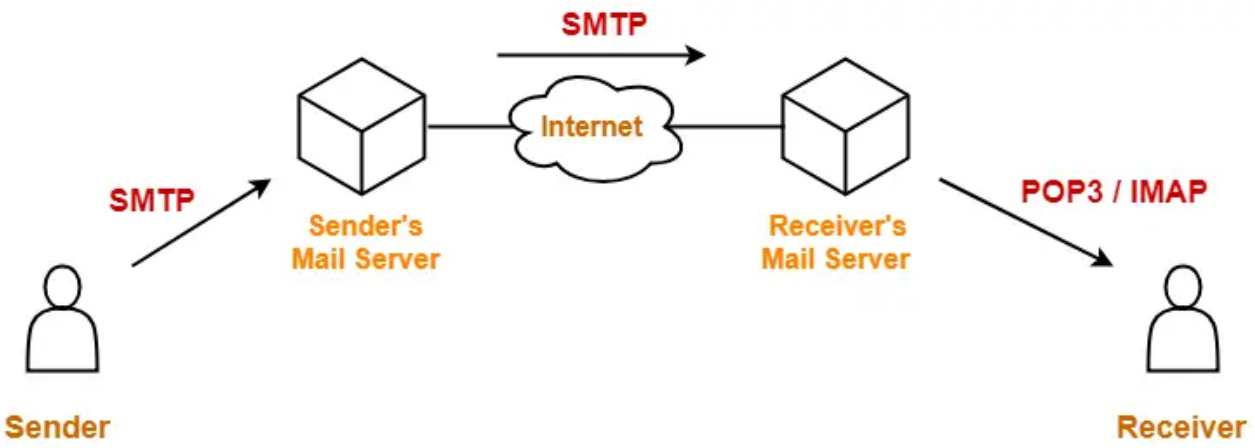
```
service nfs start
```

SMTP

It stands for Simple Mail Transfer Protocol. It is a part of the TCP/IP protocol. Using a process called "store and forward," SMTP moves your email on and across networks. It works closely with something called the Mail Transfer Agent (MTA) to send your communication to the right computer and email inbox. The Port number for SMTP is 25.

Working-

- SMTP server is always on a listening mode.
- Client initiates a TCP connection with the SMTP server.
- SMTP server listens for a connection and initiates a connection on that port.
- The connection is established.
- Client informs the SMTP server that it would like to send a mail.
- Assuming the server is OK, client sends the mail to its mail server.
- Client's mail server use DNS to get the IP Address of receiver's mail server.
- Then, SMTP transfers the mail from sender's mail server to the receiver's mail server.



While sending the mail, SMTP is used two times-

1. Between the sender and the sender's mail server
2. Between the sender's mail server and the receiver's mail server

NOTE-

To receive or download the email,

- Another protocol is needed between the receiver's mail server and the receiver.
- The most commonly used protocols are POP3 and IMAP.

Characteristics of SMTP-

- SMTP is a push protocol.
- SMTP uses TCP at the transport layer.
- SMTP uses port number 25.
- SMTP uses persistent TCP connections, so it can send multiple emails at once.
- SMTP is a connection oriented protocol.
- SMTP is an in-band protocol.
- SMTP is a stateless protocol.

Important Points-

Note-01:

SMTP is a pure text based protocol.

- SMTP can only handle the messages containing 7 bit ASCII text.
- SMTP can not transfer other types of data like images, video, audio etc.
- SMTP can not transfer executable files and binary objects.
- SMTP can not transfer the text data of other languages like French, Japanese, Chinese etc.

(since they are represented in 8 bit codes)

Note-02:

MIME extends the limited capabilities of email.

As the name suggests,

- Multipurpose Internet Email Extension (MIME) is an extension to the internet email protocol.
- It extends the limited capabilities of email by enabling the users to send and receive graphics, audio files, video files etc in the message.
- MIME was specially designed for SMTP.

Note-03:

SMTP is a stateless protocol.

This is because-

- It does not maintain the state of its clients.
- If an email is asked to be sent twice, then SMTP server resends it without saying that the email has already been sent.

Note-04:

We can not use SMTP at the receiver's side.

This is because-

- SMTP is a push protocol.
- At receiver's side, a pull protocol like POP3, IMAP is needed.
- Receiver periodically checks if he has any mail from his mail server.

Note-05:

Sender and receiver can not run SMTP between their machines.

This is because-

- Machines can not always be ON.
- So, the functionality has been divided between the client and the mail server.
- The mail server receives the mail on behalf of its client and manages the mail box of the client.

Note-06:

SMTP is not suitable for client authentication.

This is because-

- SMTP does not require authentication.
- It allows anyone on the Internet to send emails to anyone or even to a large group of people.
- SMTP Auth short for SMTP Authentication has been provided for authentication.

Command

```
MAIL FROM:<mail@abc.com?
```

SNMP

It stands for Simple Network Management Protocol. It gathers data by polling the devices on the network from a management station at fixed or random intervals, requiring them to disclose certain information. It is a way that servers can share information about their current state, and also a channel through which an administrate can modify pre-defined values. The Port number of SNMP is 161(TCP) and 162(UDP).

Command

```
snmpget -mALL -v1 -cpublic snmp_agent_Ip_address sysName.0
```

DNS

It stands for Domain Name System. Every time you use a domain name, therefore, a DNS service must translate the name into the corresponding IP address. For example, the domain name www.abc.com might translate to 198.105.232.4.
The Port number for DNS is 53.

Command

```
ipconfig /flushdns
```

Purpose-

- DNS is a host name to IP Address translation service.
- It converts the names we type in our web browser address bar to the IP Address of web servers hosting those sites.

The need for Domain Name Service arises due to the following reasons-

- IP Addresses are not static and may change dynamically.
- So, a mapping is required which maps the domain names to the IP Addresses of their web servers.
- IP Addresses are a complex series of numbers.
- So, it is difficult to remember IP Addresses directly while it is easy to remember names.

Important Notes-

- DNS uses UDP (port 53) at the transport layer.
- DNS uses UDP at the transport layer due to the following reasons-
    - UDP is much faster than TCP.
    - TCP is slow as it uses Three-way handshake to start the data transfer.
    - DNS requests are very small.
    - So, they fits well within UDP segments.
    - Although UDP is not reliable but reliability can be added on application layer.
    - Reliability can be added by using timeouts and resend at the application layer.
    - Thus, in the end both speed and protection are achieved.
- DNS is a connection less protocol.
    - DNS uses UDP at the transport layer for replying to the DNS queries of clients.
    - Therefore, it is a connection less protocol.
- DNS is non-persistent.
- DNS is a stateless protocol.
    - DNS server accepts the requests, process them, resolves the query and forget about them.
    - It does not make any assumption how long this will be.

- Mapping an IP Address onto a domain name is referred to as Inverse domain.
    - DNS can translate a domain name onto an IP Address.
    - Also, it can translate an IP Address onto a domain name.
    - For the first time,
    - There is more delay in translating the domain name onto an IP Address.
    - Converting a domain name onto an IP Address is an extra overhead.
    - This overhead is called as DNS Overhead.
    - It causes an unnecessary delay in serving the request.
    - So, there is more delay for the first time.
    - To reduce the delay next time, IP Addresses are stored in the computer using log.
    - This avoids the DNS overhead next time and takes less time in serving the request.
    - When it gets expired, the request is again served through DNS.

DHCP

It stands for Dynamic Host Configuration Protocol (DHCP). It gives IP addresses to hosts. There is a lot of information a DHCP server can provide to a host when the host is registering for an IP address with the DHCP server. Port number for DHCP is 67, 68.

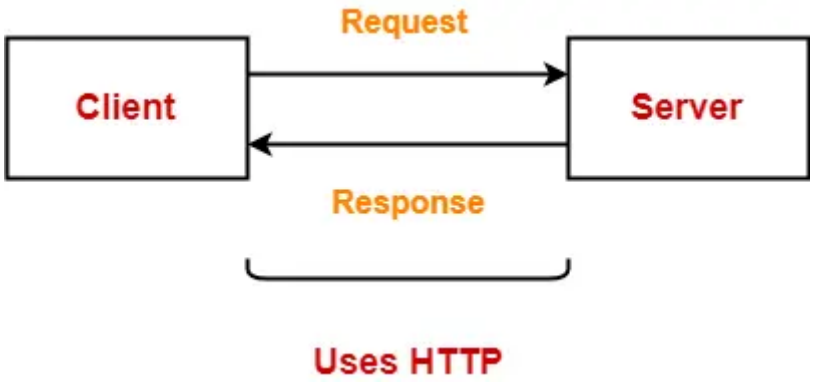
Command

```
clear ip dhcp binding {address | * }
```

HTTP/HTTPS

HTTP stands for Hypertext Transfer Protocol and HTTPS is the more secured version of HTTP, that's why HTTPS stands for Hypertext Transfer Protocol Secure. This protocol is used to access data from the World Wide Web. The Hypertext is the well-organized documentation system that is used to link pages in the text document.

- HTTP is based on the client-server model.
- It uses TCP for establishing connections.
- HTTP is a stateless protocol, which means the server doesn't maintain any information about the previous request from the client.
- HTTP uses port number 80 for establishing the connection.



Purpose-

- It is mainly used for the retrieval of data from websites throughout the internet.
- It works on the top of TCP/IP suite of protocols.

Working-

HTTP uses a client-server model where-

- Web browser is the client.
- Client communicates with the web server hosting the website.

Whenever a client requests some information (say clicks on a hyperlink) to the website server.
The browser sends a request message to the HTTP server for the requested objects.
Then-

- HTTP opens a connection between the client and server through **TCP**.
- HTTP sends a request to the server which collects the requested data.
- HTTP sends the response with the objects back to the client.
- HTTP closes the connection.

HTTP Connections-

HTTP connections can be of two types-

1. Non-persistent HTTP connection
2. Persistent HTTP connection

Non-persistent HTTP connection	Persistent HTTP connection
Non-persistent HTTP connection is one that is used for serving exactly one request and sending one response.	Persistent HTTP connection is one that can be used for serving multiple requests.
HTTP server closes the TCP connection automatically after sending a HTTP response.	HTTP server closes the TCP connection only when it is not used for a certain configurable amount of time.
A new separate TCP connection is used for each object.	A single TCP connection is used for sending multiple objects one after the other.
HTTP 1.0 supports non-persistent connections by default.	HTTP 1.1 supports persistent connections by default.
Example- Suppose a request has been made for a HTML page that contains 10 images (called objects). Then, With non-persistent connection, all the 11 objects (1 page + 10 images) will be sent one by one. For getting each object, a new separate connection will be opened and used.	**Example-** Suppose a request has been made for a HTML page that contains 10 images (called objects). Then, With persistent connection, all the 11 objects (1 page + 10 images) will be sent one after the other using a single TCP connection.

Important Notes-

Note-01:

HTTP uses TCP at the transport layer.

This is because-

- Unlike **UDP**, it guarantees the delivery of data via a **Three-way handshake**.
- It ensures the re transmission of lost packets.
- HTTP does not have any inbuilt facility for providing reliability.
- So, if HTTP uses UDP, then it will have to maintain or handle the session on its own.
- For example- If a packet gets lost, then HTTP will have to re-transmit the packet.

Note-02:

It is important to know-

- Any service which does not use TCP should have the inbuilt facility for providing reliability.

Note-03:

HTTP uses port number 80.

- HTTP clients uses port 80 to send and receive requested web pages from a HTTP server.
- Similarly, HTTP server responds to all the requests at port 80.

Note-04:

HTTP 1.0 is non-persistent and HTTP 1.1 is persistent.

- Already discussed in the above table.
- Persistent connections improve the performance by 20%.

Note-05:

HTTP 1.0 is a connectionless protocol.

This is because-

- After serving the single HTTP request, the connection is closed and it is not used again.
- So, HTTP 1.0 without connection keep alive is connectionless.

Note-06:

HTTP is an in-band protocol.

This is because-

- HTTP passes the control data (commands) and main data over the same connection.
- Both control data and main data are processed in the same way without any distinction.
- No high priority is given to the control data (commands).

Note-07:

HTTP is a stateless protocol.

This is because-

- HTTP server does not maintain any state.
- It forgets about the client after sending the response.
- It treats every new request independently.
- HTTP closes the connection automatically after generating the response for each request.
- This ensures that no client can engage connection with web server for a long time.

POP

POP stands for Post Office Protocol and the latest version is known as POP3 (Post Office Protocol version 3). This is a simple protocol used by User agents for message retrieval from mail servers.

- POP protocol work with Port number 110.
- It uses TCP for establishing connections.

POP works in dual mode- *Delete mode, Keep Mode*.
In Delete mode, it deletes the message from the mail server once they are downloaded to the local system.
In Keep mode, it doesn't delete the message from the mail server and also facilitates the users to access the mails later from the mail server.

MIME

MIME stands for Multipurpose Internet Mail Extension. This protocol is designed to extend the capabilities of the existing Internet email protocol like SMTP. MIME allows non-ASCII data to be sent via SMTP. It allows users to send/receive various kinds of files over the Internet like audio, video, programs, etc. MIME is not a standalone protocol it works in collaboration with other protocols to extend their capabilities.

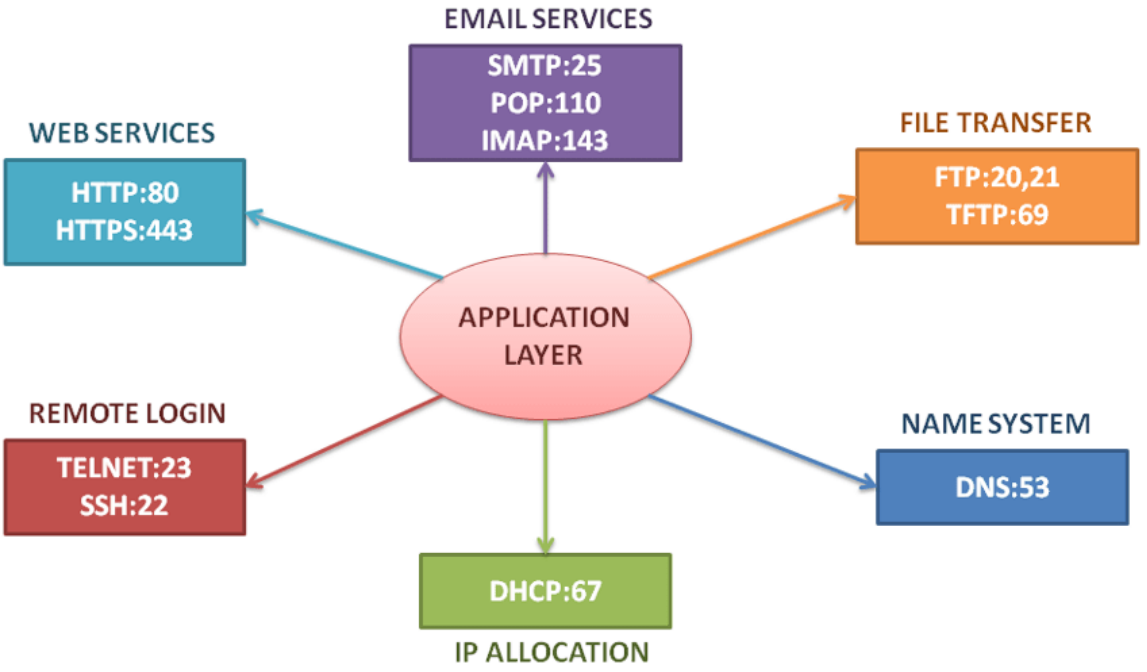
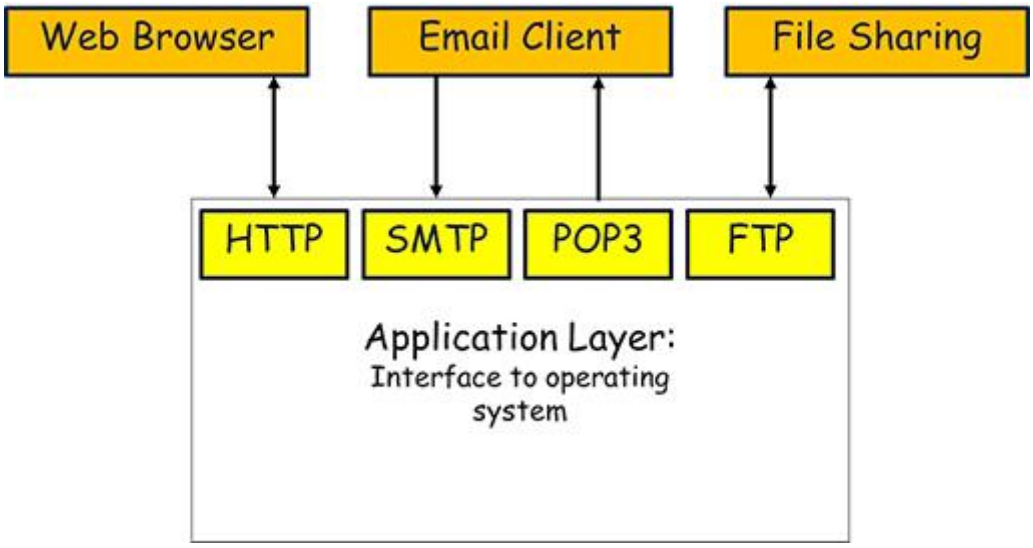


Figure 2: Application Layer Protocols



Comparison of Application Layer Protocols

	DNS	HTTP	SMTP	POP	FTP
Stateful / Stateless	Stateless	Stateless	Stateless	Stateful	Stateful
Transport Protocol Used	UDP	TCP	TCP	TCP	TCP
Connectionless / Connection Oriented	Connectionless	Connectionless	Connection Oriented	Connection Oriented	Connection Oriented
Persistent / Non-persistent	Non-persistent	HTTP 1.0 is non-persistent. HTTP 1.1 is persistent.	Persistent	Persistent	Control connection is persistent. Data connection is non-persistent.
Port Number Used	53	80	25	110	20 for data connection. 21 for control connection.
In band / Out-of-band	In band	In band	In band	In band	Out-of-band