# Chapter One

# Introduction to Internet and the Web

## Internet history, Uses and Services

**What is Internet?**

The Internet is a worldwide system of interconnected computer networks. The computers and computer networks exchange information using TCP/IP (Transmission Control Protocol/Internet Protocol) to communicate with each other. The computers are connected via the telecommunications networks, and the Internet can be used for e-mailing, transferring files and accessing information on the World Wide Web.

**Internet History**

The rapid development of Internet started at the early 1960, paralleled with the developments of the computers. Those scientists and researcher started to realize a great vision, a future that everyone will be able to communicate by using their computers.

J.C.R. Licklider of MIT, first proposed a global network of computers in 1962 and followed by Leonard Kleinrock from MIT, who published the first paper on packet switching theory.

ARPANET (Advanced Research Projects Agency NETwork ), which is the former of Internet, was a project launched by Military Department of USA. It was brought online at Oct 29, 1969 by Charley Kline at UCLA, when he attempted to perform a remote login from UCLA to SRI. In order to get attentions from public, they made the first public demonstration of ARPANET at an international conference at October 1972.

The initial ARPANET was a single, closed network. In order to communicate with an ARPANET, one had to be attached to another ARPANET IMP (interface message processor). Hence, the disadvantages of single network were realized and lead the development of open-architecture network and also different protocols of internetworking, which enable multiple networks, can be joined together. E-mail was adapted for ARPANET by Ray Tomlinson of BBN in 1972. The telnet protocol, enabling logging on to a remote computer, was published as a Request for Comments (RFC) in 1972. RFC's are a means of sharing developmental work throughout community. The FTP protocol, enabling file transfers between Internet sites, was published as an RFC in 1973, and from then on RFC's were available electronically to anyone who had use of the FTP protocol.

Before the TCP/IP protocol was introduced by BoB Kahn, the networking protocols used for the ARPANET was NCP, Network Control Protocols. NCP did not have the ability to address networks further downstream than a destination IMP on ARPANET. By 1980, the Internet had reached a certain level of maturity and started to exposed to public usage more and more often. At the same time, French launched the Minitel project to bring data networking into everyone’s home by gave away a free terminal to each household requested.

At the 1990s, the Internet predecessor, ARPANET finally came to an end, and replaced by the NSFNET (National Science Foundation) which serve as a backbone connecting regional networks in USA. However, the most significant changes of Internet at 1990s was the World Wide Web (WWW) application which truly brought Internet to our daily life. Various technologies such as VoIP, HTML, web browsers with graphical user interfaces, P2P file sharing, and instant messaging which is very familiar to us nowadays.

**Advantages**

Internet covers almost every aspect of life, one can think of. Here, we will discuss some of the advantages of Internet:

* Social Networking
* Education and Technology
* Entertainment
* Online Services

Internet allows us to communicate with the people sitting at remote locations. There are various apps available on the wed that uses Internet as a medium for communication. One can find various social networking sites such as: Facebook, Twitter, Yahoo, Google+, Flickr, Orkut etc

One can surf for any kind of information over the internet. Information regarding various topics such as Technology, Health & Science, Social Studies, Geographical Information, Information Technology, Products etc can be surfed with help of a search engine.

Apart from communication and source of information, internet also serves a medium for entertainment. Following are the various modes for entertainment over internet. These are Online Television, Online Games, Songs, Videos and Social Networking Apps.

Internet allows us to use many services like Internet Banking, Matrimonial Services, Online Shopping, Online Ticket Booking, Online Bill Payment, Data Sharing & E-mail.

Internet provides concept of **electronic commerce** that allows the business deals to be conducted on electronic systems

**Disadvantages**

However, Internet has proved to be a powerful source of information in almost every field, yet there exists many disadvantages discussed below:

* **Threat to Personal Information**: There are always chances to lose personal information such as name, address, credit card number. Therefore, one should be very careful while sharing such information. One should use credit cards only through authenticated sites.
* **Spamming:** it corresponds to the unwanted e-mails in bulk. These e-mails serve no purpose and lead to obstruction of entire system.
* **Virus attack**: Viruses can easily be spread to the computers connected to internet. Such virus attacks may cause your system to crash or your important data may get deleted.
* Also a biggest threat on internet is pornography. There are many pornographic (sexual sit) sites that can be found, letting your children to use internet which indirectly affects the children healthy mental life.
* **Cyber Crime**

### The World Wide Web (WWW)

WWW is also known as W3. It offers a way to access documents spread over the several servers over the internet. These documents may contain texts, graphics, audio, video, hyperlinks. The hyperlinks allow the users to navigate between the documents.

The World Wide Web is a system of **Internet servers** that use HTTP (Hypertext Transfer Protocol) to transfer documents formatted in HTML (Hypertext Mark-up Language). These are viewed by using software for web browsers such as Netscape, Safari, Google Chrome and Internet Explorer. Hypertext enables a document to be connected to other documents on the web through hyperlinks. It is possible to move from one document to another by using hyperlinked text found within web pages.

### Browser and Web Server

**Web Browser** is application software that allows us to view and explore information on the web. User can request for any web page by just entering a URL into address bar.

Web browser can show text, audio, video, animation and more. It is the responsibility of a web browser to interpret text and commands contained in the web page.

Earlier the web browsers were text-based while now days graphical-based or voice-based web browsers are also available. Following are the most common web browser available today:

|  |  |
| --- | --- |
| **Browser** | **Vendor** |
| Internet Explorer | Microsoft |
| Google Chrome | Google |
| Mozilla Firefox | Mozilla |
| Netscape Navigator | Netscape Communications Corp. |
| Opera | Opera Software |
| Safari | Apple |
| Sea Monkey | Mozilla Foundation |
| K-meleon | K-meleon |

**Web Server**

**Web server** is a computer where the web content is stored. Basically web server is used to host the web sites but there exists other web servers also such as gaming, storage, FTP, email etc.

**Web site**

is collection of web pages while web server is software that responds to the request for web resources.

**Web Server Working**

Web server respond to the client request in either of the following two ways:

* Sending the file to the client associated with the requested URL.
* Generating response by invoking a script and communicating with database

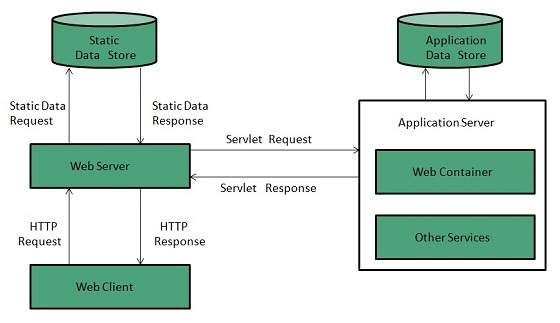


Figure 1.1.2 Client Server Architecture

### Document Types and MIME

A **media type** (formerly known as **MIME type**) is a two-part identifier for file formats (formatted file) and format contents transmitted on the Internet. The Internet Assigned Numbers Authority (IANA) is the official authority for the standardization and publication of these classifications. Media types were originally defined in Request for Comments 2045 in November 1996 as a part of *MIME (Multipurpose Internet Mail Extensions)* specification, for denoting type of email message content and attachments; hence the name *MIME type*. Media types are also used by other internet protocols such as HTTP and document file formats such as HTML, for similar purpose.

A media type consists of a *type* and a *subtype*, which is further structured into a *tree*. A media type can optionally define a *suffix* and *parameters*:

type "/" [tree "."] subtype ["+" suffix] \*[";" parameter]

The currently registered types are: application, audio, example, font, image, message, model, multipart, text and video.

As an example, an HTML file might be designated text/html; charset=UTF-8. In this example, text is the type, html is the subtype, and charset=UTF-8 is an optional parameter indicating the character encoding.

A subtype typically consists of a media format, but it may or must also contain other content, such as a tree prefix, producer, product or suffix, according to the different rules in registration trees.

Types, subtypes, and parameter names are case-insensitive. Parameter values are usually case-sensitive, but may be interpreted in a case-insensitive fashion depending on the intended use.

Common examples

* application/[javascript](https://en.wikipedia.org/wiki/Javascript)
* application/[xml](https://en.wikipedia.org/wiki/Xml)
* application/[zip](https://en.wikipedia.org/wiki/Zip_%28file_format%29)
* application/[pdf](https://en.wikipedia.org/wiki/Pdf)
* application/sql
* application/msword (.doc)
* application/vnd.openxmlformats-officedocument.wordprocessingml.document(.docx)
* application/vnd.ms-excel (.xls)
* application/vnd.openxmlformats-officedocument.spreadsheetml.sheet (.xlsx)
* application/vnd.ms-powerpoint (.ppt)
* application/vnd.openxmlformats-officedocument.presentationml.presentation (.pptx)
* application/vnd.oasis.opendocument.text (.odt)
* audio/[mpeg](https://en.wikipedia.org/wiki/Mpeg)
* audio/[ogg](https://en.wikipedia.org/wiki/Ogg)
* multipart/form-data
* text/[css](https://en.wikipedia.org/wiki/Css)
* text/[html](https://en.wikipedia.org/wiki/Html)
* text/[xml](https://en.wikipedia.org/wiki/Xml)
* text/[csv](https://en.wikipedia.org/wiki/Comma-separated_values)
* text/plain
* image/[png](https://en.wikipedia.org/wiki/Portable_Network_Graphics)
* image/[jpeg](https://en.wikipedia.org/wiki/Jpeg)
* image/[gif](https://en.wikipedia.org/wiki/Gif)

### Uniform Resource Locators (URLs)

**Uniform Resource Locator (URL)** refers to a web address which uniquely identifies a document over the internet. This document can be a web page, image, audio, video or anything else present on the web. For example: www.dmu.edu.et**/index.html**.

URL Types

There are two forms of URL as listed:

* Absolute URL
* Relative URL

**Absolute URL**

Absolute URL is a complete address of a resource on the web. This completed address comprises of protocol used, server name, path name and file name.

For example http:// www.dmu.edu.et/index.htm. where:

* **http** is the protocol.
* **dmu.edu.et** is the server name(domain name).
* **index.htm** is the file name.

The protocol part tells the web browser how to handle the file. Similarly we have some other protocols also that can be used to create URL are: FTP, https, Gopher, mailto, news

**Relative URL**

Relative URL is a partial address of a webpage. Unlike absolute URL, the protocol and server part are omitted from relative URL. **Relative URLs are used for internal links** i.e. to create links to file that are part of same website as the WebPages on which you are placing the link. For example, to link an image on dmu.edu.et/bids/bids.pdf, we can use the relative URL which can take the form like **/ bids/bids.pdf**.

Difference between Absolute and Relative URL

|  |  |
| --- | --- |
| **Absolute URL** | **Relative URL** |
| * Used to link web pages on different websites * Difficult to manage. * Changes happen when the server name or directory name changes * Take time to access | * Used to link web pages within the same website. * Easy to Manage * Remains same even of we change the server name or directory name. * Comparatively faster to access. |

### Domain Name System (DNS)

DNS is an application layer protocol used to resolve hostnames to IP addresses. Although a host can be accessed by using only its IP address, DNS makes your life easier by using domain names. For example, you can access the Google website by typing http://208.117.229.214 in your browser, but it is much easier to type http://www.google.com.

Each host that wants to use DNS needs to have a DNS server configured. When you type a URL in your browser (e.g. http://www.google.com) , your host will query the DNS server for the IP address of www.google.com. The DNs server will resolve the query and send the answer back to the host. The host will then be able to establish a connection to http://www.google.com.

## Communication Protocols

In telecommunication, a **communication protocol** is a system of rules that allow two or more entities of a communications system to transmit information via any kind of variation of a physical quantity. The protocol defines the rules, syntax, semantics and synchronization of communication and possible error recovery methods. Protocols may be implemented by hardware, software, or a combination of both.

Communicating systems use well-defined formats for exchanging various messages. Each message has an exact meaning intended to elicit a response from a range of possible responses pre-determined for that particular situation. The specified behavior is typically independent of how it is to be implemented. Communication protocols have to be agreed upon by the parties involved. To reach agreement, a protocol may be developed into a technical standard. A programming language describes the same for computations, so there is a close analogy between protocols and programming languages: *protocols are to communication what programming languages are to computations*.

Multiple protocols often describe different aspects of a single communication. A group of protocols designed to work together are known as a protocol suite; when implemented in software they are a protocol stack.

### TCP/ IP, HTTP, FTP and Email Protocols

**TCP**

TCP is a reliable, guaranteed-delivery protocol. TCP specifies the methods hosts use to acknowledge the receipt of packets, and requires the source host to resend packets that are not acknowledged. TCP protocols also govern the exchange of messages between the source and destination hosts to create a communication session. TCP is often compared to a pipeline, or a persistent connection, between hosts. Because of this, TCP is referred to as a connection-oriented protocol.

TCP requires overhead to keep track of the individual conversations between source and destination hosts and to process acknowledgements and retransmissions. In some cases, the delays caused by this overhead cannot be tolerated by the application. These applications are better suited to UDP.

**UDP**

UDP is a very simple, connectionless protocol. It has the advantage of providing for low overhead data delivery. Because UDP is a "best effort" Transport Layer protocol, UDP datagrams may arrive at the destination out of order, or may even be lost all together. UDP does not provide guaranteed data delivery or flow control. Applications that use UDP can tolerate small amounts of missing data. An example of a UDP application is Internet radio. If a piece of data is not delivered, there may only be a minor effect on the quality of the broadcast.

TCP Vs UDP

UDP is a very simple protocol. Because it is not connection-oriented and does not provide the sophisticated retransmission, sequencing, and flow control mechanisms of TCP, UDP has a much lower overhead.

UDP is often referred to as an unreliable delivery protocol; because there is no guarantee that a message has been received by the destination host. This does not mean that applications that use UDP are unreliable. It simply means that these functions are not provided by the Transport Layer protocol and must be implemented elsewhere if required.

Although the total amount of UDP traffic found on a typical network is often relatively low, key Application Layer protocols that use UDP include:

* Domain Name System (DNS)
* Simple Network Management Protocol (SNMP)
* Dynamic Host Configuration Protocol (DHCP)
* Routing Information Protocol (RIP)
* Trivial File Transfer Protocol (TFTP)
* Online games

**HTTP and HTTPs**

The Hypertext Transfer Protocol (HTTP), one of the protocols in the TCP/IP suite, was originally developed to enable the retrieval of HTML formatted web pages. It is now used for distributed, collaborative information sharing. The HTTP protocol has evolved through multiple versions. The version currently used by most ISPs to provide web-hosting services is HTTP version 1.1. Unlike earlier versions, this version enables a single web server to host multiple web sites. It also permits persistent connections, so that multiple request and response messages can use the same connection, reducing the time it takes to initiate new TCP sessions.

HTTP specifies a request/response protocol. When a client, typically a web browser, sends a request message to a server, the HTTP protocol defines the message types the client uses to request the web page. The HTTP protocol also defines the message types the server uses to respond.

Although it is remarkably flexible, HTTP is not a secure protocol. The request messages send information to the server in plain text that can be intercepted and read. Similarly, the server responses, typically HTML pages, are also sent unencrypted.

Some HTTP cmmands (your browser sends these internally)

GET resource -- requests data from a specified resource

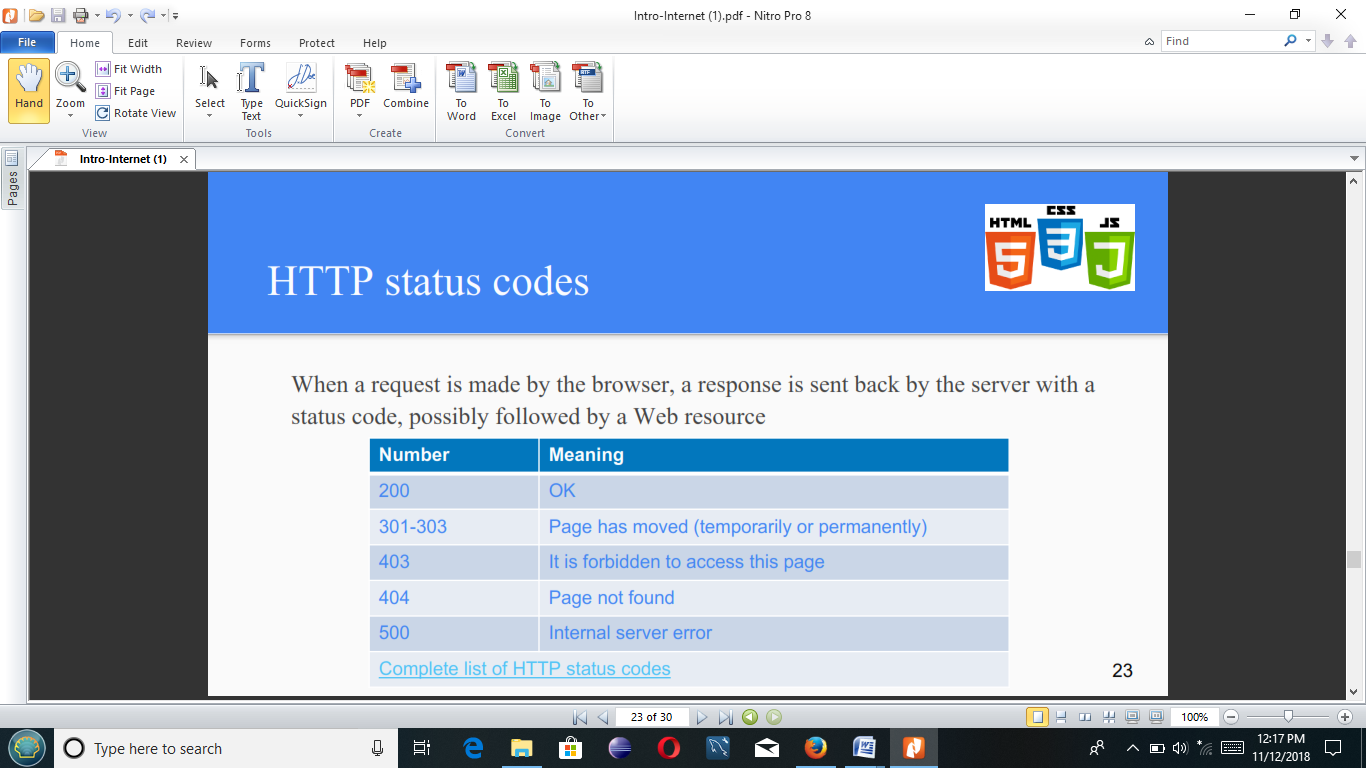
POST resource -- submits data to be processed to a specified resource

PUT resource -- uploads a representation of the specified URL

DELETE resource -- deletes the specified resource `

HTTP Status Codes

When a request is made by the browser, a response is sent back by the server with a status code, possibly followed by a Web resource.



For secure communication across the Internet, the Secure HTTP (HTTPS) protocol is used for accessing or posting web server information. HTTPS can use authentication and encryption to secure data as it travels between the client and server. HTTPS specifies additional rules for passing data between the Application Layer and the Transport Layer.

When contacting an HTTP server to download a web page, a uniform resource locator (URL) is used to locate the server and a specific resource.

The URL identifies:

* Protocol being used
* Domain name of the server needing to be accessed
* Location of the resource on the server, such as http://example.com/example1/index.htm

Many web server applications are available that allow for short URLs. Short URLs are popular because they are easier to write down, remember, or share. With a short URL, a default resource page is assumed when a specific URL is typed. When a user types in a shortened URL, like http://example.com, the default page that is sent to the user is actually the http://example.com/example1/index.htm web page.

To safeguard data, especially confidential information, some ISPs provide secure web services. To support secure web services ISPs use HTTPS (HTTP over secure sockets layer (SSL)). HTTPS uses the same client request-server response process as HTTP, but the data stream is encrypted with SSL before being transported across the network.

When the HTTP data stream arrives at the server, the TCP layer passes it up to SSL in the server's Application Layer, where it is decrypted.

The maximum number of simultaneous connections that a server can support for HTTPS is less than that for HTTP. HTTPS creates additional load and processing time on the server due to the encryption and decryption of traffic. To keep server performance up, HTTPS should only be used when necessary, such as when exchanging confidential information.

**File Transfer Protocol (FTP)**

FTP is a connection-oriented protocol that uses TCP to communicate between a client FTP process and an FTP process on a server. FTP implementations include the functions of a protocol interpreter (PI) and a data transfer process (DTP). PI and DTP define two separate processes that work together to transfer files. As a result, FTP requires two connections to exist between the client and server, one to send control information and commands, and a second one for the actual file data transfer.

**Protocol Interpreter (PI)**

The PI function is the main control connection between the FTP client and the FTP server. It establishes the TCP connection and passes control information to the server. Control information includes things such as commands to navigate through a file hierarchy, as well as renaming or moving files. The control connection, or control stream, stays open until closed by the user. When a user wants to connect to an FTP server:

1. The user-PI sends a connection request to the server-PI on well-known port 21.

2. The server-PI replies and the connection is established.

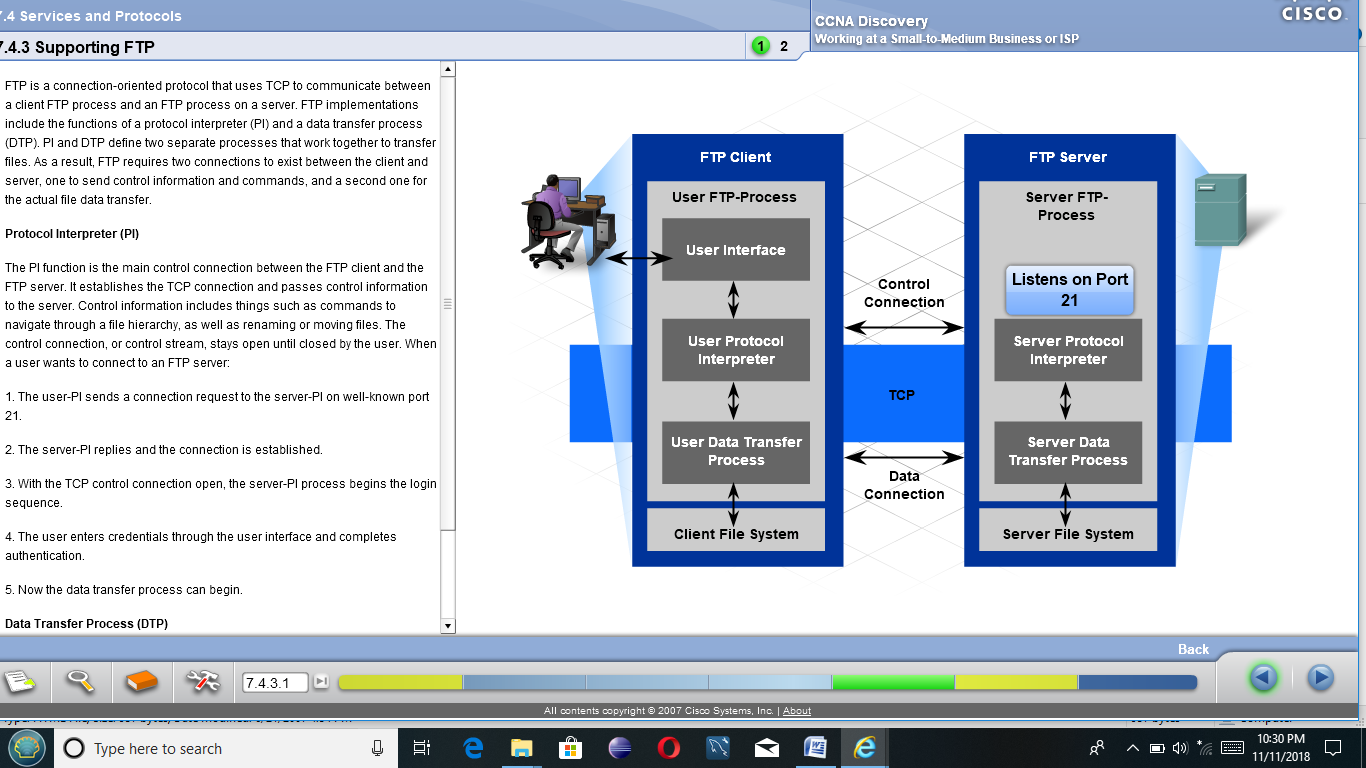
3. With the TCP control connection open, the server-PI process begins the login sequence.

4. The user enters credentials through the user interface and completes authentication.

5. Now the data transfer process can begin.

**Data Transfer Process (DTP)**

DTP is a separate data transfer function. This function is enabled only when the user wants to actually transfer files to or from the FTP server. Unlike the PI connection, which remains open, the DTP connection closes automatically when the file transfer is complete.



**Email Protocols**

One of the primary services offered by an ISP(internet service provider) is email hosting. Email is a store and forward method of sending, storing, and retrieving electronic messages across a network. Email messages are stored in databases on mail servers. ISPs often maintain mail servers that support many different customer accounts.

Email clients communicate with mail servers in order to send and receive email. Mail servers communicate with other mail servers to transport messages from one domain to another. In other words, an email client does not communicate directly with another email client when sending email. Both clients must rely upon the mail server for transport of the messages. This is true even when both users are in the same domain.

Email clients send messages to the email server configured in the application settings. When the server receives the message, it checks to see if the recipient domain is located on its local database. If it is not, it sends a DNS request to determine the mail server for the destination domain. Once the IP address of the destination mail server is known, the email is sent to the appropriate server.

Email supports three separate protocols for operation: SMTP, POP3, and IMAP4. The Application Layer process that sends mail, either from a client to a server or between servers, implements the SMTP protocol. A client retrieves email using one of two application layer protocols: POP3 or IMAP.

**Simple Mail Transfer Protocol (SMTP)**

The functions specified by the Simple Mail Transfer Protocol (SMTP) enable the transfer of mail reliably and efficiently.

For SMTP applications to do this, two conditions must be met:

* The mail message must be formatted properly
* SMTP processes must be running on both client and server

SMTP message formats require a message header and a message body. While the message body can contain any amount of text, the message header must have a properly formatted recipient email address and a sender address. Any other header information is optional.

When a client sends email, the client SMTP process connects with a server SMTP process on well-known port 25. Once the connection is made, the client attempts to send mail to the server across the connection. Once the server receives the message, it either places the message in a local account or forwards the message using the same SMTP connection process to another mail server.

The destination email server may not be online, or may be busy, when email messages are sent. Therefore, SMTP provides for the spooling of messages to be sent at a later time. Periodically, the server checks the queue for messages and attempts to send them again. After a predetermined expiration time, if the message is still undelivered, it will be returned to the sender as undeliverable.

One of the required fields in an email message header is the recipient email address. The structure of an email address includes the email account name or an alias, as well as the domain name of the mail server. An example of an email address:

recipient@cisco.com.

The @ symbol separates the account and the domain name of the server.

When a message is sent to recipient@cisco.com, the domain name is sent to the DNS server in order to obtain the IP address of the domain mail server. Mail servers are identified in DNS by an MX record indicator. When the destination mail server receives the message, it stores the message in the appropriate mailbox. The mailbox location is determined based on the account specified in the first part of the email address, in this case, the recipient account. The message will remain in the mailbox until the recipient connects to the server to retrieve the email.

If the mail server receives an email message that references an account that does not exist, the email is returned to the sender as undeliverable.

**Post Office Protocol (POP)**

The Post Office Protocol - Version 3 (POP3) is used to enable a workstation to retrieve mail from a mail server. With POP3, mail is downloaded from the server to the client and then deleted on the server.

The server starts the POP3 service by passively listening on TCP port 110 for client connection requests. When a client wishes to make use of the service, it sends a request to establish a TCP connection with the server. Once the connection is established, the POP3 server sends a greeting. The client and POP3 server then exchange commands and responses until the connection is closed or aborted.

Since email messages are downloaded to the client and removed from the server, this means that there is not a centralized location where email messages are kept. This makes the POP3 protocol undesirable in a centralized backup solution for a small business.

The POP3 protocol is desirable for an ISP since it alleviates the ISP's responsibility of managing large amounts of storage for their email servers.

**Internet Message Access Protocol (IMAP)**

Internet Message Access Protocol (IMAP4) is another protocol that describes a method to retrieve email messages. However, unlike POP3, when the user connects to an IMAP-capable server, copies of the messages are downloaded to the client application. The original messages are kept on the server until manually deleted. Users view copies of the messages in their email client software.

Users can create a file hierarchy on the server to organize and store mail. That file structure is duplicated on the email client as well. When a user decides to delete a message, the server synchronizes that action and deletes the message from the server.

For small to medium-sized businesses, there are many advantages to the IMAP protocol. IMAP can lead to long-term storage of email messages on mail servers and allow for centralized backup. It also enables employees to access email messages from multiple locations, using different devices or client software. The mailbox folder structure that a user is used to seeing is available for viewing regardless of how the user accesses the mailbox.

For an ISP, IMAP may not be the protocol of choice. It can be expensive to purchase and maintain the disk space to support the large number of stored emails. Additionally, if customers expect their mailboxes to be backed up routinely, that can further increase the costs to the ISP.