UNIT 5

[Application Layer- Standard Client-Server Protocols]

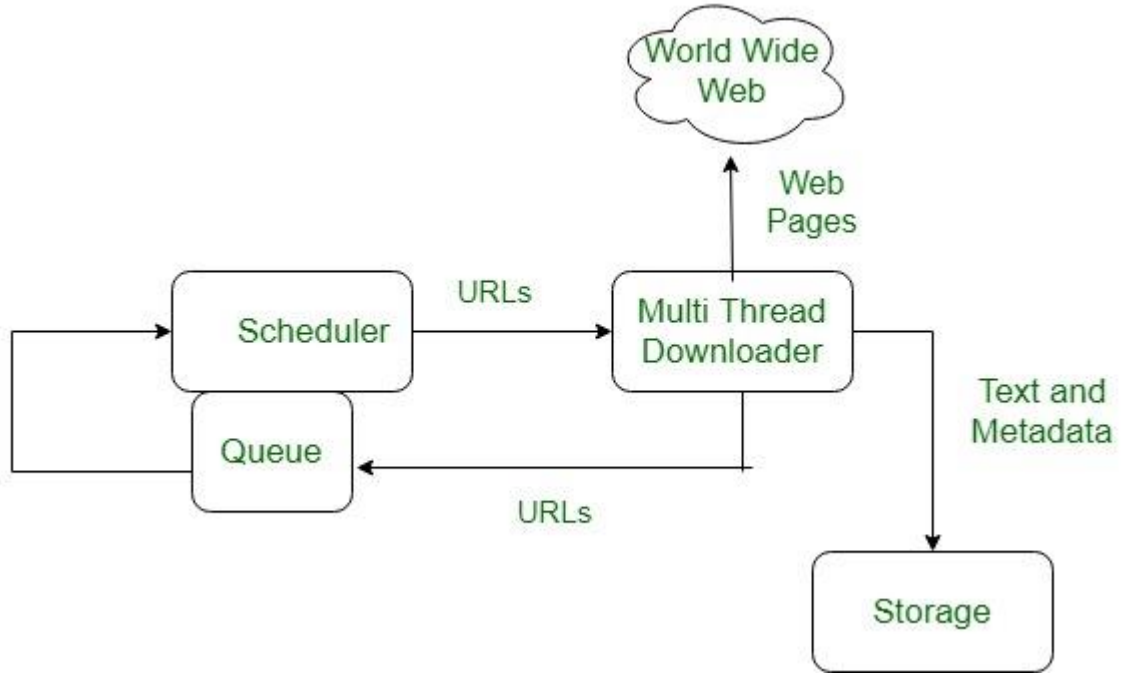
World Wide Web (WWW)

The **World Wide Web** is abbreviated as WWW and is commonly known as the web. The WWW was initiated by CERN (European library for Nuclear Research) in 1989.
WWW can be defined as the collection of different websites around the world, containing different information shared via local servers(or computers).

History:
It is a project created, by Timothy Berner Lee in 1989, for researchers to work together effectively at CERN. is an organization, named the World Wide Web Consortium (W3C), which was developed for further development of the web. This organization is directed by Tim Berner's Lee, aka the father of the web.

System Architecture:
From the user's point of view, the web consists of a vast, worldwide connection of documents or web pages. Each page may contain links to other pages anywhere in the world. The pages can be retrieved and viewed by using browsers of which internet explorer, Netscape Navigator, Google Chrome, etc are the popular ones. The browser fetches the page requested interprets the text and formatting commands on it, and displays the page, properly formatted, on the screen.

The basic model of how the web works are shown in the figure below. Here the browser is displaying a web page on the client machine. When the user clicks on a line of text that is linked to a page on the abd.com server, the browser follows the hyperlink by sending a message to the abd.com server asking it for the page.



Here the browser displays a web page on the client machine when the user clicks on a line of text that is linked to a page on abd.com, the browser follows the hyperlink by sending a message to the abd.com server asking for the page.

Working of WWW:

The World Wide Web is based on several different technologies: Web browsers, Hypertext Markup Language (HTML) and Hypertext Transfer Protocol (HTTP).
A Web browser is used to access web pages. Web browsers can be defined as programs which display text, data, pictures, animation and video on the Internet. Hyperlinked resources on the World Wide Web can be accessed using software interfaces provided by Web browsers. Initially, Web browsers were used only for surfing the Web but now they have become more universal. Web browsers can be used for several tasks including conducting searches, mailing, transferring files, and much more. Some of the commonly used browsers are Internet Explorer, Opera Mini, and Google Chrome.

Features of WWW:

- HyperText Information System
- Cross-Platform
- Distributed
- Open Standards and Open Source
- Uses Web Browsers to provide a single interface for many services
- Dynamic, Interactive and Evolving.
- “Web 2.0”

Components of the Web: There are 3 components of the web:

1. **Uniform Resource Locator (URL):** serves as a system for resources on the web.
2. **HyperText Transfer Protocol (HTTP):** specifies communication of browser and server.
3. **Hyper Text Markup Language (HTML):** defines the structure, organisation and content of a webpage.

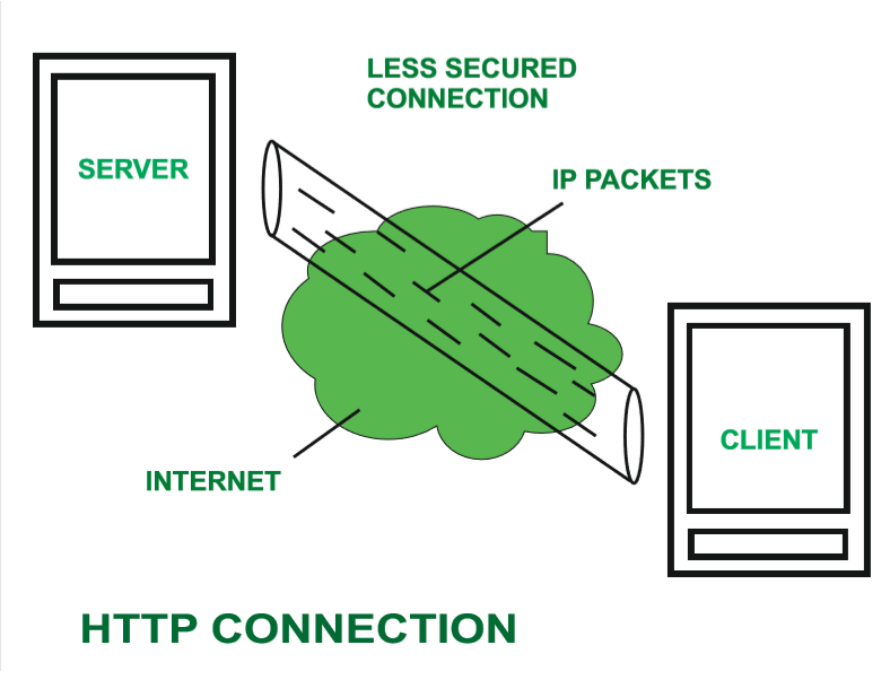
HTTP

HTTP stands for HyperText Transfer Protocol. It is invented by **Tim Berner**. HyperText is the type of text which is specially coded with the help of some standard coding language called HyperText Markup Language (HTML). **HTTP/2** is the successor version of HTTP, which was published on May 2015. HTTP/3 is the latest version of HTTP, which is published in 2022.The protocols that are used to transfer hypertext between two computers is known as HyperText Transfer Protocol.

HTTP provides standard between a web browser and web server to establish communication. It is set of rules for transferring data from one computer to another. Data such as text, images, and other multimedia files are shared on the World Wide Web. Whenever a web user opens their web browser, user indirectly uses HTTP. It is an application protocol which is used for distributed, collaborative, hypermedia information systems.

How it works ?

First of all, whenever we want to open any website then first we open web browser after that we will type URL of that website (e.g., www.facebook.com). This URL is now sent to Domain Name Server (DNS). Then DNS first check records for this URL in their database, then DNS will return IP address to web browser corresponding to this URL. Now browser is able to sent request to actual server. After server sends data to client, connection will be closed. If we want something else from server we should have to re-establish connection between client and server.



History ::
Tim Berners Lee and his team at CERN gets credit for inventing original HTTP and associated technologies.

1. **HTTP version 0.9 –**
   This was first version of HTTP which was introduced in 1991.

2. **HTTP version 1.0 –**
   In 1996, RFC 1945 (Request For Comments) was introduced in HTTP version 1.0.

3. **HTTP version 1.1 –**
   In January 1997, RFC 2068 was introduced in HTTP version 1.1. Improvements and updates to HTTP version 1.1 standard were released under RFC 2616 in June 1999.

4. **HTTP version 2.0 –**
   The HTTP version 2.0 specification was published as RFC 7540 on May 14, 2015.

5. **HTTP version 3.0 –**
   HTTP version 3.0 is based on previous RFC draft. It is renamed as HyperText Transfer Protocol QUIC which is a transport layer network protocol developed

by Google.

Characteristics of HTTP: HTTP is IP based communication protocol which is used to deliver data from server to client or vice-versa.

1. Server processes a request, which is raised by client and also server and client knows each other only during current request and response period.
2. Any type of content can be exchanged as long as server and client are compatible with it.
3. Once data is exchanged then servers and client are no more connected with each other.
4. It is a request and response protocol based on client and server requirements.
5. It is connection less protocol because after connection is closed, server does not remember anything about client and client does not remember anything about server.
6. It is stateless protocol because both client and server does not expecting anything from each other but they are still able to communicate.

Advantages :

- Memory usage and CPU usage are low because of less simultaneous connections.
- Since there are few TCP connections hence network congestion are less.
- Since handshaking is done at initial connection stage, then latency is reduced because there is no further need of handshaking for subsequent requests.
- The error can be reports without closing connection.
- HTTP allows HTTP pipe-lining of request or response.

Disadvantages :

- HTTP requires high power to establish communication and transfer data.
- HTTP is less secure, because it does not uses any encryption method like https use TLS to encrypt normal http requests and response.
- HTTP is not optimized for cellular phone and it is too gabby.
- HTTP does not offer genuine exchange of data because it is less secure.
- Client does not close connection until it receives complete data from server and hence server needs to wait for data completion and cannot be available for other clients during this time.

FTP

- FTP stands for File transfer protocol.
- FTP is a standard internet protocol provided by TCP/IP used for transmitting the files from one host to another.
- It is mainly used for transferring the web page files from their creator to the computer that acts as a server for other computers on the internet.

- It is also used for downloading the files to computer from other servers.

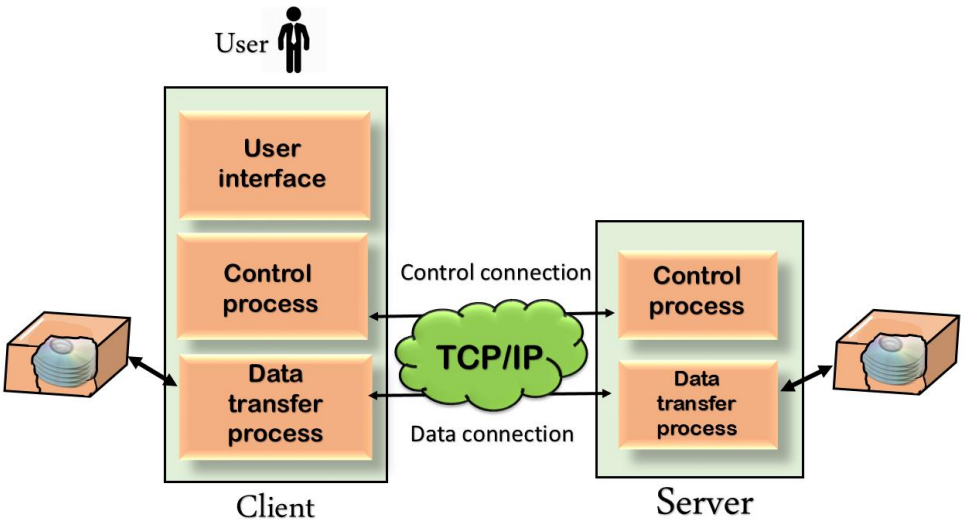
Objectives of FTP

- It provides the sharing of files.
- It is used to encourage the use of remote computers.
- It transfers the data more reliably and efficiently.

Why FTP?

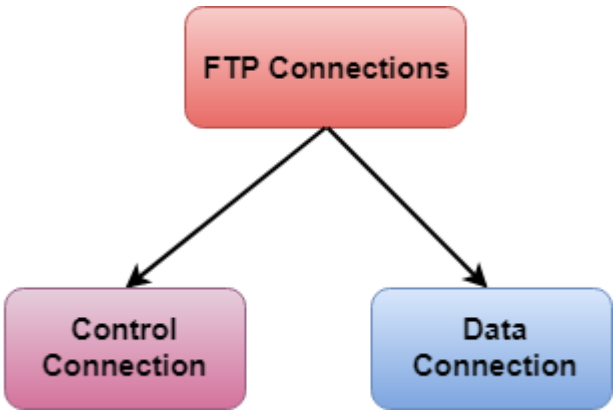
Although transferring files from one system to another is very simple and straightforward, but sometimes it can cause problems. For example, two systems may have different file conventions. Two systems may have different ways to represent text and data. Two systems may have different directory structures. FTP protocol overcomes these problems by establishing two connections between hosts. One connection is used for data transfer, and another connection is used for the control connection.

Mechanism of FTP



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:



- **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.
- **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.

FTP Clients

- FTP client is a program that implements a file transfer protocol which allows you to transfer files between two hosts on the internet.
- It allows a user to connect to a remote host and upload or download the files.
- It has a set of commands that we can use to connect to a host, transfer the files between you and your host and close the connection.
- The FTP program is also available as a built-in component in a Web browser. This GUI based FTP client makes the file transfer very easy and also does not require to remember the FTP commands.

Advantages of FTP:

- **Speed:** One of the biggest advantages of FTP is speed. The FTP is one of the fastest way 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.

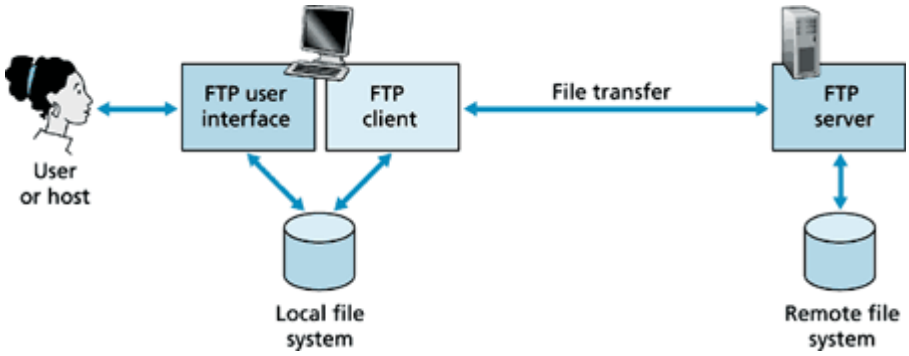
Disadvantages of FTP:

- The standard requirement of the industry is that all the FTP transmissions should be encrypted. However, not all the FTP providers are equal and not all the providers offer encryption. So, we will have to look out for the FTP providers that provides encryption.
- FTP serves two operations, i.e., to send and receive large files on a network. However, the size limit of the file is 2GB that can be sent. It also doesn't allow you to run simultaneous transfers to multiple receivers.
- Passwords and file contents are sent in clear text that allows unwanted eavesdropping. So, it is quite possible that attackers can carry out the brute force attack by trying to guess the FTP password.
- It is not compatible with every system.

Security for FTP

A Definition of FTP Security

File Transfer Protocol (FTP) is a standard network protocol used to transfer files between computers over the Internet. FTP is built on client-server architecture and was developed by Abhay Bhushan in 1971. The protocol is still commonly used today, but FTP security is a major concern that can limit its usage when not addressed.



Security Challenges of FTP

FTP was not built to be secure. It is generally considered to be an insecure protocol because it relies on clear-text usernames and passwords for authentication and does not use encryption. Data sent via FTP is vulnerable to sniffing, spoofing, and brute force attacks, among other basic attack methods.

There are several common approaches to addressing these challenges and securing FTP usage. FTPS is an extension of FTP that can encrypt connections at the client's request. Transport Layer Security (TLS), Secure Socket Layer (SSL), and SSH File Transfer Protocol (also known as Secure File Transfer Protocol or SFTP) are often used as more secure alternatives to FTP because they use encrypted connections

Network Data Loss Prevention Improves FTP Security

Network data loss prevention solutions are often used to secure data sent over FTP sessions. Network DLP solutions are able to inspect and control FTP traffic, blocking or allowing transfers based on policies governing what users can take what actions with data. NDLP solutions can also encrypt data sent via FTP to ensure it is only readable by authorized parties.

Network data loss prevention solutions also are crucial for FTP security in cases when employees may inadvertently share sensitive data and confidential files using FTP. By prompting users, encrypting files, or blocking unauthorized FTP transfers altogether, network DLP tools ensure that sensitive data is not being put at risk of interception or exfiltration.

While FTP has inherent data security risks, the use of alternative secure protocols and data protection solutions such as network DLP can enable secure FTP usage.

ELECTRONIC MAIL: Architecture

Introduction

Electronic mail (e-mail) is a computer-based program that allows users to send and receive messages. E-mail is the electronic version of a letter, but with time and flexibility advantages. While a letter can take anywhere from a week to a couple of months to reach its intended destination, an e-mail is sent virtually almost instantly.

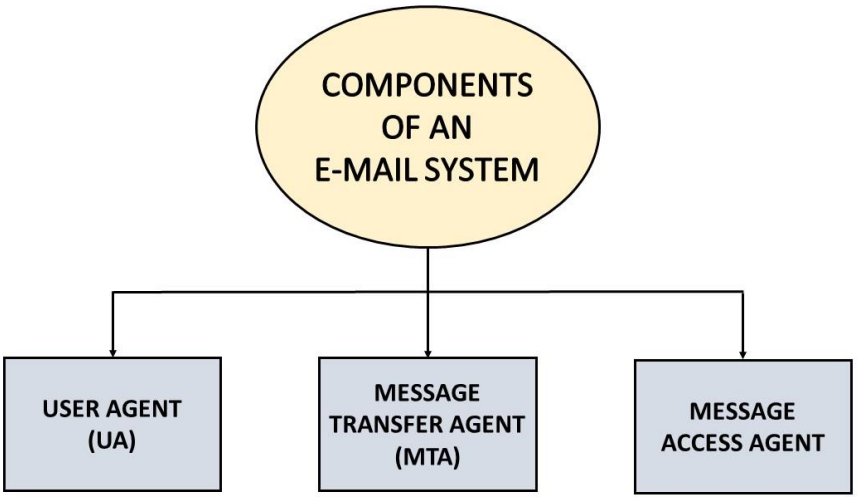
Messages in the mail contain not just text but also photos, audio, and video data. A person sending an e-mail is a **sender**, and the person receiving it is the **recipient**.

Components Of Electronic Mail

The following are the essential components of an e-mail system:

1. User Agent (UA)

2. Message Transfer Agent (MTA)
3. Message Access Agent



User Agent (UA)

The User-Agent is a simple software that sends and receives mail. It is also known as a mail reader. It supports a wide range of instructions for sending, receiving, and replying to messages and manipulating mailboxes.

Some of the services supplied by the User-Agent are listed below:

- Reading a Message
- Sending a reply to a Message
- Message Composition
- Forwarding a Message
- Handling the Message

Message Transfer Agent

The Message Transfer Agent manages the actual e-mail transfer operation (MTA). Simple Mail Transfer Protocol sends messages from one MTA to another. A system must have a client MTA and a system MTA to send an e-mail. If the recipients are connected to the same computer, it sends mail to their mailboxes. If the destination mailbox is on another computer, it sends mail to the receiver's MTA.

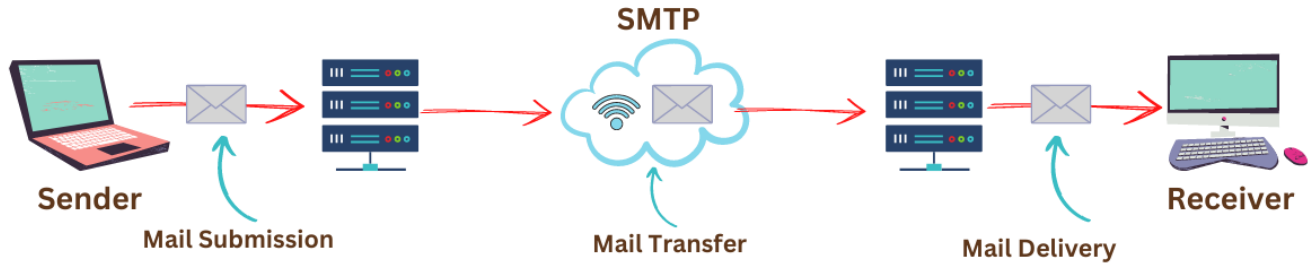
Message Access Agent

The Simple Mail Transfer Protocol is used for the first and second stages of e-mail delivery.

The pull protocol is mainly required at the third stage of e-mail delivery, and the message access agent is used at this point.

POP and IMAP4 are the two protocols used to access messages.

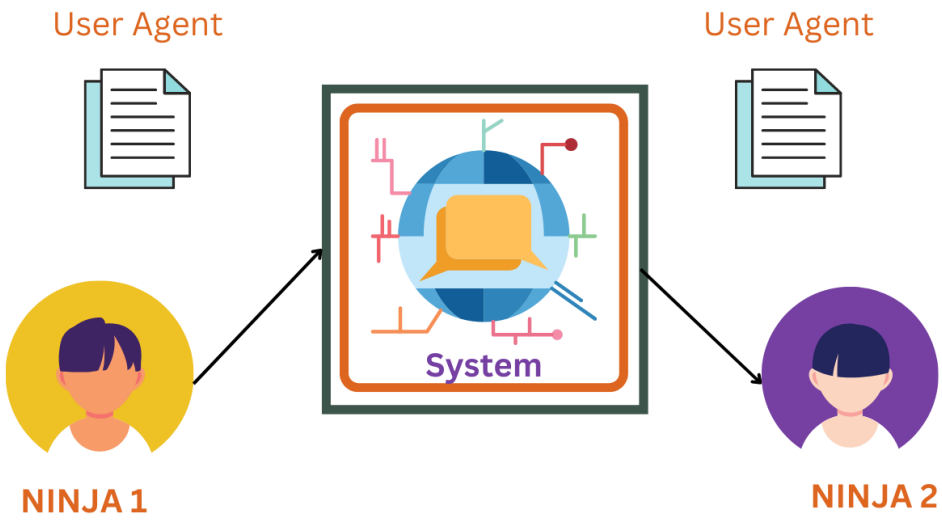
Architecture of Electronic Mail



First Scenario

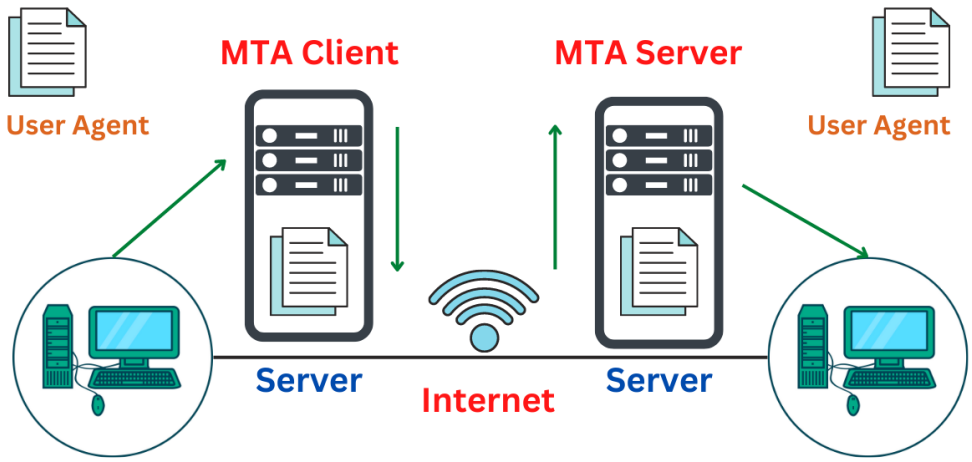
In the first scenario, two user agents are required. The sender and recipient of the e-mail share the same machine directly connected to the server.

For example, let us consider two user agents, Ninja1 and Ninja2. When Ninja1 sends an e-mail to Ninja2, the user agent (UA) programme is used to prepare the message. Following that, this e-mail gets saved in the Ninja2 inbox.



Second Scenario

In this case, the sender and recipient of an e-mail are essentially users on two different machines over the internet. User-Agents and Message Transfer Agents(MTA) are required in this scenario.

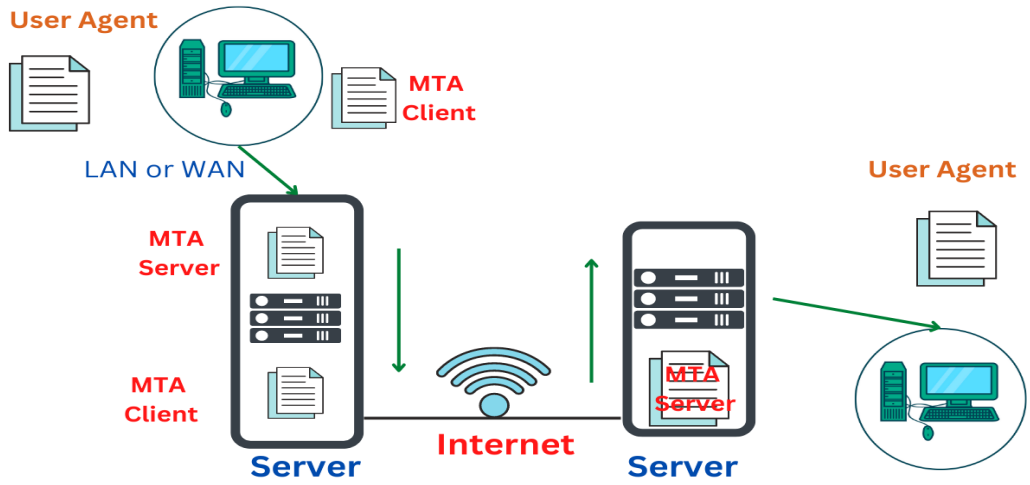


Take, for example, two user agents (Ninja1 and Ninja2), as illustrated in the diagram. When Ninja1 sends an e-mail to Ninja2, the user agent (UA) and message transfer agents (MTAs) programmes prepare the e-mail for transmission over the internet. Following that, this e-mail gets stored in Ninja2's inbox.

Third Scenario

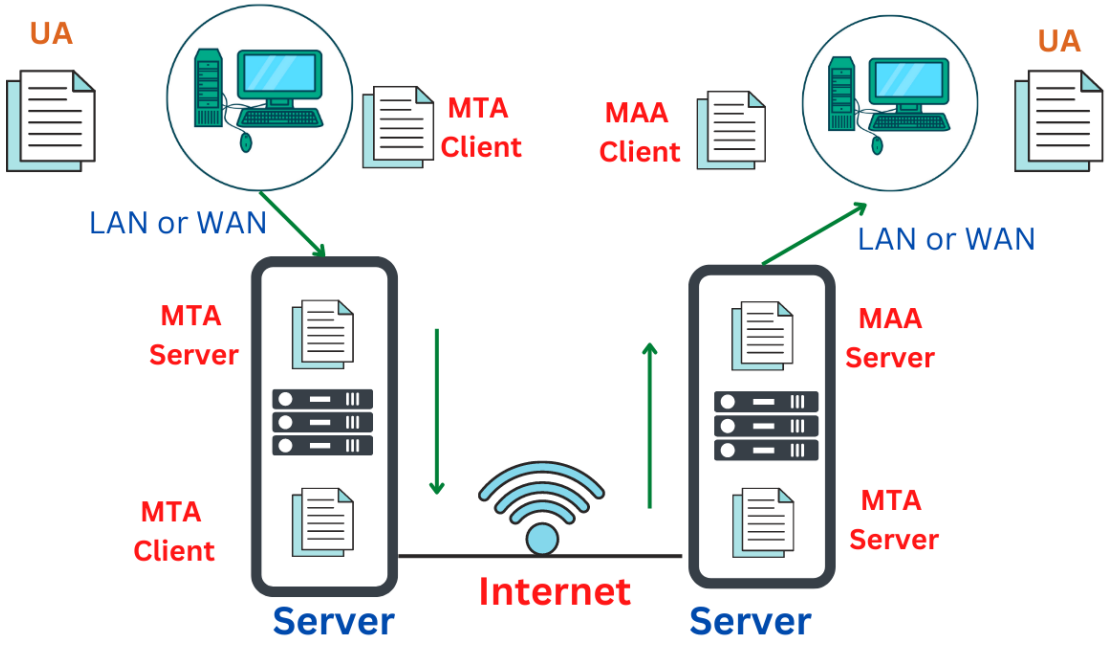
The sender is connected to the system by a point-to-point WAN, which can be a dial-up modem or a cable modem in this case. On the other hand, the receiver is directly attached to the system, as it was in the second scenario.

The sender also needs a User agent (UA) to prepare the message in this situation. After preparing the statement, the sender delivers it over LAN or WAN via a pair of MTAs.



Fourth Scenario

In this scenario, the recipient is linked to the mail server via WAN or LAN. When the message arrives, the recipient must retrieve it, which needs additional client/server agents. This scenario requires two user agents (UAs), two pairs of message transfer agents (MTAs), and a couple of message access agents (MAAs).



Web based e-mail

The term Webmail (or Web-based email) is used to describe two things. One use of the word is to describe a Webmail client: an email client implemented as a web application accessed via a web browser.

The other use of the word is to describe a Web-based email service: an email service offered through a web site (a webmail provider) such as Gmail, Yahoo! Mail, Hotmail and AOL Mail. Practically every webmail provider offers email access using a webmail client, and many of them also offer email access by a desktop email client using standard email protocols, while many internet service providers provide a webmail client as part of the email service included in their internet service package.

As with any web application, webmail's main advantage over the use of a desktop email client is the ability to send and receive email anywhere from a web browser. Its main disadvantage is the need to be connected to the internet while using it (Gmail offers offline use of its webmail client through the installation of Gears.[2]). There exist also other software tools to integrate parts of the webmail functionality into the OS (e.g. creating messages directly from third party applications via MAPI).

Rendering and compatibility

Email users may find the use of both a webmail client and a desktop client using the POP3 protocol a bit incompatible: email messages that are downloaded by the desktop client and are removed from the server will no longer be available on the webmail client. The use of a webmail client in this mode is limited to previewing messages using a web client before they are downloaded by the desktop email client. On the other hand, the use of both a webmail client and a desktop client using the IMAP4 protocol has no such incompatibility: the contents of the mailbox will be consistently displayed in both the webmail and the desktop email client and any action the user performs on messages in one interface would be reflected when email is accessed using the other interface. There are significant differences in rendering capabilities for many popular webmail services such as Yahoo! Mail, Gmail, and Windows Live Hotmail. Due to the various treatment of HTML tags, such as <style> and <head>, as well as CSS rendering inconsistencies, email marketing companies rely on older web development techniques to send cross-platform mail. This usually means a greater reliance on tables and inline stylesheets.

Privacy concerns

Although every email service provider can read the email unless encrypted since it is stored on their servers, concerns have been raised about webmail specifically. Most popular webmail services tend to use what's called targeting ads and online spam-filter(instead of a client-based), these services searches through email for certain target words and even if the service providers claim that no humans reads the emails some of them have been forced to make it possible to opt this feature out. Because web browser is the expected way of viewing the inbox webmail providers store emails longer than usual providers which often delete the email from their servers after they have sent it to the email client.

Another concern is considering the fact that most webmail service providers are U.S.-based and therefore are subject to the Patriot Act which means that U.S. authorities can demand the company to handover what information they have about a user, without necessarily letting the user know, no matter what citizenship you have or where the information is stored

Example Webmail usage

Webmail operations are same as that of e-mail with difference on web browser based access instead of an e-mail client software and the various operations for an example webmail account are discussed

Reading and managing mails

When you first login, your inbox is automatically displayed. To view an email, simply click on the subject. The selected email will be highlighted and the entire message will get loaded on a new screen. Bold messages are new or unread. Messages you have already looked at will be in un-bolded text. To view a different folder, just click on the name of the folder you want

to view. The options for each folder work the same as your inbox. Let's go through what each of the buttons will do:

1. **Check Mail** – This button will download any new messages received. This happens automatically each time you log in, click on the inbox link or explicitly click on this button.
2. **Reply** – This is the reply button. It will automatically set up the composition page with the information necessary to reply to the sender of the selected message.
3. **Reply All** – This is just like the reply button, but it's used when the selected email is addressed to more the one person and you want to reply to everyone, not just the person who sent the message.
4. **Forward** – This will forward the selected message, and direct you to the composition page to enter your recipient.
5. **Delete** – No points for guessing. Clicking on this button will delete the selected/current message and move it to the trash folder. If you wish to permanently delete a message you will have to either Empty the trash folder or select the message explicitly and delete it.
6. **Compose** – This is the button you would click on to compose, write, or send a new email. When you click on this button, you will be sent to a new page to type out your email, subject, senders etc.
7. **Actions** – This button will gives you the option to mark an email or multiple emails as Read, Unread, Flagged or Unflagged.

Note: You may return to the current folder anytime either by clicking on the Back button or selecting that particular folder from the Email navigation page.

Mark messages as read/unread

Marking a particular email or a selected set of emails as read or unread is part of the "Actions" button available with
the webmail. Each action has its own significant impact on the listed email. After you have selected emails upon which an action is to be performed, you have the following list of actions to chose from:

1. Mark as *read* – Marks the selected email/s as read. The selected messages will be un-bolded after performing this action.
2. Mark as *unread* – Marks the selected email/s as unread. The selected messages will now appear in bold font indicating an unread message.
3. Mark as *flagged* – Marking a selected mail as flagged will illuminate a gold star at the end of the message line. This option is usually selected by users who wish to keep track or follow up on emails at a later time.
4. Mark as *unflagged* – Mails already marked as flagged will get unflagged upon selecting this action.

Forward received emails

Forwarding an email form your inbox to another email id can be achieved by checking the desired email and clicking 'Forward' button. This will forward the selected message, and direct you to the
email composition page to enter your recipient. The 'Forward' option is also available when you are reading an email.

Reply to received emails

You may reply to an email from your inbox or any of the created folders. You need to first select the email you wish to send a reply by checking the adjacent checkbox. This will activate the
Reply and **Reply All** buttons. Click on the 'Reply' button to automatically set up the composition page with the information necessary to reply to the sender of the selected message.

The 'Reply All' is just like the reply button, but it's used when the selected email is addressed to more the one person and you want to reply to everyone, not just the person who sent the message. The 'Reply' and 'Reply All' options are also available when you are reading an email.

Compose a new email

Click on the Compose button to go to the email composition page. This is the page you are sent to any time you reply, forward, or compose a new message. To CC or BCC someone, you need to click on the "Add CC" or "Add BCC" link. If you wish to have a Reply-To address click on the "Add Reply-To" link.

1. Manage Attachments – With this webmail, you can compose your emails in Rich/Plain text formats. You can also add attachments to your email on this page. To do this, click the *Add Attachment* link given under the
   Subject and click browse. Choose the file on your computer you wish to add, then click upload. To add another attachment, click the *Add Attachment* again.
2. Spell Check – The webmail has a spell check feature that scans through your message text and highlights any existing spelling errors. The *Check spelling*link can be located on the email composition page in the plain text editor
   mode. Clicking on this link will scan through your composed message and highlight spelling errors in it. When you click on a highlighted word in the spell check mode, the tool generates and lists suggestions to replace the misspelled word. Similarly, in the Rich Text editor mode the spell check tool
   can be toggled by clicking on Toggle spellchecker button located on the rich text editor toolbar.
3. Switch between Plain-Rich Text editors – The two editor types available with .pw webmail are Rich Formatting and Plain Text. The option to switch between the two is available at the bottom right corner of the screen of an email composition page**Note:** Switching from Rich Text to plain text will result in loss of formatting.
4. Change fonts/formatting – Changing fonts is part of the standard list features provided in the Rich Text editor on an email composition page. With this editor not only can you change fonts but also:
    1. *Align Text*
    2. *Insert Bullets and Numbering*
    3. *Insert Smileys and Symbols*
    4. *Insert and Edit images*
    5. *Change text and background color*
    6. *Perform Spell Checks*
    7. *Roll back changes*

When you are done and wish to send the message, just click on the *Send now* button at the bottom of the page. The *Cancel* button will take you back to the Inbox.

Folder Management

Folder management is part of the webmail settings. Click on the *Settings* link given on the top left side of the screen to enter the webmail settings page. Select the *Folders* tab to enter the folder management tool. This page provides you the option to:

1.
    1. Create a new folder – Enter a folder name in the given text box and click on the Create button. To create a sub-folder under an already existing one, you simply select it before creating a new folder.
    2. Rename an existing folder – You also have the option to rename an existing folder by clicking on the rename icon
    3. Delete an existing folder – Click on the *trash* icon to delete an existing folder. Please **note** that emails within a folder will also be deleted upon deletion of that folder

What is Email Security?

Email security refers to the methods and processes used to safeguard email accounts, information, and communications from unauthorized access, data loss, and other hostile threats.

Significance of Email Security Practices

Hackers and cybercriminals use email as a means to disseminate malware, spam, and phishing assaults. It's also one of the common ways to get into a business network and steal sensitive data.

Approximately 92 percent of all malware is distributed via email. Every day, 15 billion spam emails are sent, accounting for around 45 percent of all emails. Furthermore, 95% of corporate email hack damages ranged from $250 to 250 to 984,855.

Threats to Email Marketing

- **Spam** – Spam is defined as unsolicited emails sent in large numbers. Vector spam can contain links that download malware files in some situations.
- **Phishing** – Phishing is when hackers use false emails, adverts, links, or messages to steal personal information or gain access to internet accounts. Phishing is involved in 36% of breaches, according to Verizon.
- **Malware** – Malware is when cybercriminals use harmful code distributed in email communications to infect one or more machines. Email virus infections will increase by 600 percent in 2020.

- **Spoofing** – Spoofing is a spam and phishing assault tactic used by hackers. It is meant to deceive consumers into believing that the communication comes from someone or something they know or can trust.
- **Botnet Messages** – A botnet is a network of computers that have been infected with malware. It commands the 'bot-header,' a single assaulting party. It's used to hack into devices, steal data, send spam, and get access to the device and its network.
- **BEC (Business Email Compromise)** – The tracker uses this approach to acquire access to a business email account and impersonate the owner. The attacker usually targets organizations that use wire transfers to send money to overseas vendors.

How Can You Identify an Email as a Threat?

Dangerous emails have some common features. Look out for the following attributes to identify emails that have been sent with a malicious intent −

Untrustworthy Email Address

Look for emails that utilize display name spoofing to hide the sender's true identity. These emails look like to have been sent by respectable organizations or trustworthy persons. Examine the sender's email address in the header for any small variations, such as extra characters or letters.

A Sense of Immediacy

In addition to verifying the email's header, you should also check the email's body. If you receive an unusual request that makes you feel compelled to act, it's likely that it includes malware. As a result, examine the email's wording for any feeling of urgency. Check for grammar and spelling issues, as most spam emails are poorly written.

Requests for Information Verification

Any email that requests you to verify, evaluate, check, or confirm any information is most likely a virus email. As a result, double-check the sender's email address before responding.

Suspicious Links

Malware might be contained in an email with an unexpected attachment that asks you to open it. .zip, .xls, .js, .pdf, .ace, .arj, .wsh, .scr, .exe, .com, .bat, and .doc are examples of suspicious attachment file extensions.

A Link Must Be Clicked

Keep an eye out for emails that push you to visit a website. It might be infected with malware! Check the URL before clicking on the link. If it's a hyperlinked link, hover your cursor over the text and double-check the link before clicking.

Email Security Best Practices

Following are some of the best practices in email security that work −

- *Email marketing should be encrypted*. Customer-sensitive information is sometimes included in emails, making them susceptible. As a result, it's critical to protect these communications by encrypting all emails sent to and from your customers.
- *Email security software should be used*. Additionally, employ high-quality email and security solutions that aren't easily manipulated or hacked. Invest in password management software, as well as anti-phishing and anti-spoofing software.
- *Use two-factor authentication*. It's a common habit as well as an efficient security precaution. Before logging in, a user must submit two pieces of identifying information, making it far more difficult for hackers to get access to an account, even if they know the password.
- *Make sure the devices you use to log in are up to date*. With the rise of remote work, many workers are encouraged to work from home and use personal devices to access company email accounts. Personal gadgets, on the other hand, are far more difficult for an organization to track, posing a serious security concern.
- *Only connect to secure Wi-Fi networks*. If your firm doesn't utilize Wi-Fi or work from home, make sure you're always connected to the internet over a secure connection.

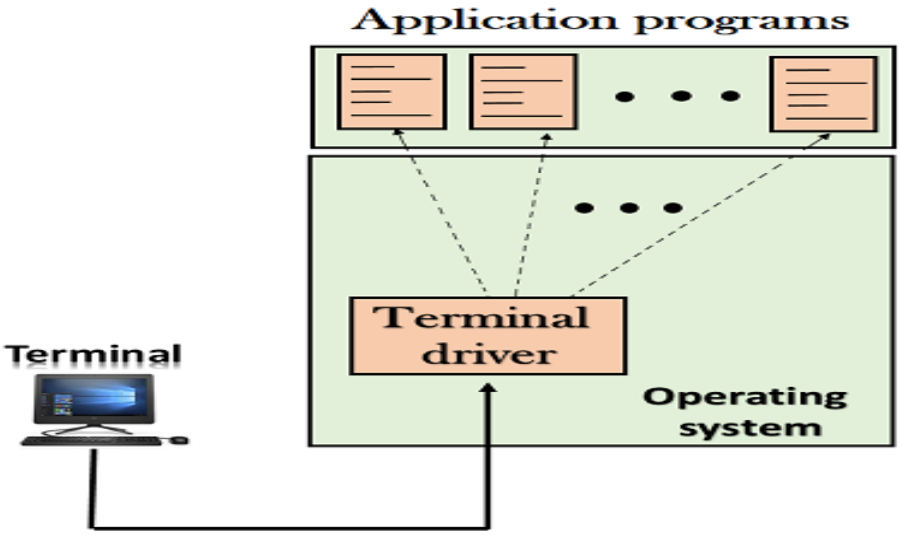
Telnet

- The main task of the internet is to provide services to users. For example, users want to run different application programs at the remote site and transfers a result to the local site. This requires a client-server program such as FTP, SMTP. But this would not allow us to create a specific program for each demand.
- The better solution is to provide a general client-server program that lets the user access any application program on a remote computer. Therefore, a program that allows a user to log on to a remote computer. A popular client-server program Telnet is used to meet such demands. Telnet is an abbreviation for **Terminal Network**.
- Telnet provides a connection to the remote computer in such a way that a local terminal appears to be at the remote side.

There are two types of login:

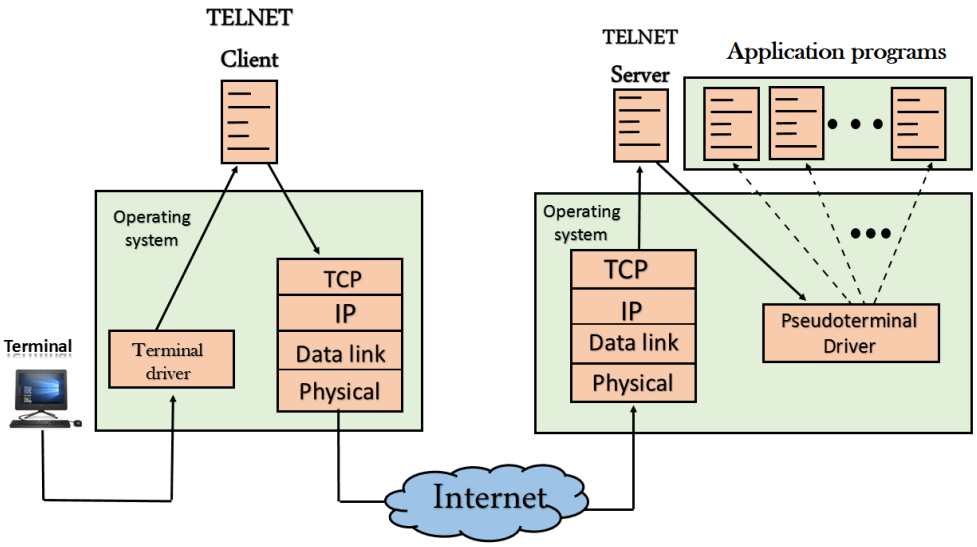
- Local Logging
- Remote Logging

Local Login



- When a user logs into a local computer, then it is known as local login.
- When the workstation running terminal emulator, the keystrokes entered by the user are accepted by the terminal driver. The terminal driver then passes these characters to the operating system which in turn, invokes the desired application program.
- However, the operating system has special meaning to special characters. For example, in UNIX some combination of characters have special meanings such as control character with "z" means suspend. Such situations do not create any problem as the terminal driver knows the meaning of such characters. But, it can cause the problems in remote login.

Remote login



- When the user wants to access an application program on a remote computer, then the user must perform remote login.
-

How remote login occurs

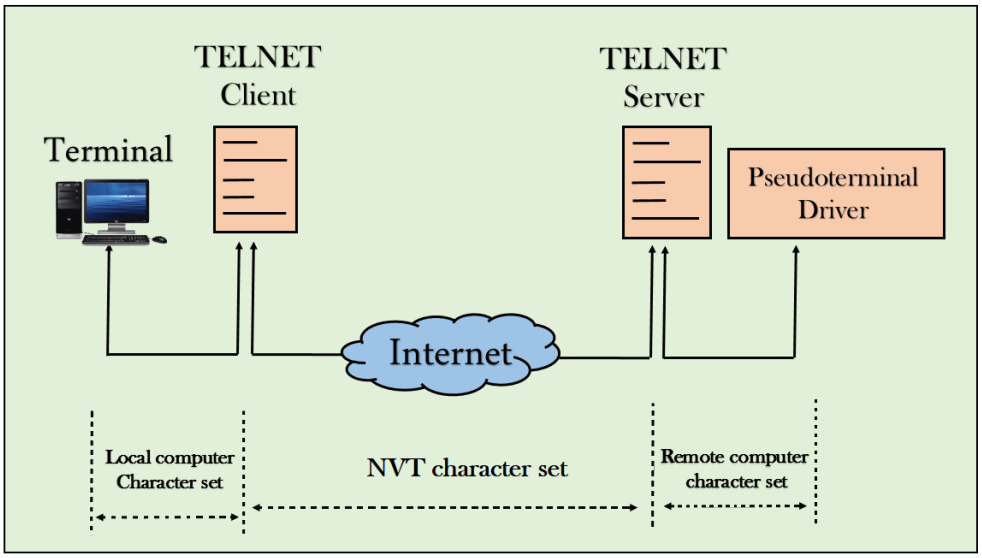
At the local site

The user sends the keystrokes to the terminal driver, the characters are then sent to the TELNET client. The TELNET client which in turn, transforms the characters to a universal character set known as network virtual terminal characters and delivers them to the local TCP/IP stack

At the remote site

The commands in NVT forms are transmitted to the TCP/IP at the remote machine. Here, the characters are delivered to the operating system and then pass to the TELNET server. The TELNET server transforms the characters which can be understandable by a remote computer. However, the characters cannot be directly passed to the operating system as a remote operating system does not receive the characters from the TELNET server. Therefore it requires some piece of software that can accept the characters from the TELNET server. The operating system then passes these characters to the appropriate application program.

Network Virtual Terminal (NVT)



- The network virtual terminal is an interface that defines how data and commands are sent across the network.
- In today's world, systems are heterogeneous. For example, the operating system accepts a special combination of characters such as end-of-file token running a DOS operating system *ctrl+z* while the token running a UNIX operating system is *ctrl+d*.
- TELNET solves this issue by defining a universal interface known as network virtual interface.
- The TELNET client translates the characters that come from the local terminal into NVT form and then delivers them to the network. The Telnet server then translates the data from NVT form into a form which can be understandable by a remote computer.

SSH (Secure Shell)

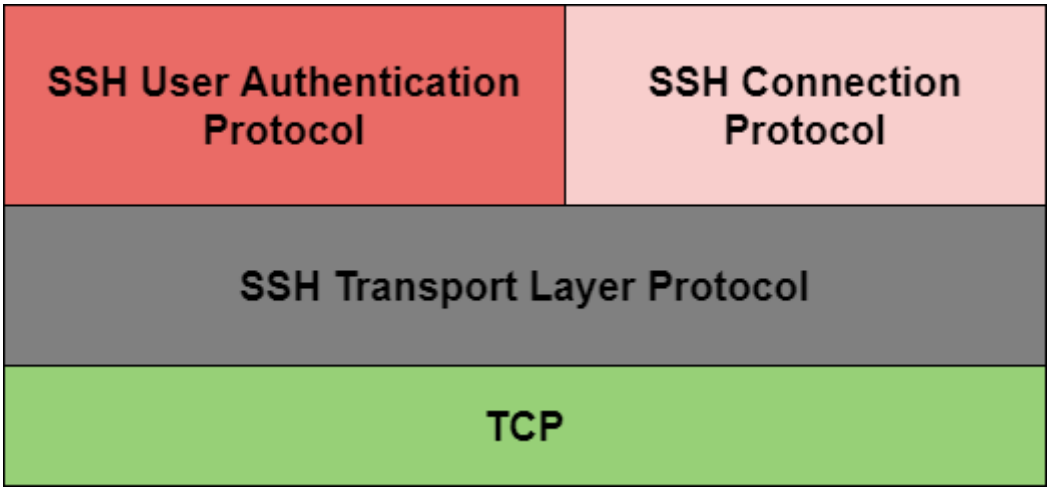
SSH is an abbreviation of Secure Shell. It is one of the major protocols that is used in order to access the network devices and servers over the Internet.

- It is basically a network protocol and it mainly runs on top of TCP/IP protocol.
- It is widely used to manage and access devices remotely.
- Also, the secure shell(SSH) mainly enables the two remotely connected users in order to perform network communication and other services on the top of an unsecured network.

- Thus it provides secure client/server communication and it can also be used for other tasks like **file transfer** and **e-mail**.
- With the help of SSH, you can log in to another computer over the network and it allows you to execute the commands on the remote machine.
- You can easily move files from one machine to another.
- This protocol mainly encrypts the traffic in both directions; with the help of this feature, you can prevent trafficking, sniffing, and password theft.
- By default, SSH runs on **Port number 22** and you can also change it.
- It is suitable for Public Networks.

Components of SSH

SSH is mainly organized in the form of three sub-protocols:



Let us discuss the above given in detail in the below section one by one:

1.SSH Transport Layer protocol

The Transport Layer protocol part of the SSH mainly used to provide the confidentiality of the data, the server /host authentication, and data integrity.

- Optionally it also provides data compression as well.
- **Server Authentication**
    - The Host keys are asymmetric in nature like public/private keys.
    - The server makes use of a Public key in order to prove its identity to the client.
    - Mainly the client verifies that contacted server is a "known" host with the help of the database that it maintains.
    - Once the authentication of the server is done then session keys are generated.

- **Session Key Establishment**
    - After the authentication of the server, the client and the server agree upon the cipher that is to be used.
    - The Session keys are usually generated by both the client as well as the server.
    - These keys are mainly generated before the user authentication so that usernames and passwords can be sent are encrypted.
    - The Session keys are generally replaced at regular intervals(we can say like an hour) during the session and then are destroyed immediately after use.
- **Data Integrity** SSH mainly makes the use of Message Authentication Code (MAC) algorithms in the order of the data integrity check.

2. SSH User Authentication Protocol

As the name suggests this part of the SSH is mainly used to authenticate the user to the server.

- This protocol is used for confirming the identity of the agent that is operating as the client.
- The server mainly identifies that the access should be given to intended users only.
- For the authentication purpose there are several methods that can be used;
    - Typed Passwords
    - Public-key authentication etc.

3. SSH Connection Protocol

The SSH Connection Protocol is mainly used to create distinct streams of data or logical channels, from the single client/server connection.

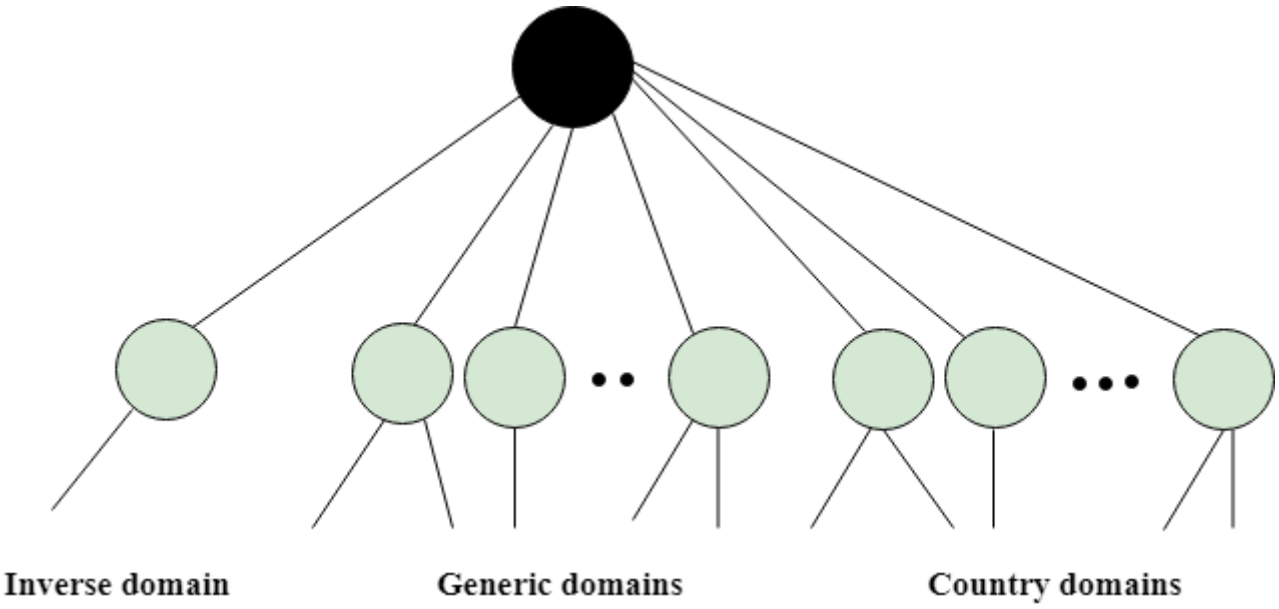
- Thus this protocol mainly provides multiple logic channels over the single underlying SSH connection.

DNS in the Internet

An application layer protocol defines how the application processes running on different systems, pass the messages to each other.

- DNS stands for Domain Name System.
- DNS is a directory service that provides a mapping between the name of a host on the network and its numerical address.
- DNS is required for the functioning of the internet.
- Each node in a tree has a domain name, and a full domain name is a sequence of symbols specified by dots.
- DNS is a service that translates the domain name into IP addresses. This allows the users of networks to utilize user-friendly names when looking for other hosts instead of remembering the IP addresses.
- For example, suppose the FTP site at EduSoft had an IP address of 132.147.165.50, most people would reach this site by specifying ftp.EduSoft.com. Therefore, the domain name is more reliable than IP address.

DNS is a TCP/IP protocol used on different platforms. The domain name space is divided into three different sections: generic domains, country domains, and inverse domain.

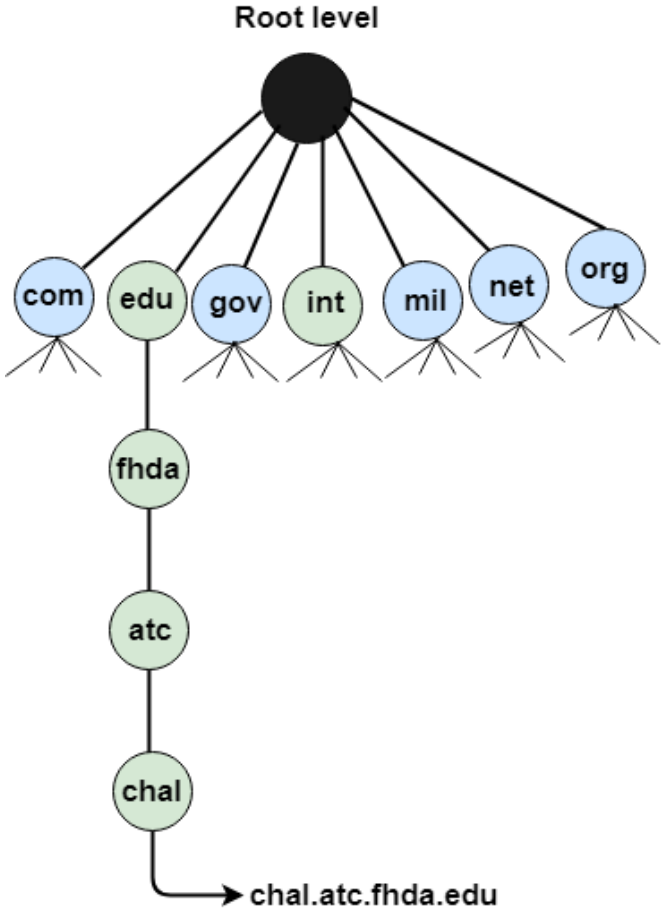


Generic Domains

- It defines the registered hosts according to their generic behavior.
- Each node in a tree defines the domain name, which is an index to the DNS database.
- It uses three-character labels, and these labels describe the organization type.

Label	Description
aero	Airlines and aerospace companies
biz	Businesses or firms
com	Commercial Organizations
coop	Cooperative business Organizations
edu	Educational institutions
gov	Government institutions
info	Information service providers
int	International Organizations

mil	Military groups
museum	Museum & other nonprofit organizations
name	Personal names
net	Network Support centers
org	Nonprofit Organizations
pro	Professional individual Organizations



Country Domain

The format of country domain is same as a generic domain, but it uses two-character country abbreviations (e.g., us for the United States) in place of three character organizational abbreviations.

Inverse Domain

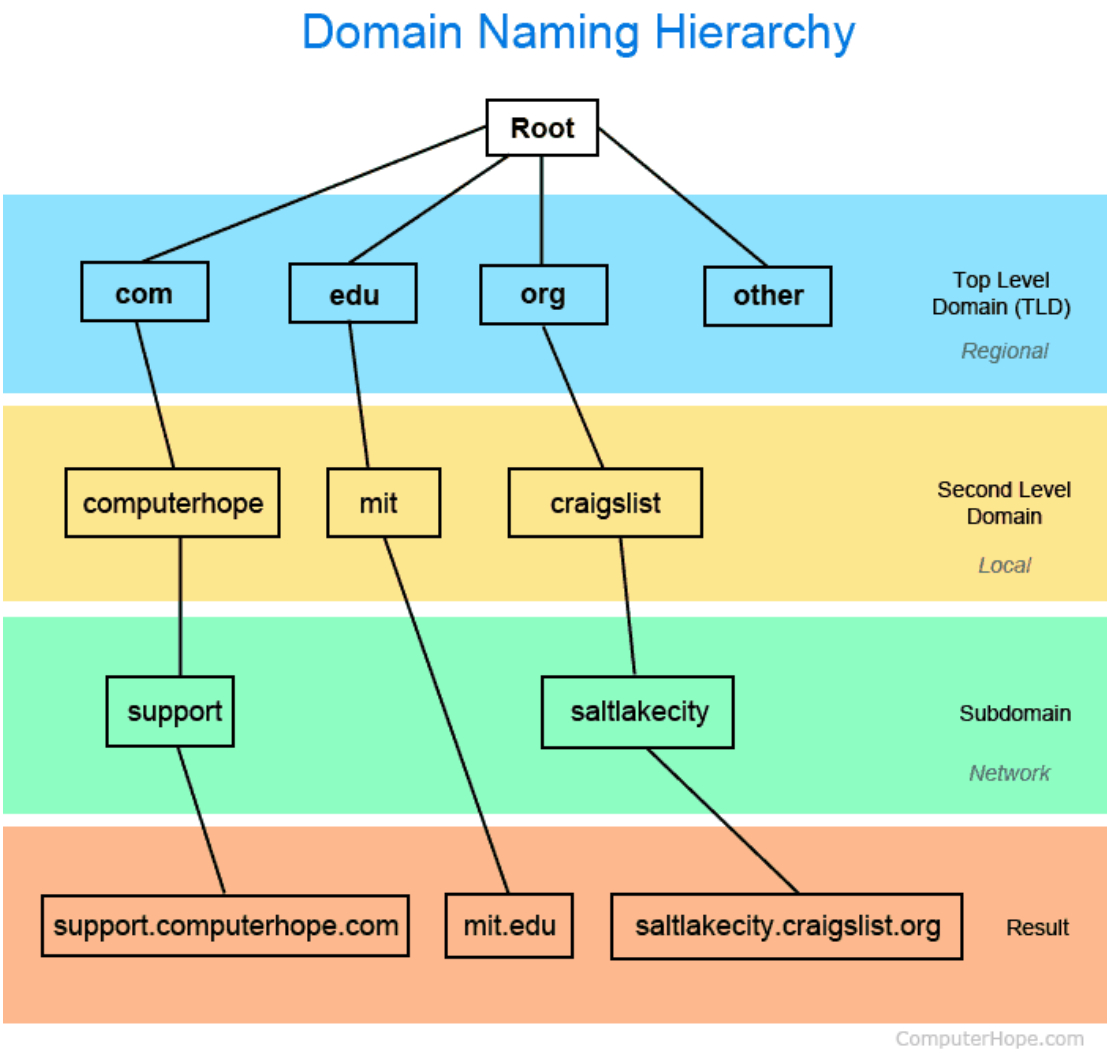
The inverse domain is used for mapping an address to a name. When the server has received a request from the client, and the server contains the files of only authorized clients. To determine whether the client is on the authorized list or not, it sends a query to the DNS server and ask for mapping an address to the name.

Working of DNS

- 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 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 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.

Domain namespace

Alternatively referred to as a **namespace**, a **domain namespace** is a name service provided by the Internet for Transmission Control Protocol networks/Internet Protocol (TCP/IP). DNS is broken up into domains, a logical organization of computers that exist in a larger network. Below is an example of the hierarchy of domain naming on the Internet.



In the example above, all websites are broken into regional sections based on the TLD (top-level domain).
With *http://support.computerhope.com*, it has a ".com" TLD, "computerhope" as its second level domain (local to the .com TLD), and "support" as its subdomain, which is determined by its server.

What is DNS caching? Where does DNS caching occur?

The purpose of caching is to temporarily stored data in a location that results in improvements in performance and reliability for data requests. DNS caching involves storing data closer to the requesting client so that the DNS query can be resolved earlier and additional queries further down the DNS lookup chain can be avoided, thereby improving load times and reducing bandwidth/CPU consumption. DNS data can be cached in a variety of locations, each of which will store DNS records for a set amount of time determined by a time-to-live (TTL).

Browser DNS caching

Modern web browsers are designed by default to cache DNS records for a set amount of time. The purpose here is obvious; the closer the DNS caching occurs to the web browser, the fewer processing steps must be taken in order to check the cache and make the correct requests to an IP address. When a request is made for a DNS record, the browser cache is the first location checked for the requested record.

In Chrome, you can see the status of your DNS cache by going to chrome://net-internals/#dns.

Operating system (OS) level DNS caching

The operating system level DNS resolver is the second and last local stop before a DNS query leaves your machine. The process inside your operating system that is designed to handle this query is commonly called a “stub resolver” or DNS client. When a stub resolver gets a request from an application, it first checks its own cache to see if it has the record. If it does not, it then sends a DNS query (with a recursive flag set), outside the local network to a DNS recursive resolver inside the Internet service provider (ISP).

DNS Record

What Does DNS Record Mean?

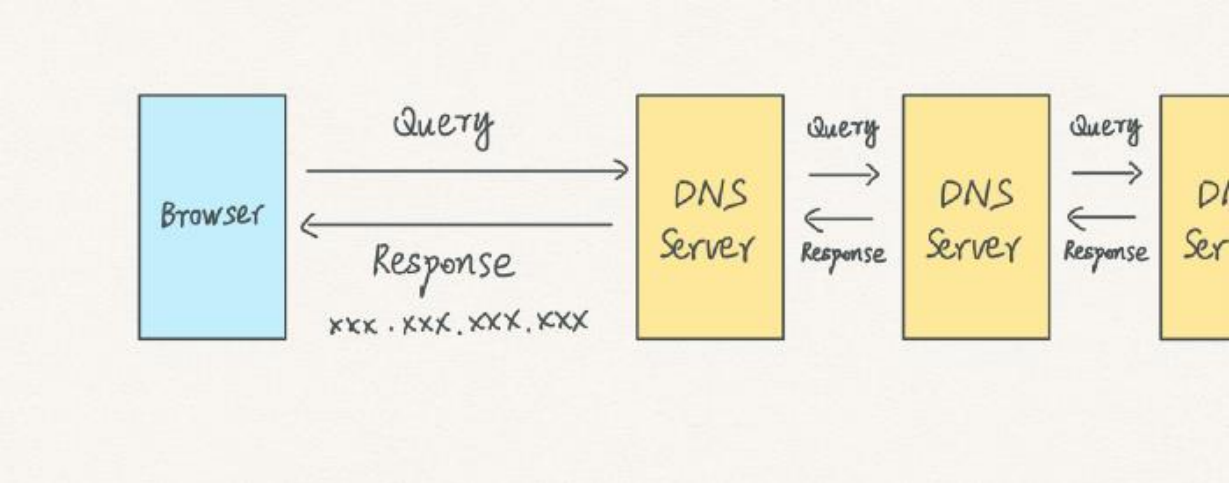
A DNS record is a database record used to map a URL to an IP address. DNS records are stored in DNS servers and work to help users connect their websites to the outside world. When the URL is entered and searched in the browser, that URL is forwarded to the DNS servers and then directed to the specific Web server. This Web server then serves the queried website outlined in the URL or directs the user to an email server that manages the incoming mail.

The most common record types are A (address), CNAME (canonical name), MX (mail exchange), NS (name server), PTR (pointer), SOA (start of authority) and TXT (text record).

Different types of DNS records are as follows:

- Name Server (NS) Record: Describes a name server for the domain that permits DNS lookups within several zones. Every primary as well as secondary name server must be reported via this record.
- Mail Exchange (MX) Record: Permits mail to be sent to the right mail servers located in the domain. Other than IP addresses, MX records include fully-qualified domain names.
- Address (A) Record: Used to map a host name to an IP address. Generally, A records are IP addresses. If a computer consists of multiple IP addresses, adapter cards, or both, it must possess multiple address records.
- Canonical Name (CNAME) Record: Can be used to set an alias for the host name
- Text (TXT) Record: Permits the insertion of arbitrary text into a DNS record. These records add SPF records into a domain.
- Time-to-Live (TTL) Record: Sets the period of data, which is ideal when a recursive DNS server queries the domain name information
- Start of Authority (SOA) Record: Declares the most authoritative host for the zone. Every zone file should include an SOA record, which is generated automatically when the user adds a zone.
- Pointer (PTR) Record: Creates a pointer, which maps an IP address to the host name in order to do reverse lookups.

DNS Message — How to Read Query and Response Message

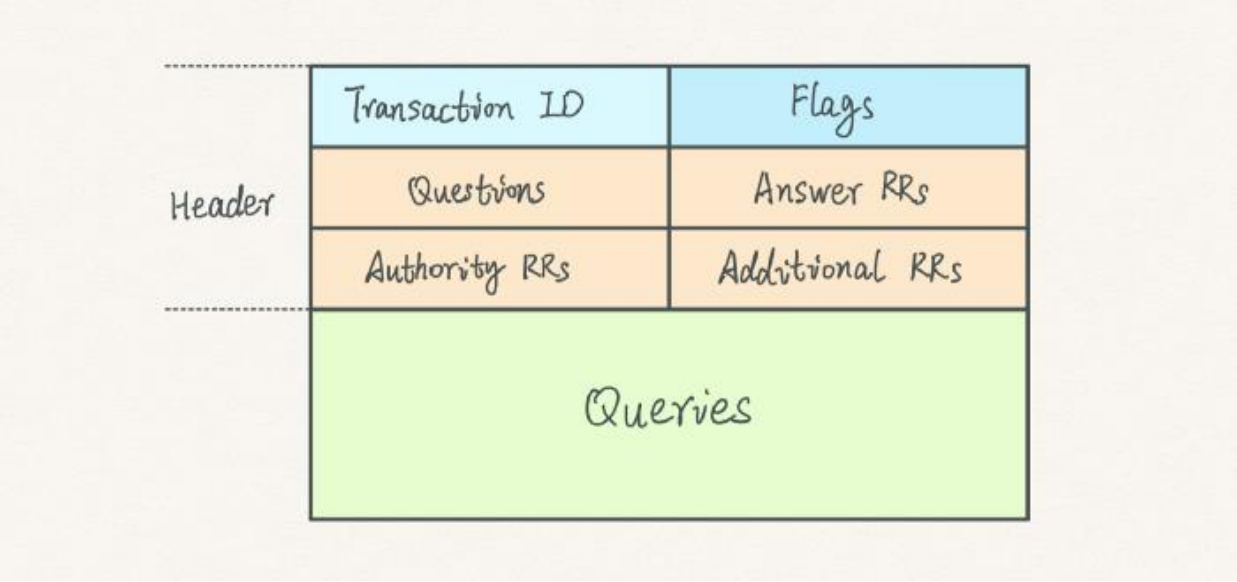


DNS message is relatively simple: the browser queries a domain name and gets an IP address.

If a DNS server doesn't recognize the domain name, it will pass the query along to the following DNS server. Later, when receiving a response, it carries the response to the browser.

Interesting in how DNS resolution works? Hope this post could help.

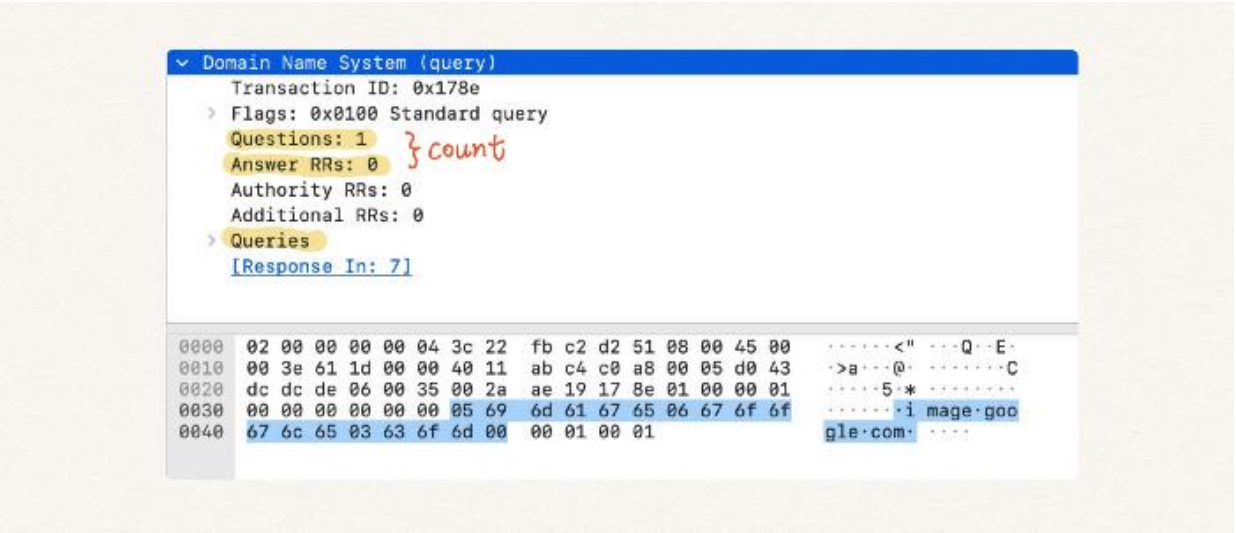
Query Message



Here is the query's message structure.

- **Transaction ID:** for matching response to queries
- **Flags:** specifies the requested operation and a response code
- **Questions:** count of entries in the queries section
- **Answer RRs:** count of entries in the answers section (RR stands for "resource record")
- **Authority RRs:** count of entries in the authority section
- **Additional RRs:** count of entries in the additional section
- **Queries:** queries data

Among them, what needs attention are **Questions, Answer RRs**, and **Queries**.



Here is an example of the query message for `image.google.com`.

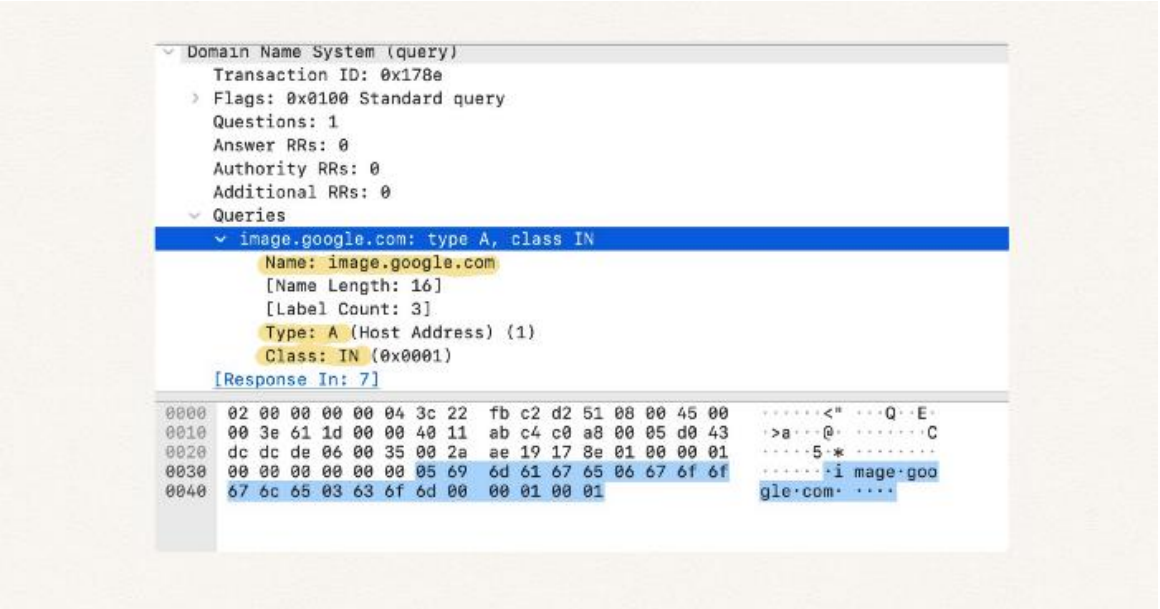
- `Questions: 1` means this message has one entry in the Queries.
- `Answer RRs: 0` means there are no answers. This is expected as a query message has only questions and no answers.

Next, let's dive into the entry structure of queries — merely 3 sections.

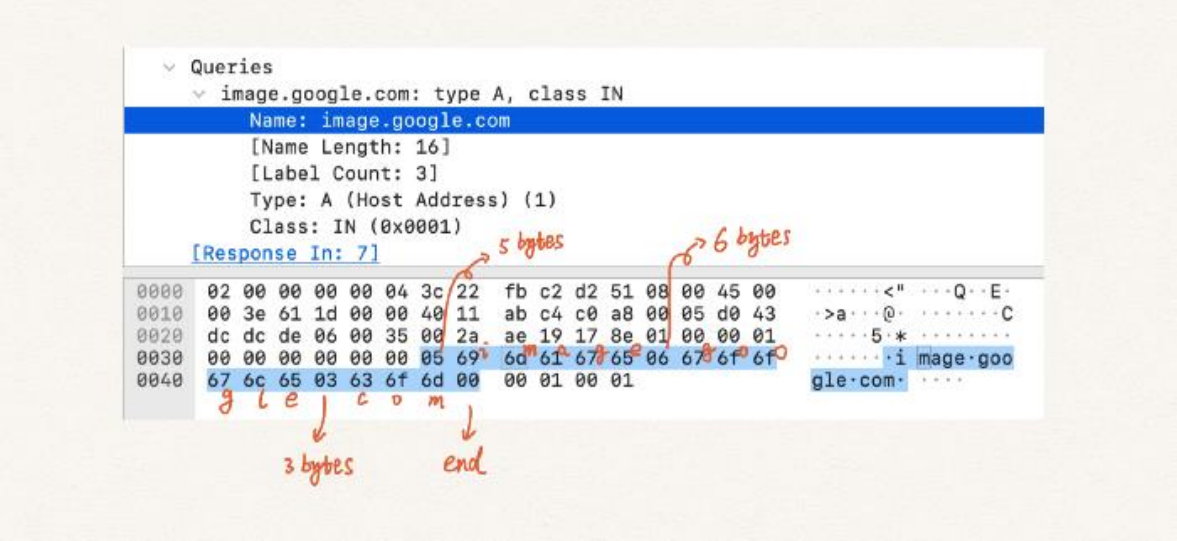
- **Name:** the domain name
- **Type:** DNS record type (e.g., A, CNAME, and MX)

- **Class:** allows domain names to be used for arbitrary objects

It is easier to understand the structure by taking a look at the example.



- `Name` is the requested domain `image.google.com`.
- `Type: A` means it is an A record. A record is the most basic and the most commonly used DNS record type.
- `Classs: IN` refers to "internet." It doesn't matter much in our browser context.



The interesting part is how the message codes the `Name` field.

Using `.` as a separator, the example domain can be divided into 3 groups.

- `image`
- `google`
- `com`

In the example marked in blue, the first byte is `05`, meaning the following 5 bytes are the 1st group of the domain.

In the screenshot, bytes are presented in ASCII codes. We can easily decode them into characters.

- `69` → i
- `6d` → m

- `61` → a
- `67` → g
- `65` → e

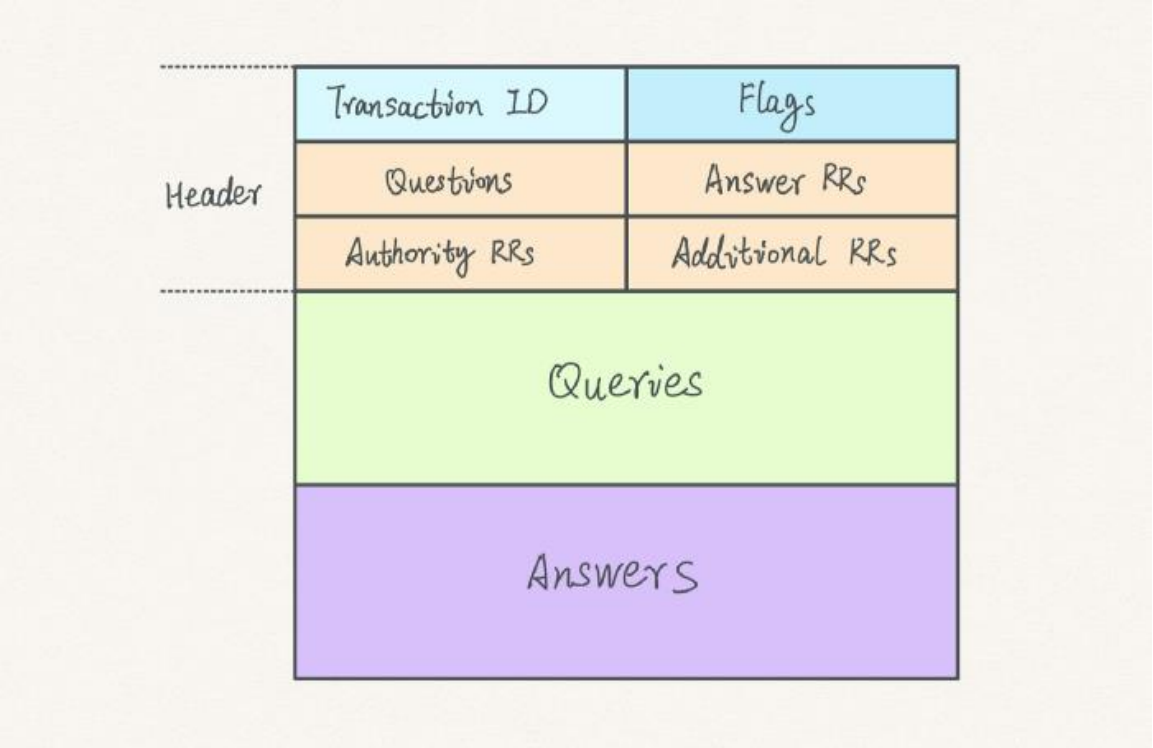
We get the `image`.

Following the same rule, we can find the remaining part of the domain — `google` and `com`.

Finally, at the end of the domain, a `00` marks the end of the section.

That's it for the query. With all required information provided by the query, the DNS server will send a response message.

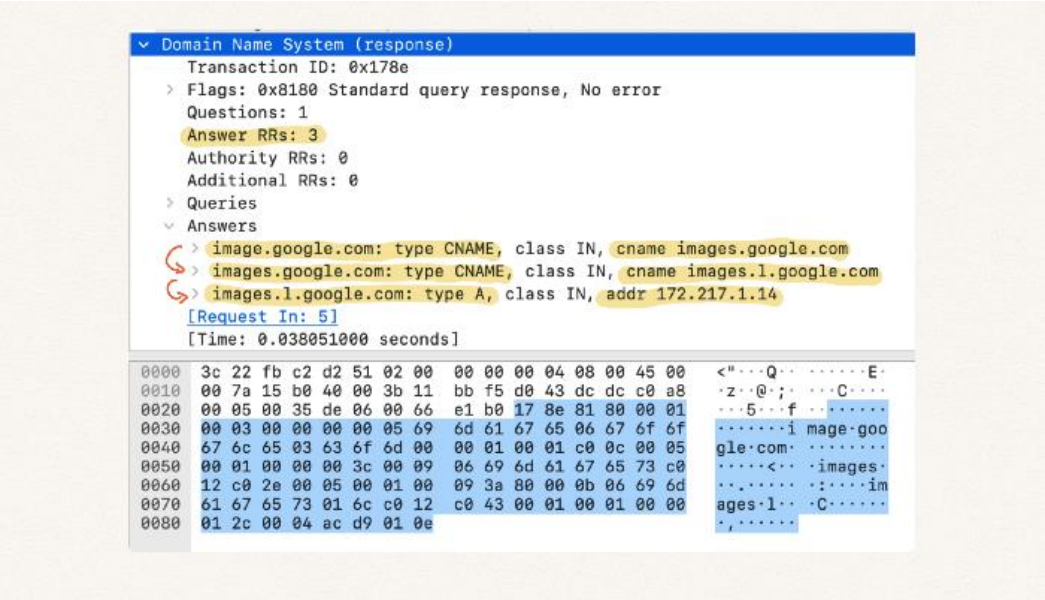
Response Message



A response message shares the same header and `Queries` with an additional `Answers` section.

Why does a response message include the origin `Queries` section? It is for reference. We will get to it soon.

Here is a response example from querying `image.google.com`.

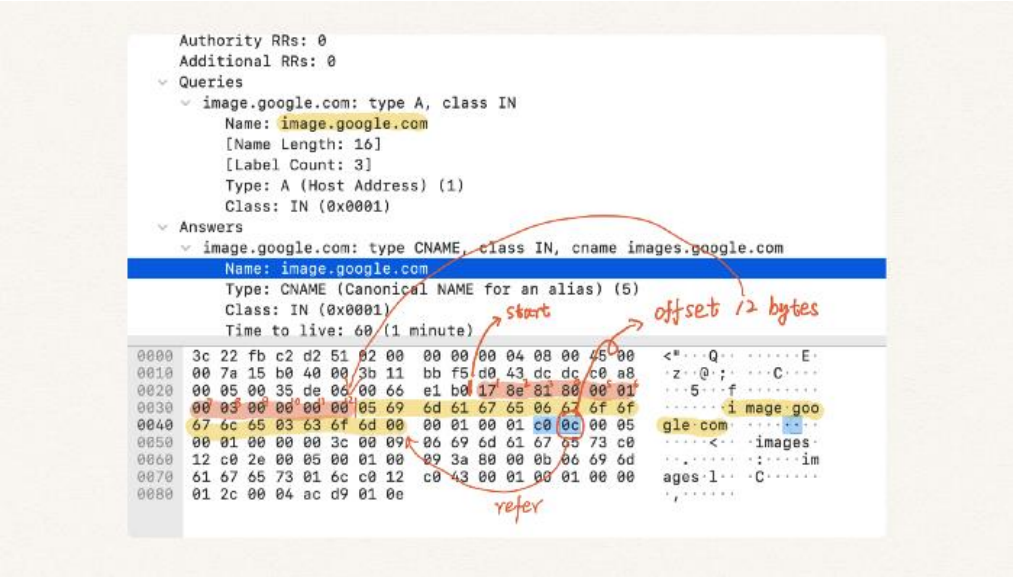


In the message, we receive 3 entries in the `Answers` section. Therefore, `Answer RRs` is set to 3.

1. In the **1st entry**, the DNS server returns a CNAME `images.google.com` for the initial query.
2. Then, a new query for `images.google.com` is sent, and another CNAME `images.1.google.com` is returned in the **2nd entry**.
3. Finally, by querying `images.1.google.com`, the client receives the IP address `172.217.1.14` in the **last entry**.

Besides the same 3 sections found in a query entry, an answer entry has 3 additional pieces.

- **Time to Live (TTL):** number of seconds this record can live
- **Data Length:** the length of the data
- **Data:** the returned data, such as an IP address or CNAME



Let's take a look at the `Name` section, which has merely two bytes: `c0 0c`.

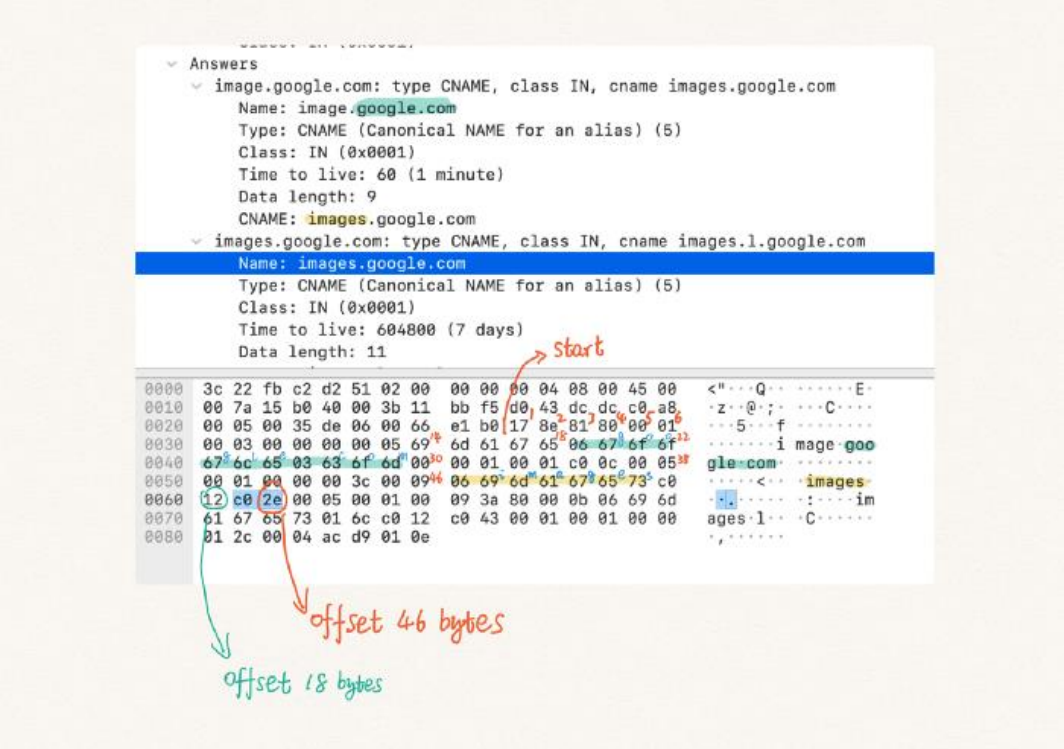
How a domain coded in two bytes?

It turns out that the bytes are an **offset**, referring to the coded domain name in the `Queries` section.

`c0` is a beginning mark, while `0c` is the actual offset, which is 12.

We count 12 bytes from the start byte of the message, `17`, marked red in the screenshot. In the end, we reach the 13th byte, `05`, the beginning of `image.google.com`, marked in yellow.

Not complicated, right? Here comes a complex one.



In the 2nd entry of answers, the `Name` offset is `2e`, 46 bytes.

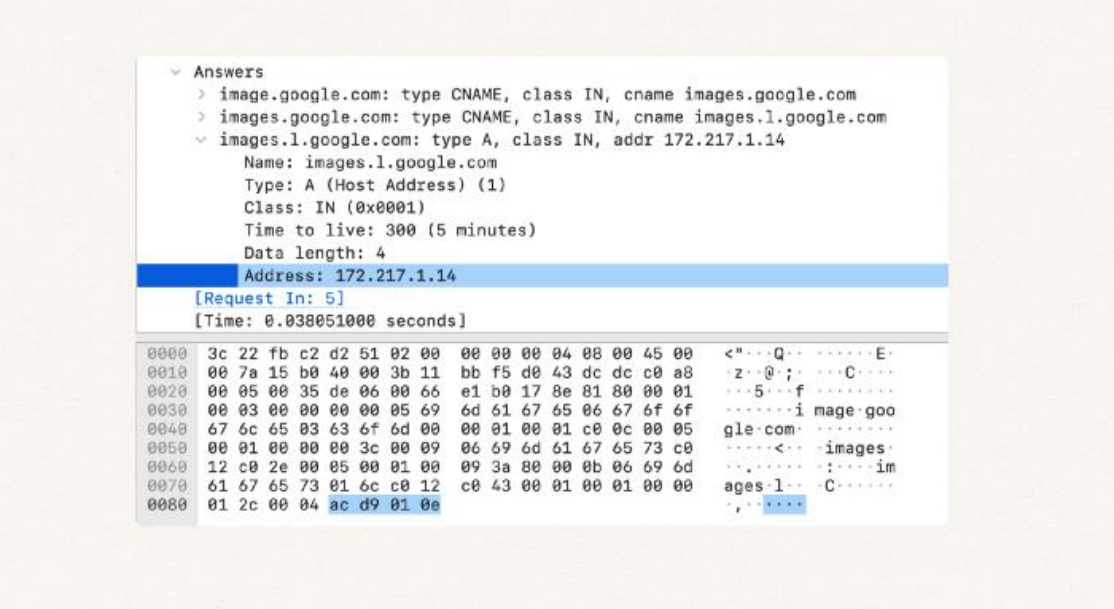
By counting 46 bytes, we find the encoded `images` in the CNAME of the previous entry marked in yellow:

- `06` → the following 6 bytes are in the same group
- `69` → i
- `6d` → m
- `61` → a
- `67` → g
- `65` → e
- `73` → s

At the end of the images, we recognize another offset reference `c0 12`. That's 18 bytes.

Again, by counting 18, we reach the referred part marked in green — `google.com` in the `Name` of preview entry.

The offset idea is an inspiring design. With it, the message saves considerable space.



Finally, we can decode the address in the last answer entry:

- `ac` → 172
- `d9` → 217
- `01` → 1
- `0e` → 14

DNS Registrars

A **domain name registrar** is a company or organization whose purpose is to conduct the registration of domain names. Popular examples include GoDaddy, Dreamhost, Namecheap, and Network solutions.

What Does Domain Name Registrar Mean?

A domain name registrar is company that has been accredited by the Internet Corporation for Assigned Names and Numbers (ICANN) or a national country code top-level domain (TLD) (such as .uk or .ca) to register domain names. Domain name registration is a competitive industry, in which domains may be sold in a number of TLDs, including ".com," ".net," and ".org." among others.

Suppose that an enterpreneur wants to register a domain through which he intends to sell inexpensive laptop computers. The enterpreneur might approach a domain name registrar to register the domain "laptopsforcheap.com." If the domain name hasn't already been registered to another person, the entrepreneur can register it and gain the right to use it by paying the registrar a yearly fee to secure the space.

DDNS

What is DDNS?

The Dynamic Domain Name System (DDNS) is a protocol that provides DNS extensions that allow DNS servers to accept requests to dynamically add, update, and delete entries in the DNS database.

- A DDNS server can serve both static and dynamic domains at the same time, since DDNS is a functional superset of existing DNS servers.
- Rather than allowing any server to change its DNS records, the secure version of DDNS authenticates update requests from DDNS hosts using critical public security and digital signatures.
- Dynamic DNS was created to address the problem of frequent IP changes. For example, when you search for a domain name, you'll get a dynamic IP address mapped to that domain. The Internet Service Provider (ISP) provides this dynamic IP address.
- When the same domain is searched again later, the ISP may be given a different IP address from the IP address pool, resulting in a different IP address being returned.
- When the IP address changes, the DDNS system refreshes the DNS database, which is always up to date with the domain-IP mapping. The outside world will be able to access the domain name at all times without having to worry about IP changes.

Applications of DDNS

Since domain controllers register their network service types in DNS so that other computers in the domain (or forest) can access them, dynamic DNS is an essential aspect of Active Directory in Microsoft Windows networks.

DDNS for Internet Access Devices

- Dynamic DNS providers provide a software client programme that automates discovering and registering public IP addresses for the client system. On a computer or device in the private network, the client programme is run. It uses a unique login name to connect to the DDNS provider's systems.
- The provider uses the name to associate the found public IP address of the home network with a domain name system hostname.

For Security

- For IP-based security products like DVRs and IP cameras, dynamic DNS is an expected feature, if not a requirement. The usage of current DDNS services or new services hosted by the manufacturer are only two of the alternatives available to manufacturers these days.
- A simple HTTP-based update API is nearly always utilised because it enables straightforward integration of a DDNS client into a device's firmware.
- **MintDNS, cURL,** and **Inadyn** are just a few examples of pre-made tools to help with server and client development.
- Most web-based DDNS providers use a common username and password security scheme. A user must first create an account on the DDNS server website, after which they must set up their device to submit updates to the DDNS server anytime an IP address change is detected.

Benefits of Using a DDNS

Following are some of the benefits of using a dynamic DNS

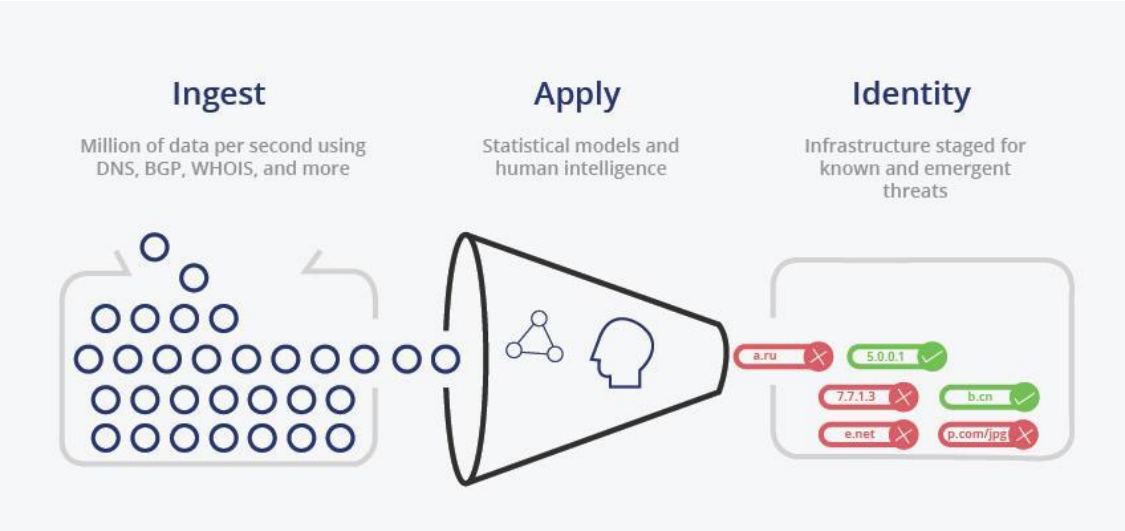
- You can visit your website or server from any location on the planet without worrying about your IP address changing. A device on your network sends your IP address to the DDNS service regularly.
- Your network administrators will save time by not having to update settings with each IP address change, allowing them to focus on the health of your network.
- You won't have to manually update all of your records whenever your IP address changes. In the long term, DDNS is less expensive than static DNS.

What is DNS Security?

Network attacks are increasingly targeting DNS. A DNS has the reputation of being one of the oldest and most relied-upon protocols on the Internet, making it a viable target for attackers. To ensure that DNS infrastructure can continue functioning quickly and reliably, it is important to use secure DNS tools to protect it against cyberattacks.

Why DNS Security is Important?

The DNS system has several design limitations, like many Internet protocols. Including spoofing, amplification, DoS (Denial of Service), or the interception of private information, these limitations, together with advancements in technology, make DNS servers vulnerable to a wide range of attacks. Therefore, organizations should handle DNS security issues carefully since it is integral to most internet requests.



Importance of DNS Security

Without DNS security, cybercriminals can easily identify security vulnerabilities and redirect a domain name to their desired location. It is unimaginable how uncomfortable it would be if we couldn’t access our company website because of

an attack. The DNS threat can corrupt an online banking system and steal confidential consumer information. Thus, DNS Security is among the most critical cyber security tools, and organizations must take DNS security issues seriously.

Types of DNS Security Threats

1. Typosquatting

Social engineering attacks such as typosquatting target internet users who type URLs incorrectly. For instance, the attacker requested to click on https://www.aplle.com/ instead of https://www.apple.com. When URLs misspell the original/authentic websites, users are typically tricked into visiting malicious websites. These fake sites trick users into entering sensitive information and have the potential to do significant damage to organizations by stealing sensitive information.

2. Distributed Denial of Service Attacks (DDoS)

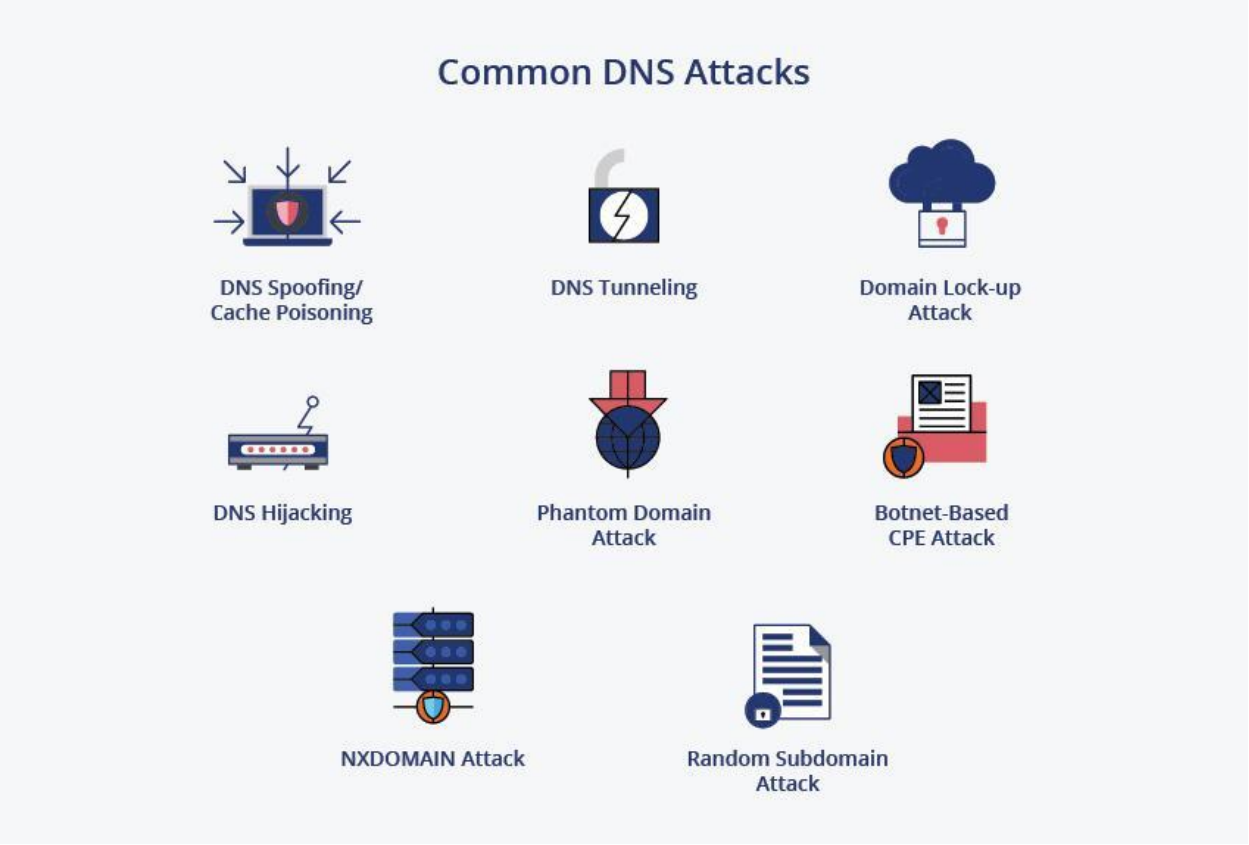
Multiple connected online devices, collectively known as botnets, are used in DDoS attacks to overwhelm a target website.

3. DNS Cache Poisoning

Users are sent to the wrong website due to DNS cache poisoning, which creates incorrect DNS responses and inaccurate DNS cache entries.

4. DNS Amplification

In DNS Amplification, attackers amplify their traffic using open DNS resolver functionality to dominate a target server or network with a large traffic volume, thereby rendering it inaccessible to the attacker. This is an effective distributed reflection-based DDoS (denial-of-service) attack.



Best Domain Name System (DNS) Security Tools

DNS security tools are designed to prevent cyber attacks by acting as an intermediary between the web browser and the websites the user is trying to access online. In addition to securing public DNS, its security tools eliminate botnet servers, filter content such as advertising or adult websites and fix typos in domain names. Below are the top six best DNS security tools to use:

1. Cisco Umbrella

Cisco Umbrella offers DNS security tools through its cloud server. A single DNS solution integrates multiple security functions, protecting devices, remote users, and distributed locations. It takes minutes to install the Cisco Umbrella DNS security tool and secure the user's data to max level. Cisco Umbrella reports provide information about the activity of each device or network within the system.

2. TitanHQ WebTitan

In terms of protecting against web-based cyber threats such as malicious websites, malware, or ransomware, TitanHQ Web Filter stands out among DNS-based security solutions. Global enterprises trust TitanHQ Web Filter because its API set allows for advanced control over web and DNS filtering, allowing them to filter web traffic and DNS traffic. Additionally, the system provides real-time automated detection and blocking of malicious threats.

3. Infoblox BloxOne Threat Defense

In addition to detecting risks, Infoblox BloxOne Threat Defense helps you prevent attacks early in their lifecycle. This solution enhances security stack effectiveness, safeguards digital operations, and lowers cybersecurity costs by integrating universal automation and ecosystems. Infoblox BloxOne Threat Defense includes DNS protection tools to help users protect their systems in hybrid workplaces with visibility, command, and automation.

4. F5 BIG-IP DNS

F5 BIG-IP DNS distributes DNS and application requests according to company regulations and data center and cloud service conditions, as well as the user's location and performance. As a complete proxy, F5 BIG-IP DNS can be configured across architectures and globally to supply global server load balancing for applications and DNS.

5. Palo Alto Networks DNS Security

The user can automatically prevent phishing attacks by enabling URL Filtering in Palo Alto Networks DNS security, including links in online ads, emails, SMS links, websites, HTTP-based command and control, and malicious sites. Palo Alto Networks DNS security is a great pick for SMEs (small-medium enterprises).

6. Infoblox Advanced DNS Protection

The Infoblox ADP blocks various attacks, including DNS hijackings, volumetric attacks and NXDOMAINs. In addition to detecting and mitigating DNS attacks, The Infoblox ADP does not require security patches because it uses constantly updated threat intelligence.

What are Some Common Attacks Involving DNS?

1. DNS Spoofing/Cache Poisoning

The DNS cache poisoning method is also called DNS spoofing. The purpose of DNS spoofing is to redirect organic traffic from a legitimate server to a fake server by exploiting vulnerabilities in the DNS.

2. DNS Tunnelling

In most organizations, DNS is used freely within and outside their networks since it is considered a trusted protocol. Cyber crooks exploit DNS for data exfiltration using malware that contains the data being exfiltrated in DNS requests. Attackers ensure that the data users are transmitting in the DNS response packet reaches a server controlled by them, not by the website owner.

3. DNS Hijacking

Users are fooled into believing they are connected to a legitimate domain when they are connected to a malicious one by DNS hijackers. DNS servers can be compromised to store incorrect data by using malicious or compromised DNS servers.

4. NXDOMAIN Attack

Clients are prevented from accessing the roadmap through the DNS NXDOMAIN flood attack. DNS servers are swamped with invalid or nonexistent requests when this attack occurs.

5. Phantom Domain Attack

Phantom domain attacks are types of DoS attacks that target authoritative nameservers. An attack is conducted by setting up several DNS servers that fail to respond to DNS requests or do so sluggishly, disrupting communication.

To find an IP address, a DNS server searches the addresses of other DNS servers connected to it; this process is known as recursive DNS. Attacks against phantom domains result in inefficient lookups or non-functional searches and waste of server resources.

When recursive DNS servers fully consume resources, they can cause serious performance problems by ignoring legitimate queries and focusing on non-responsive servers.

6. Random Subdomain Attack

Random subdomain attacks are similar to NXDOMAIN attacks except that they ask for nonexistent subdomains instead of nonexistent domains.

7. Domain Lock-up Attack

A DNS resolver is locked up by these attacks, as their name implies. This is accomplished by connecting to a resolver with TCP, and then allowing domains to send randomly generated junk packets, which overwhelm a resolver.

8. Botnet-based CPE Attack

These attacks are developed by exploiting devices such as modems, routers, cable boxes, etc., used as CPE (Customer Premise Equipment). The attackers compromise CPEs, and the devices are made part of a botnet that attacks one or more sites or domains at random.

Measures Against DNS Attacks

- One can protect private data with digital signatures and certificates.
- One should do a DNS zone review regularly. It's easy to forget about checking domain names or subdomains that may run outdated software or expose unrestricted areas to attackers as time goes on.
- To ensure that the A, CNAME, and MX records are accurate, reviewing all your zones, records, IPs, and SSL certificates is crucial.
- Make sure to use the latest Hide BIND version software. Several features are available in BIND, including DNSSEC, DNSTAP, Scaleable Primary-Secondary Hierarchy, Minimal ANY Responses, and many more. Major DNS servers on the Internet use BIND software.
- Maintain regular software check-ups and fix faulty bugs as soon as they arise.
- Make backups of data on different servers; in case of corruption or loss on one server, other servers can restore the data.
- Ensure the data centers are connected to various networks; it helps in reducing the risk of single-point failures.
- Ensure that your DNS configuration is as secure as possible. The domain names that need to be resolved can be randomly cased, and one can randomly generate the query ID instead of the standard DNS port.

Best DNS Servers for Security

You can improve and use secure dynamic DNS by switching DNS providers. With so many things on the Internet involving DNS requests, choosing the fastest DNS directory across all your devices and securing public DNS will allow you to do almost everything faster. The top three recommended best DNS servers enhance your online security.

1. Cloudflare DNS

There is no doubt that Cloudflare DNS is an excellent DNS protection service worldwide. Newly, Cloudfare DNS built 1.1.1.1, the fastest DNS service globally.

You cannot restrict what websites you visit with Cloudflare DNS protection, but your privacy comes first. Cloudflare does not log your DNS traffic or your IP address. The Cloudflare DNS filtering automatically deletes everything logged within 24 hours. Thus, Cloudfare DNS makes the best choice among many enterprises.

Recently, in response to a massive DDoS attack against Minecraft, *Cloudflare mitigated the damage*.

2. OpenDNS

A popular DNS server offers free, public, web-based DNS servers. Open DNS is a popular choice among many enterprises and is used by millions of people. Cisco has owned OpenDNS since 2016.

When it comes to protecting yourself from malicious attacks, OpenDNS is a reliable choice. You can speed up page loading by connecting to the nearest DNS server through anycast routing. In addition to high-speed internet transfer, OpenDNS also blocks phishing websites, filters adult web content, and records your online activity for a year.

OpenDNS offers three services in their Home package, two of which are free: OpenDNS Family Shield and OpenDNS Home. Aside from not recording internet activity history and not providing access to specific websites, it offers all the same features as the paid version.

3. Google Public DNS

The speed of Google DNS is its greatest advantage. In addition to global coverage, Google DNS is DNSSEC-encrypted as standard; it has load balancing and shared caching to improve cache hit rates. Using Google Public DNS will make your browsing experience faster and more secure, and you won’t be redirected.

Key Tips for Maintaining DNS Security

1. DNS Security Extensions or DNSSEC

A pair of public keys is attached to every DNS zone and digital signatures are generated over DNS data using the owner's key. Keeping this key material secured is the responsibility of the owner.

In DNSSEC, private key cryptography strengthens authentication through digital signatures. DNSSEC does not cryptographically sign DNS queries or responses but rather certificates that certify DNS data directly from the owner.

2. Encrypting Data

The encryption of data included in DNS requests and responses provides an enhanced level of security. A layer of security can be provided by encrypting data to prevent hackers from intercepting or inserting malware.

3. Implementing Secure DNS Configurations

DNS servers can be configured in a secret vault without being connected to other DNS servers inside an organization. As a result, two DNS servers cannot establish a connection with each other. It is, therefore, unlikely that other servers will be affected if one server is compromised.

Furthermore, limiting the amount of data that is stored on each server can be achieved with secure DNS configurations. An encrypted configuration helps prevent data from being compromised on a broader scale.

4. Running Regular System Updates

Updates to DNS servers are scheduled regularly. Keeping these updates up-to-date is crucial. These updates introduce new security protocols that enable the servers to identify and fix vulnerabilities before they impact other systems.

5. Strengthening Detection Protocols

Increased DNS activity about a particular domain from a single source is a critical warning sign of malicious activity. In addition, the number of domain names encountered by one source can increase when there is an attempt to enter the DNS server for spoofing. Thus, monitoring and preventing malware attacks requires strong detection protocols.

6. Security Training

DNS servers require robust security training, which is mandatory in most organizations today. Using safe practices when interacting with the internet becomes easier for users when they know the potential risks. One should include several key techniques in training:

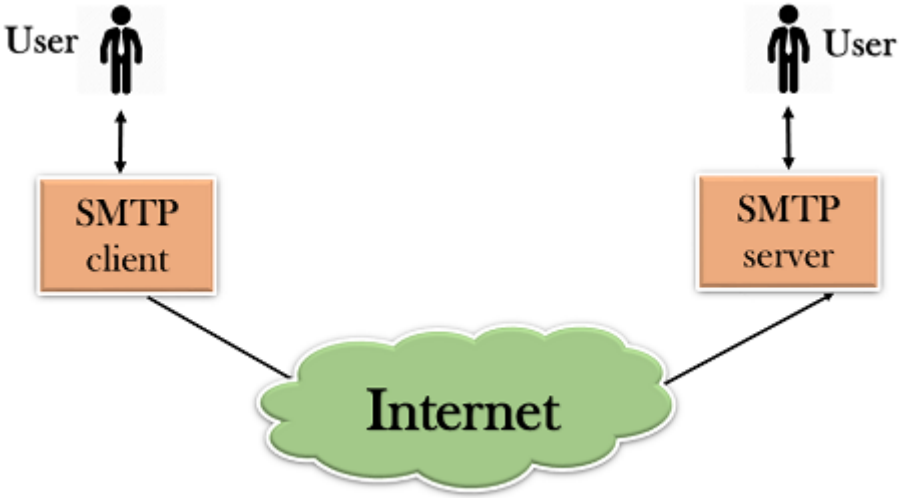
- *Checking the SSL (Secure Sockets Layer) or TLS (Transport Layer Security certificates) of websites you access.*
- *Be cautious when clicking unrecognized links.*
- *Run security checks as soon as the system requests them. One should not delay the process. Delay makes the system more vulnerable.*

SMTP

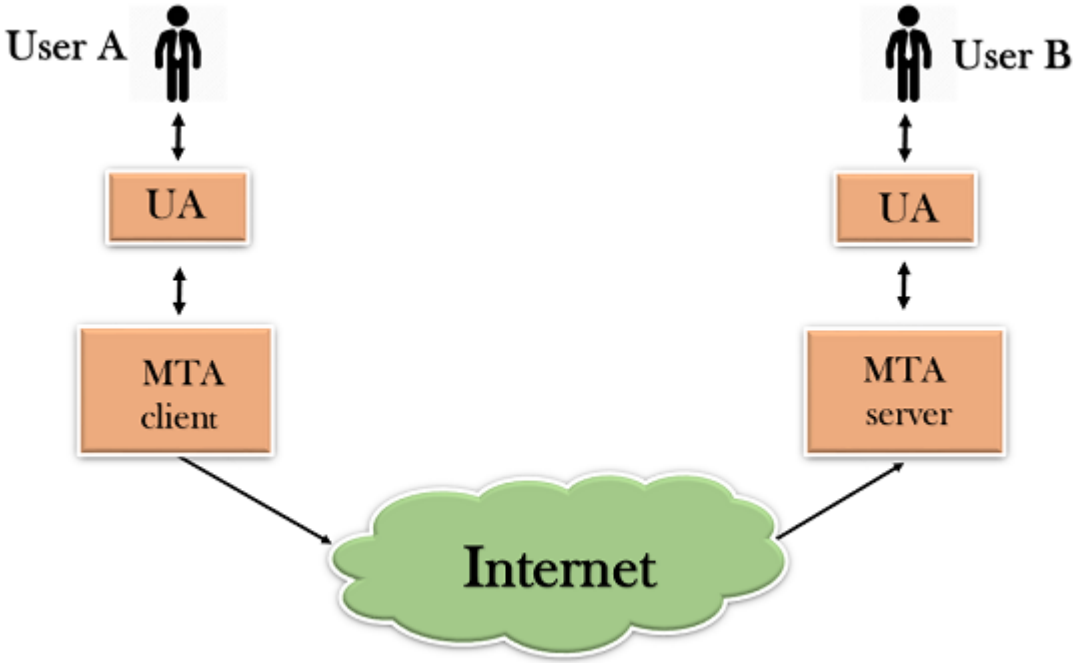
- SMTP stands for Simple Mail Transfer Protocol.
- SMTP is a set of communication guidelines that allow software to transmit an electronic mail over the internet is called **Simple Mail Transfer Protocol**.
- It is a program used for sending messages to other computer users based on e-mail addresses.
- It provides a mail exchange between users on the same or different computers, and it also supports:
    - It can send a single message to one or more recipients.
    - Sending message can include text, voice, video or graphics.
    - It can also send the messages on networks outside the internet.
- The main purpose of SMTP is used to set up communication rules between servers. The servers have a way of identifying themselves and announcing what kind of

communication they are trying to perform. They also have a way of handling the errors such as incorrect email address. For example, if the recipient address is wrong, then receiving server reply with an error message of some kind.

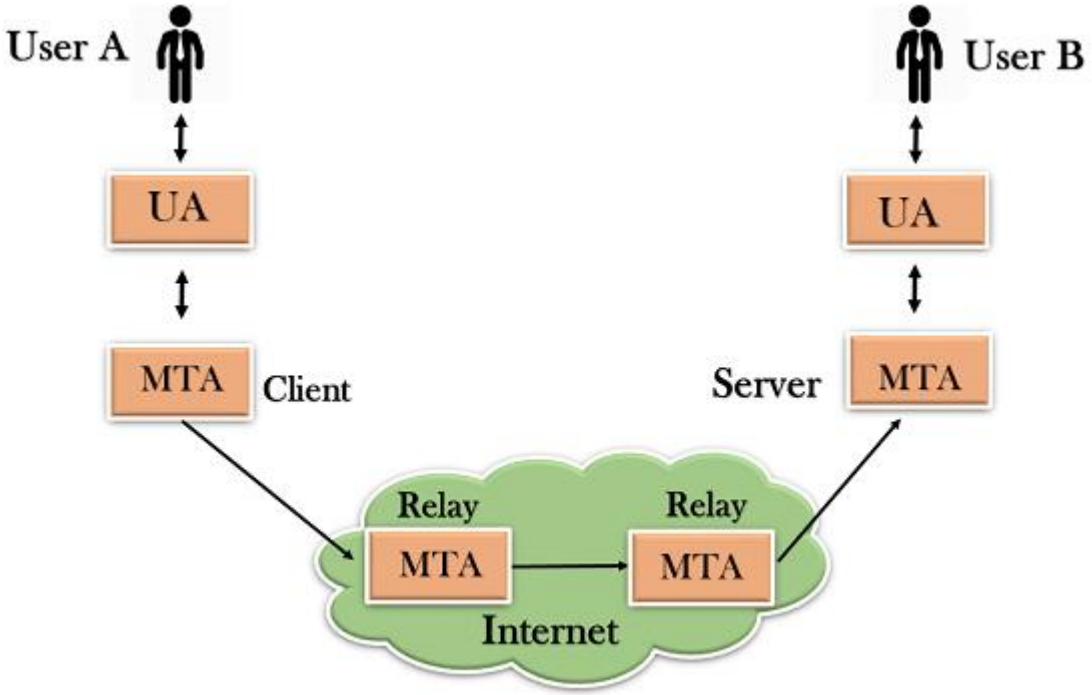
Components of SMTP



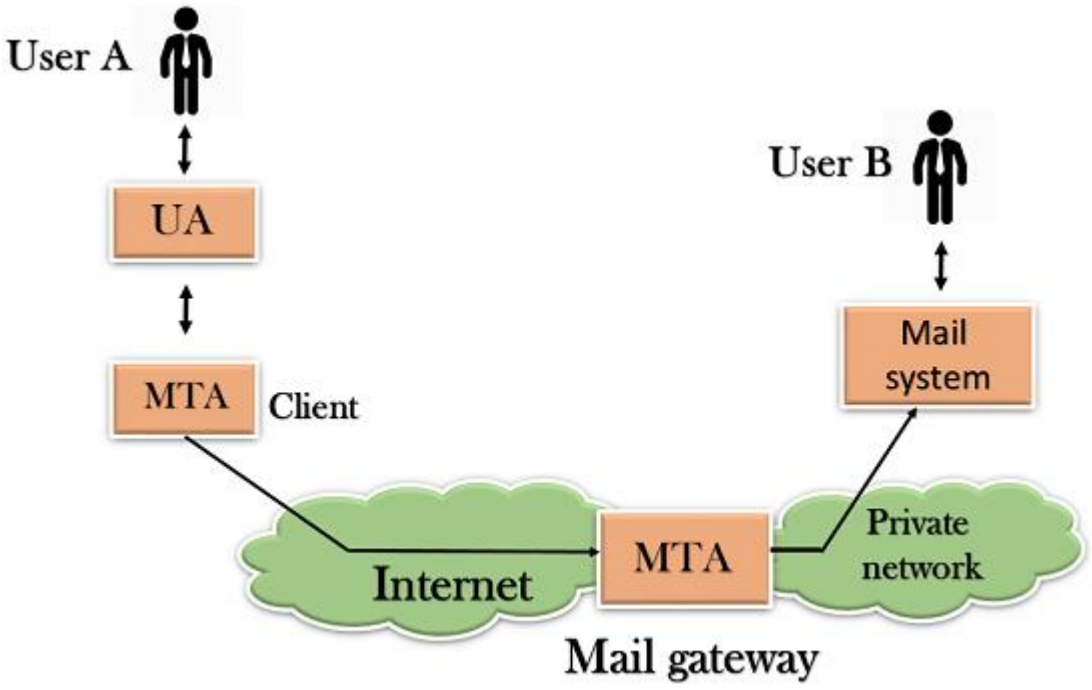
- First, we will break the SMTP client and SMTP server into two components such as user agent (UA) and mail transfer agent (MTA). The user agent (UA) prepares the message, creates the envelope and then puts the message in the envelope. The mail transfer agent (MTA) transfers this mail across the internet.



- SMTP allows a more complex system by adding a relaying system. Instead of just having one MTA at sending side and one at receiving side, more MTAs can be added, acting either as a client or server to relay the email.



- The relaying system without TCP/IP protocol can also be used to send the emails to users, and this is achieved by the use of the mail gateway. The mail gateway is a relay MTA that can be used to receive an email.



Working of SMTP

1. **Composition of Mail:** A user sends an e-mail by composing an electronic mail message using a Mail User Agent (MUA). Mail User Agent is a program which is used to send and receive mail. The message contains two parts: body and header. The body is the main part of the message while the header includes information such as the sender and recipient address. The header also includes descriptive information such as the subject of the message. In this case, the message body is like a letter and header is like an envelope that contains the recipient's address.
2. **Submission of Mail:** After composing an email, the mail client then submits the completed e-mail to the SMTP server by using SMTP on TCP port 25.
3. **Delivery of Mail:** E-mail addresses contain two parts: username of the recipient and domain name. For example, vivek@gmail.com, where "vivek" is the username of the recipient and "gmail.com" is the domain name.
   If the domain name of the recipient's email address is different from the sender's domain name, then MSA will send the mail to the Mail Transfer Agent (MTA). To relay the email, the MTA will find the target domain. It checks the MX record from Domain Name System to obtain the target domain. The MX record contains the domain name and IP address of the recipient's domain. Once the record is located, MTA connects to the exchange server to relay the message.
4. **Receipt and Processing of Mail:** Once the incoming message is received, the exchange server delivers it to the incoming server (Mail Delivery Agent) which stores the e-mail where it waits for the user to retrieve it.
5. **Access and Retrieval of Mail:** The stored email in MDA can be retrieved by using MUA (Mail User Agent). MUA can be accessed by using login and password.

Multipurpose Internet Mail Extension (MIME) Protocol

Multipurpose Internet Mail Extension (MIME) is a standard that was proposed by Bell Communications in 1991 in order to expand the limited capabilities of email.
MIME is a kind of add-on *or a supplementary protocol* that allows non-ASCII data to be sent through SMTP. It allows the users to exchange different kinds of data files on the Internet: audio, video, images, application programs as well.

Why do we need MIME?
Limitations of Simple Mail Transfer Protocol (SMTP):
1. SMTP has a very simple structure
2. Its simplicity however comes with a price as it only sends messages in NVT 7-bit ASCII format.
3. It cannot be used for languages that do not support 7-bit ASCII format such as French, German, Russian, Chinese and Japanese, etc. so it cannot be transmitted using SMTP. So, in order *to make SMTP more broad, we use MIME*.
4. It cannot be used to send binary files or video or audio data.

Purpose and Functionality of MIME –
Growing demand for Email Messages as people also want to express themselves in terms of Multimedia. So, MIME another email application is introduced as it is not restricted to textual data.
MIME *transforms non-ASCII data* at the sender side to NVT 7-bit data and delivers it to the client SMTP. The message on the receiver side is transferred back to the original data. As well as we can send video and audio data using MIME as it transfers them also in 7-bit ASCII data.

Features of MIME –
1. It is able to send multiple attachments with a single message.
2. Unlimited message length.
3. Binary attachments (executables, images, audio, or video files) may be divided if needed.
4. MIME provided support for varying content types and multi-part messages.

Working of MIME –
Suppose a user wants to send an email through a user agent and it is in a non-ASCII format so there is a MIME protocol that converts it into 7-bit NVT ASCII format. The message is transferred through the e-mail system to the other side in the 7-bit format now MIME protocol again converts it back into non-ASCII code and now the user agent of the receiver side reads it and then information is finally read by the receiver. MIME header is basically inserted at the beginning of any e-mail transfer.

MIME with SMTP and POP –
SMTP transfers the mail being a message transfer agent from the sender's side to the mailbox of the receiver side and stores it and MIME header is added to the original header and provides additional information. while POP being the message access agent organizes the mails from the mail server to the receiver's computer. POP allows the user agent to connect with the message transfer agent.

MIME Header:
It is added to the original e-mail header section to define transformation. There are *five headers* that we add to the original header:
1. **MIME-Version –** Defines the version of the MIME protocol. It must have the parameter *Value 1.0*, which indicates that message is formatted using MIME.

2. **Content-Type –** Type of data used in the body of the message. They are of different types like text data (plain, HTML), audio content, or video content.
3. **Content-Type Encoding –** It defines the method used for encoding the message. Like 7-bit encoding, 8-bit encoding, etc.
4. **Content Id –** It is used for uniquely identifying the message.
5. **Content description –** It defines whether the body is actually an image, video, or audio.

Concept of web caching & proxy servers.

Web caching is done by a Proxy Server – an intermediate entity between the original server and the client. When a client requests for some information (through an HTTP message), it goes through the proxy server, which –

- First checks if it has the copy locally stored.
- If it has, then it forwards the result directly to the client.
- Otherwise, it queries on behalf of the end host, stores a copy of the result locally, and forwards the result back to the end host.

Web caches (or) Proxy Servers are usually installed by ISPs (Internet service providers), Universities, or even Corporate Offices, wherein multiple end hosts are connected to the proxy server.

Installing a proxy server has multiple advantages –

1. It can substantially reduce the response time for repeated requests. (Especially if the bottleneck between the original server and receiver is less than bottleneck between the proxy server and receiver.)
2. It reduces the access link bandwidth (of the university or the office), thereby reducing the cost.
3. It reduces traffic on the Internet as a whole.

But there is one problem.

What if the content was modified on the original server, rendering the copy on the proxy server to be an outdated one?

This is where Conditional GET statements kicks in. When a Proxy server receives an HTTP request, and it has the result stored locally, it still queries the original server, asking if that particular object was modified since the last time it was requested by the proxy server.

The “Conditional GET” statement has an additional field than a normal GET statement, called the “If-modified-since” field, which specifies the last time when the same request was made. The original server either –

- Tells the proxy server that the content was not modified – HTTP 304 status code, or
- Sends the updated content (in case there was some modification done) – HTTP 200 response-message code

If the Proxy server gets a 304 – "No Modification" message, it forwards its local copy to the client. If modification had been there, the Cache forwards the new object, whilst storing it locally along with the date and time it received the new object (so that it can ask the original server later for modifications).
For obvious reasons, an HTTP 304 message does not contain a message body.

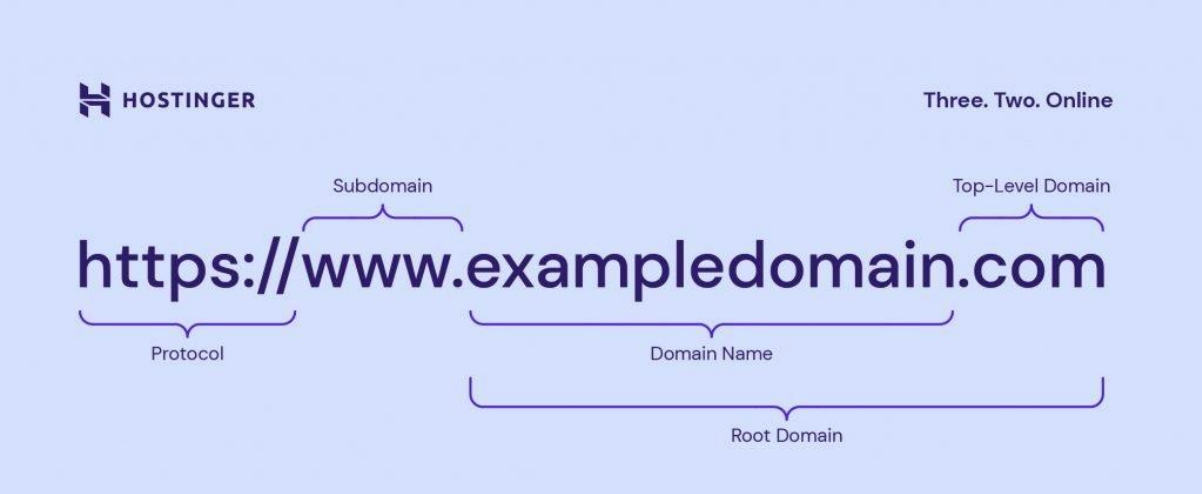
Q) In DNS, which of the following are FQDNs and which are PQDNs? Give few examples of each.

What is FQDN?

A fully-qualified domain name (FQDN) is a complete domain name that specifies the exact location of a computer or a host on the internet. It consists of the hostname and domain name. In addition, a FQDN can be found through terminal on MacOS and Linux or through the advanced system settings on Windows.

Examples of an FQDN

A fully qualified domain name, also known as an absolute domain name, specifies all domain levels written in the **hostname.domain.tld** format. For example, a mail server of yahoo.com would be **mail.yahoo.com**. Other examples include **www.wordpress.org** and **news.bbc.co.uk**.
Also you may want to read our article explaining **what a domain name is**.



Let’s explore each element on an FQDN hierarchy:

- **Hostname.** It is a label assigned to a server service available on a network. A **DNS** server uses a hostname to make an **IP address** easy to remember. Examples of a hostname are **“www”** in **www.hostinger.com** and **“en”** in **en.wikipedia.org** .
- **Subdomain.** This part is located on the left side of a second-level domain and sometimes indicates a section of a larger domain. For instance, **support.hostinger.com** is a part of **hostinger.com**, and the word **“support”** is the subdomain. Note that not all domains have this element.
- **A domain name.** It consists of a second-level and a **top-level-domain (TLD)**. For example, in **hostinger.com**, **“hostinger”** is the second-level domain, and **“.com”** is the TLD.

The maximum hostname and fully qualified domain name length is **63 bytes per label** and 255 bytes per FQDN.

What Is a Partially Qualified Domain Name?

Like an FQDN, a partially qualified domain name (PQDN) also indicates a web address but only includes a hostname or a domain name. Generally, website developers configure their sites' DNS to redirect visitors to the same page whether they type the FQDN or PQDN.
For example, if you type **hostinger.com** or **www.hostinger.com**, you will land on **https://hostinger.com** – the URL for our homepage. PQDNs are usually favorable since they are shorter, helping visitors find the site easier.

Differentiate between FQDN and PQDN

The difference between FQDN and PQDN

FQDN

A fully qualified domain name (FQDN) is the complete domain name for a specific computer, or host, on the Internet. The FQDN consists of two parts: the hostname and the domain name. For example, an FQDN for a hypothetical mail server might bemymail.somecollege.edu. The hostname is my mail, and the host is located within the domainsomecollege.edu.

PQDN

If a label is not terminated by a null string, it is called a partially qualified domain name (PQDN). A PQDN starts from a node, but it does not reach the root. It is used when the name to be resolved belongs to the same site as the client. Here the resolver can supply the missing part, called suffix, to create an FQDN.