Computer Networking
How things work

Name
Dicha Zelianivan Arkana

NIM
2241720002

Class
2i

Department
Information Technology

Study Program
D4 Informatics Engineering

Contents

1   Introduction

The internet that we use everyday comprises of many different devices and technologies. Each of those devices communicate through different types of protocols, which have they're own purpose and use case. This article will explain how those devices communicate with each other through the aforementioned protocols.

This article is in no way tries to be a comprehensive guide over these protocols. It should serve as an introduction to the topic and should be used as a starting point for further research. Some of the protocols that will be discussed in this article are:

- DNS

- HTTP

- Email Services (SMTP / POP3 / IMAP)

- FTP

- DHCP

- Telnet/SSH Services

2   Domain Name System (DNS)

Domain Name System or commonly abbreviated as DNS is a system that translates domain names to IP addresses. Think of it like a phonebook where it converts a name to a phone number. This system is crucial for the internet to function as it is today. Without DNS, we would have to remember the IP address of every website we want to visit.

In order to understand the process behind the DNS resolution, it's important to learn about the different hardware components a DNS query must pass between. For the web browser, the DNS lookup occurs "behind the scenes" and requires no interaction from the user's computer apart from the initial request.

The process of resolving a domain name involves the following steps:

1. A user types 'example.com' into a web browser and the query travels into the Internet and is received by a DNS recursive resolver.

2. The resolver then queries a DNS root nameserver

3. The root server then responds to the resolver with the address of a Top Level Domain (TLD) DNS server (such as .com or .net), which stores the information for its domains. When searching for example.com, our request is pointed toward the .com TLD.

4. The resolver then makes a request to the .com TLD.

5. The TLD server then responds with the IP address of the domain's nameserver, example.com.

6. Lastly, the recursive resolver sends a query to the domain's nameserver.

7. The IP address for example.com is then returned to the resolver from the nameserver.

8. The DNS resolver then responds to the web browser with the IP address of the domain requested initially.

Once the 8 steps of the DNS lookup have returned the IP address for example.com, the browser is able to make the request for the web page:

1. The browser makes a HTTP request to the IP address.

2. The server at that IP returns the webpage to be rendered in the browser.

All of these steps happen in a fraction of a second. The DNS system is a critical component of the Internet infrastructure, providing a way for your computer to look up the addresses of the servers where your requested pages are hosted.

3   Hypertext Transfer Protocol (HTTP)

Hypertext Transfer Protocol, more commonly known as HTTP is probably the most widely known protocol on the Internet. It's basically the foundation of World Wide Web, and is used to load webpages using hypertext links. It operates in the application layer protocol designed to transfer information between networked devices on top of other layers of network protocol stack. A typical flow of HTTP communication involves a client and server devices.

HTTP follows a classical client-server model, with a client opening a connection to make a request, then waiting until it receives a response. HTTP is a stateless protocol, meaning that the server does not keep any data (state) between two requests.

When we make a HTTP request to a server, it typically contains several things:

1. **HTTP Version Type** - As of the time when this article was written, there are 3 versions of HTTP that are widely used. Namely, HTTP/1.1, HTTP/2.0, and HTTP/3.0

2. **URL** - Uniform Resource Locator (URL) is an address to a resource on the Internet. It can be webpage, image, video, or any other file.

3. **HTTP Method** - Also known as HTTP verb. Each request that performs different action carries different HTTP verb. The most common HTTP verbs are GET, POST, PUT, DELETE, and PATCH.

4. **HTTP Request Headers** - These are key-value pairs that are sent to the server along with the request. This part functions as a way to send additional information to the server that acts like a metadata such as the user agent, format of the data being sent, etc.

5. **HTTP Request Body (optional)** - This part is optional and is only used when the request method is POST, PUT, or PATH. The body of an HTTP request contains any information being submitted to the web server, such as a username and password, or any other data entered into a form.

After making a request, we usually get a HTTP response back from the server that consists of:

1. **HTTP Status Code** - It's a 3 digit numbers used to indicate whether a HTTP request is successful or not. They are broken into 5 blocks:

   - **1xx** - Informational
   - **2xx** - Success
   - **3xx** - Redirection
   - **4xx** - Client Error
   - **5xx** - Server Error

   The xx refers to different numbers between 00 - 99.

2. **HTTP Response Headers** - Much like its request counterparts, these are key-value pairs that are sent from the server to the client and it serves pretty much the same purpose as the request headers.

3. **HTTP Response Body** - It contains the actual content that the server sends back to the client. Most commonly it's a HTML file, but it can also be a JSON, XML, or any other format.

4 Email Services (SMTP / POP3 / IMAP)

4.1 Simple Mail Transfer Protocol (SMTP)

SMTP stands for Simple Mail Transfer Protocol. It is a communication protocol used for sending and receiving email messages over the Internet. Mail servers and other message transfer agents (MTAs) use SMTP to send, receive and relay mail messages.

4.2   Post Office Protocol version 3 (POP3)

POP3 stands for Post Office Protocol version 3. It is a standard mail protocol used to receive emails from a remote server to a local email client. POP3 allows you to download email messages on your local computer and read them even when you are offline. This protocol works locally and is used to retrieve emails from the server. Since it is a one-way protocol, it doesn't allow syncronisation across multiple devices.

4.3   Internet Message Access Protocol (IMAP)

IMAP stands for Internet Message Access Protocol. It is a standard protocol for accessing e-mail from your local server. IMAP is a two-way protocol that allows you to access your email from multiple devices and maintains the same state across all devices. This means that if you read an email on your phone, it will also be marked as read on your computer.

5   File Transfer Protocol (FTP)

FTP stands for File Transfer Protocol. It is a standard network protocol used to transfer files from one host to another host over a TCP-based network, such as the Internet. FTP is built on a client-server architecture and uses separate control and data connections between the client and the server.

FTP works by sending commands and receiving responses through the control connection. When a user requests a file, the server sends a file transfer request to the client. The client then opens a data connection to the server and the file is transferred using the data connection. Once the file transfer is complete, the data connection is closed and the control connection remains open.

6   Dynamic Host Configuration Protocol (DHCP)

DHCP stands for Dynamic Host Configuration Protocol. It is a network management protocol used to automate the process of configuring devices on IP networks. DHCP allows hosts to obtain required TCP/IP configuration information from a DHCP server. Usually this information includes IP address, subnet mask, default gateway, and DNS server addresses.

7 Telnet/SSH Services

Telnet and SSH are both network protocols that allow you to connect to a remote computer over a network.

7.1 Telnet

Telnet is a network protocol that allows a user to remotely access and control another computer over the Internet or local area network (LAN). It enables a user to establish a connection to a remote system and perform tasks as if they were sitting in front of that computer.

It is a client-server protocol, which means that a client device initiates the connection to a server device. The client sends commands to the server, and the server responds with output, allowing the user to interact with the remote system's command-line interface. It uses the Transmission Control Protocol (TCP) as its underlying transport protocol.

7.2 Secure Shell (SSH)

The Secure Shell (SSH) protocol is a method for securely sending commands to a computer over an unsecured network. SSH uses cryptography to authenticate and encrypt connections between devices. SSH also allows for tunneling, or port forwarding, which is when data packets are able to cross networks that they would not otherwise be able to cross. SSH is often used for controlling servers remotely, for managing infrastructure, and for transferring files.