Understanding the Domain Name System (DNS)

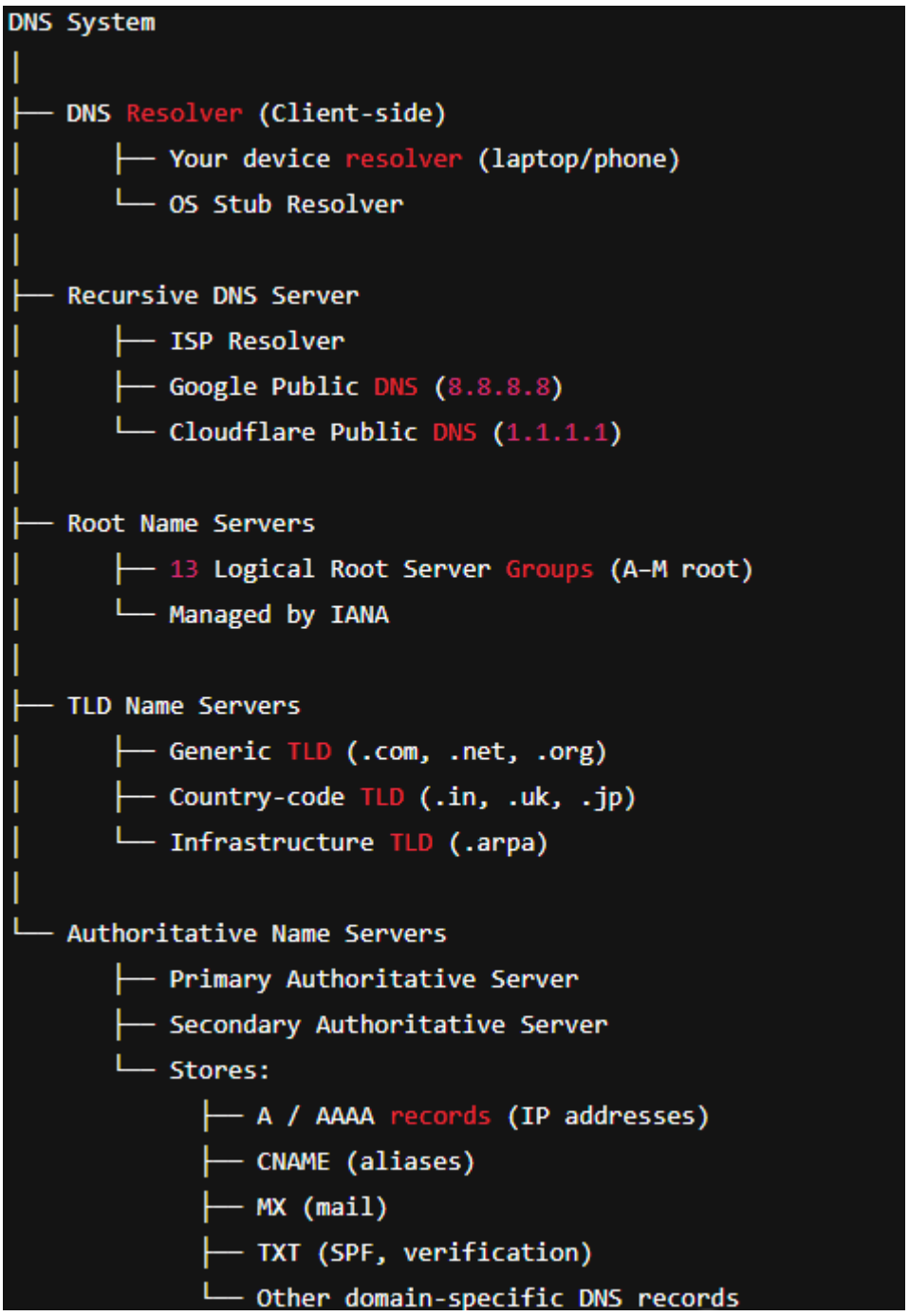
Introduction

The Domain Name System (DNS) is a foundational element of the internet Essentially, it translates human-readable domain names into machine-readable Internet Protocol (IP) addresses.Without DNS, users would have to remember complex numerical IP addresses for every website they wished to visit. DNS is not just one server. It is a family or hierarchy of servers located worldwide and works through 3 different terms: Zones- that refer to the parts of a domain, Delegation- pointer to the next server, Queries- are the request made to the server for getting the address.

Core Functionality: The Resolution Process

The primary function of DNS is name resolution, which is the process of finding the corresponding IP address for a given domain name. This process involves a collaboration between several types of DNS servers and clients (resolvers).

Key Components of the DNS Resolution Process



The resolution process typically involves four main types of servers:

1. **DNS Resolver:** This is usually the client device (e.g., your laptop or smartphone) or a local server on the network. It initiates the query and receives the final IP address.
2. **Recursive DNS Server (or Resolver):** This server, often provided by your Internet Service Provider (ISP) or a third-party service (like Google Public DNS or Cloudflare), is responsible for making all the necessary queries to find the IP address for the client. It caches results to speed up future requests.

3. **Root Name Server:** The top-most level of the DNS hierarchy. It does not know the IP address for the domain but directs the query to the appropriate Top-Level Domain (TLD) Name Server. There are 13 logical root server operators globally.
4. **TLD Name Server:** This server manages the information for specific TLDs (e.g., `.com`, `.org`, `.net`, or country codes like `.uk`, `.jp`). It directs the query to the Authoritative Name Server.
5. **Authoritative Name Server:** This server holds the actual DNS records (e.g., A records, CNAME records) for a specific domain (e.g., `example.com`). It provides the final IP address to the Recursive Resolver.

The Resolution Steps (A Typical Query)

Step	Initiator	Recipient
1	Client (Resolver)	Recursive Server
2	Recursive Server	Root Server
3	Root Server	Recursive Server
4	Recursive Server	TLD Server
5	TLD Server	Recursive Server
6	Recursive Server	Authoritative Server
7	Authoritative Server	Recursive Server
8	Recursive Server	Client (Resolver)

DNS Record Types

DNS relies on various record types, stored on Authoritative Name Servers, to direct traffic and provide information.

Record Type	Description	Purpose
A	Address Record	Maps a domain name to an IPv4 address. (The most common type for websites.)
AAAA	IPv6 Address Record	Maps a domain name to an IPv6 address.
CNAME	Canonical Name Record	Used to alias one domain name to another (e.g., `www.example.com` points to `example.com`).
MX	Mail Exchange Record	Specifies the mail servers responsible for accepting email on behalf of a domain.
TXT	Text Record	Used to hold arbitrary text strings, often for verification (e.g., SPF, DKIM, DMARC records for email security).
NS	Name Server Record	Indicates the authoritative name servers for the domain.
PTR	Pointer Record	Used for reverse DNS lookups (mapping an IP address back to a domain name).

Security and Performance Considerations

DNS Caching

Caching is crucial for DNS performance. Resolvers store previous lookup results for a specific duration defined by the **Time-To-Live (TTL)** value set on the DNS record. This reduces latency and load on higher-level servers by allowing the recursive server to answer a query directly from its cache.

Security Concerns and Solutions

Traditional DNS queries are often unencrypted, leading to potential security and privacy risks, including:

- **DNS Spoofing/Cache Poisoning:** An attacker injects false DNS data into a resolver's cache, redirecting users to malicious sites.
- **Eavesdropping:** Queries can be monitored by third parties to track browsing history.

Modern protocols address these issues:

- **DNSSEC (Domain Name System Security Extensions):** Adds cryptographic signatures to DNS data, ensuring the data is authentic and hasn't been tampered with.
- **DNS over HTTPS (DoH) and DNS over TLS (DoT):** Encrypts DNS queries between the client (or recursive resolver) and the server, preventing eavesdropping and tampering.

Practical Implementation

Let’s consider we are using Raspberry Pi for setting up our DNS Resolver.

1. First we run a quick command for Pi OS and package manager are up-to date
   Command: sudo apt update
   sudo apt upgrade -y

2. Next we install a recursive resolver with default configurations called Unbound.
   Command: sudo apt install unbound -y

3. We enable and start Unbound
   Command: sudo systemctl enable unbound
   sudo systemctl start unbound
   Verify using “systemctl status unbound”

`systemctl` is the main command used to control the **systemd** system and service manager on Linux.

4. Obtain Root Trust Anchors (for DNSSEC) using
   sudo unbound-anchor -a "/var/lib/unbound/root.key"

   DNSSEC adds a **Digital Signature** to every DNS record. Your computer checks this signature to verify the data is real.

5. Modify config files which would be located at **/etc/unbound/unbound.conf**

6. Download latest root hints and restart the service
   Command: sudo wget -O /var/lib/unbound/root.hints
   https://www.internic.net/domain/named.root
   sudo systemctl restart unbound

Root Hints is a small text file stored on the hard drive of the DNS server. It contains exactly one piece of information: **The IP addresses of the 13 Root Name Servers of the Internet.** These are the "Master Servers" of the internet.

7. To run a sample test locally, use the following command and ensure if it returns an IP or not.
   Command: dig @127.0.0.1 google.com
8. Finally to verify DNSSEC, run: dig @127.0.0.1 dnssec-failed.org

Conclusion

DNS is an invisible, yet indispensable, layer of the internet infrastructure. Its hierarchical and distributed nature ensures rapid, scalable, and resilient navigation across the global network, continually evolving through security enhancements like DNSSEC, DoH, and DoT to protect user privacy and integrity.