╰(..:: DNS handBook ::..)╯

1. What is DNS?

DNS (Domain Name System) provides a simple way for us to communicate with devices on the internet without remembering complex numbers (IP addresses). It works as the internet's phone book, translating human-readable domain names into machine-readable IP addresses.

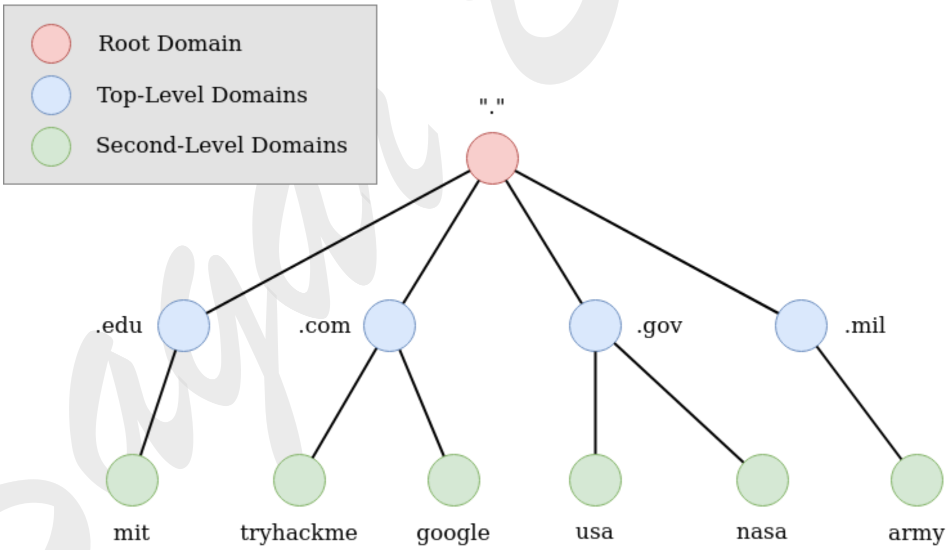
For example, instead of typing 45.64.132.70 to access a website, you can simply type portal.aiub.edu.

How it works:

- Every device on the internet has a unique IP address (e.g., 104.26.10.229)
- IP addresses are difficult for humans to remember
- DNS translates friendly domain names to IP addresses
- This translation happens behind the scenes whenever you visit a website

---

2. Domain Hierarchy

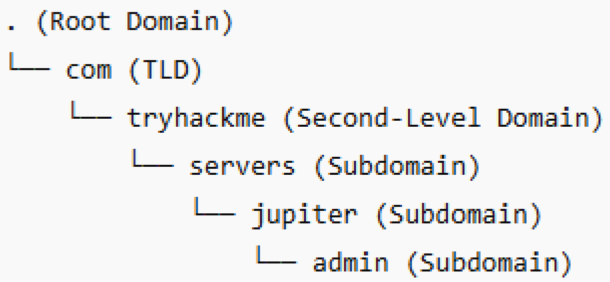


The domain name system is organized in a hierarchical structure:

Example:

Domain: admin.jupiter.servers.tryhackme.com.

The hierarchy looks like this:



```
. (Root Domain)
└── com (TLD)
    └── tryhackme (Second-Level Domain)
        └── servers (Subdomain)
            └── jupiter (Subdomain)
                └── admin (Subdomain)
```

TLD (Top-Level Domain)

The rightmost portion of a domain name, following the final dot.

Example:

Domain	: portal.aiub.edu
Second-Level Domain	: edu

Types of TLDs:

- **gTLD (Generic Top-Level Domain)**
  - Examples: .com, .org, .edu, .gov
  - Originally indicated purpose (.com for commercial, .org for organization)
  - Now includes newer options like .online, .club, .website, .biz
- **ccTLD (Country Code Top Level Domain)**
  - Examples: .ca (Canada), .co.uk (United Kingdom)
  - Indicates geographical association

Second-Level Domain

The portion directly to the left of the TLD, separated by a dot.

Example:

Domain	: portal.aiub.edu
Second-Level Domain	: aiub

Restrictions:

- Maximum **63** characters (plus the TLD)
- Can only use **a-z**, **0-9**, and **hyphens**
- **Cannot start or end** with hyphens
- Cannot contain **consecutive hyphens**

Subdomain

The portion to the left of the Second-Level Domain, separated by a dot.

Example:

Domain	: portal.aiub.edu

    - Subdomain : portal

Characteristics:

- Same character restrictions as Second-Level Domains
    - Maximum 63 characters per label
    - Can only use a-z, 0-9, and hyphens (no underscores)
    - Cannot start or end with hyphens
    - Cannot contain consecutive hyphens
- Multiple subdomains can be used (e.g., jupiter.servers.tryhackme.com, jupiter.jupiter.tryhackme.com)
- Total domain name length must be 253 characters or less
- No limit to the number of subdomains you can create

🌢 **Domain Name Rules Summary:**

- **Per Label (each section separated by dots):**
    - Maximum **63** characters
    - Allowed characters: **a-z**, **0-9**, **hyphens**
    - Cannot begin or end with hyphens
    - Cannot contain consecutive hyphens
    - Cannot contain underscores or other special characters
- **Total Domain Name:**
    - Maximum **253 characters** total length
    - **No limit** to number of subdomains

---

3. Record Types

DNS isn't just for websites - multiple types of DNS records exist for different purposes. Here are the most common ones:

♠ **A Record**

- Resolves domain names to **IPv4** addresses
- Example: portal.aiub.edu → 45.64.132.70

- Direct mapping between a name and an IPv4 address

♠ **AAAA Record**

- Resolves domain names to **IPv6** addresses
- Example: aiub.edu → 2002:2d40:8442::2d40:8442
- Used for next-generation IP addressing

♠ **CNAME Record (Canonical Name)**

- **Maps one domain/subdomain** to **another domain name** (creates an alias)
- Example: store.tryhackme.com → CNAME → shops.shopify.com

How CNAME works:

1. When someone visits store.tryhackme.com, **DNS sees** it's a CNAME
2. It redirects the **lookup** to shops.shopify.com
3. DNS then looks up shops.shopify.com, which has an **A record** pointing to an **IP address**
4. The chain is: store.tryhackme.com → shops.shopify.com → 192.0.2.5

Key Points:

- A record = Name (to)→ IP address
- CNAME record = Name (to)→ Another Name
- Eventually, DNS **must** **always end up at an A (or AAAA) record** to get an actual IP
- Think of CNAME like a **nickname** or alias

♠ **MX Record (Mail Exchange)**

- Specifies which mail servers handle email for a domain
- Includes **priority flags** to determine server order
- Example:

  Priority 10 → alt1.aspmx.l.google.com

  Priority 20 → alt2.aspmx.l.google.com

How MX works:

1. You send an email to someone@**tryhackme.com**
2. Your **mail server asks DNS** for **MX records** for **tryhackme.com**

3. DNS **replies** with the mail servers (in priority order)

4. The mail server **first tries** the lowest number priority (10)
    - If that server is available, mail is delivered there
    - If not, it tries the next priority server

Key Points:

- MX records **point to hostnames**, not directly to IPs
- Lower priority number = higher preference
- Multiple MX records provide **redundancy** for email delivery
- These hostnames then resolve to IPs via their own A records

♠ **TXT Record (Text)**

- Free text fields to store any text-based data about a domain
- Originally for **human-readable** notes, now mainly used for security and verification

Common Uses:

1. **Email Security** (SPF, DKIM, DMARC)
    - Prevent spammers from sending fake emails using your domain

💣 **Example** (SPF): v=spf1 include:_spf.google.com ~all

This tells receiving mail servers:

> *"Only these servers (e.g., Google's mail servers) are allowed to send emails on behalf of my domain."*

When spammers try to send fake emails, they often use a technique called **email spoofing** — making an email look like it came from your domain (e.g., support@yourdomain.com) even though it was sent from their own servers.

⌦ **DKIM** = Digital signature (like signing a cheque, only the bank can verify it's real).

⌦ **SPF** = Allowed sender list (like a guest list at an event).

⌦ **DMARC** = The bouncer's rulebook (decides what to do if the guest is n't on the list or has a fake ID).

In short:
Spammers fail because their mail servers aren't listed in your **SPF**, can't forge your **DKIM signature**, and get blocked by your **DMARC policy**.

2. **Domain Verification**
    - Prove domain ownership for third-party services
    - Example: google-site-verification=abcd1234xyz

3. **General Information**
   - Any arbitrary text information
   - Example: This domain belongs to TryHackMe Training

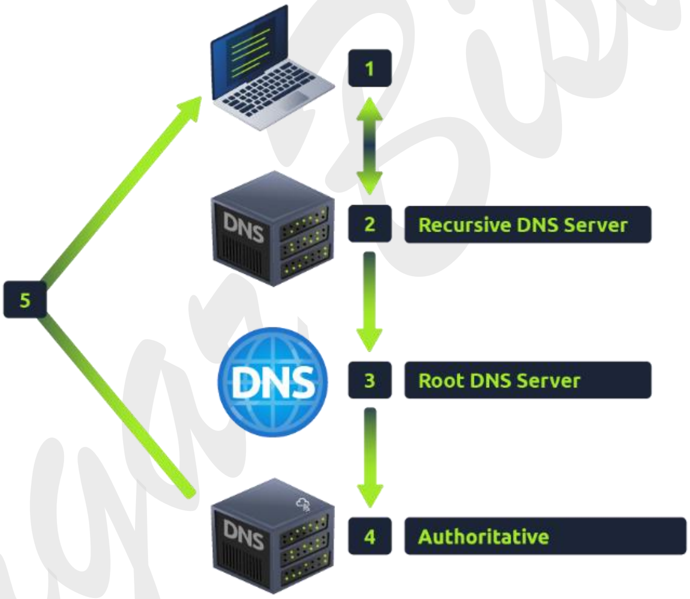
Key Points:

- TXT records don't resolve to servers or IPs—they**'re just text storage**
- They're mostly used behind the scenes for security and verification
- Mail servers **rely** on TXT records to fight spam and email spoofing

---

4. Making A DNS Request

When you type a domain name in your browser, a sequence of steps occurs to convert that name into an IP address:



♠ DNS Resolution Process

1. **Local Check**
   - Your computer first checks its local cache for recent lookups

the **local cache** is mainly stored in **RAM (temporary memory)** managed by your OS and sometimes by browsers or apps so that the next time you visit the same domain, it loads faster without asking DNS servers again.

Windows: Cached by the **DNS Client Service**. You can view it with:

```
ipconfig /displaydns
```

Browser Cache: Example (Chrome): Type **chrome://net-internals/#dns** in the address bar to view/flush it.

   - If found locally, no further lookups needed

2. **Recursive DNS Server Query**
   - If not in local cache, your computer asks a recursive DNS server
   - This server is typically provided by your ISP but can be customized
   - The recursive server **also maintains a cache** of recent lookups
   - If found in the recursive server's cache, the answer is returned immediately

3. **Root DNS Servers**
   - If the recursive server doesn't have the answer cached, it contacts a root server
   - Root servers are the backbone of DNS infrastructure
   - They direct the query to the appropriate Top Level Domain (TLD) server
   - For example, for "tryhackme.com", they direct to the **.com** TLD server

TLD Server Query

   - The TLD server maintains information about all domains under its TLD
   - It responds with the **authoritative** nameservers for the requested domain
   - For example, the **.com** TLD server provides information about nameservers for tryhackme.com

A **nameserver** is a specialized server on the internet that helps translate **domain names** (like tryhackme.com) into the **IP addresses** (like 104.20.29.66) that computers use to communicate.

4. **Authoritative Nameserver Query**
   - The recursive server contacts the authoritative nameserver
   - This nameserver is the definitive source of information for the domain
   - It's where DNS records for the domain are managed and updated
   - For **example**, portal.aiub.edu use ns2.aiub.edu as a nameserver

portal.aiub.edu itself doesn't *have its own separate nameservers*. Instead, it inherits its nameservers from the parent zone aiub.edu. Those authoritative nameservers are: ns.aiub.edu, ns2.aiub.edu

   - Multiple nameservers are typically configured for redundancy

5. **Result Return and Caching**
   - The authoritative server returns the requested DNS record

    - The recursive server caches this result according to the TTL value
    - The result is finally returned to your computer
    - Your computer also caches the result for future use

💣 **Key Components:**

- **TTL (Time To Live)**: Specifies how long (in seconds) DNS records should be cached
- **Recursive DNS Server**: Usually provided by your ISP, performs lookups on your behalf
- **Authoritative DNS Server**: Holds the official records for a specific domain
- **Caching**: Occurs at multiple levels to improve performance and reduce DNS traffic

5. Practical DNS Lookups

DNS lookup tools like nslookup or dig can be used to query different types of DNS records. Here are some examples:

💧 CNAME Record Lookup

```
C:\Windows\System32>nslookup -type=CNAME www.example.com
Server:  dns1.xpress.ltd
Address:  103.43.148.148

Non-authoritative answer:
www.example.com canonical name = www.example.com-v4.edgesuite.net
```

This shows that www.example.com is an alias (CNAME) that points to www.example.com-v4.edgesuite.net

💧 TXT Record Lookup

```
C:\Windows\System32>nslookup -type=TXT aiub.edu
Server:  dns1.xpress.ltd
Address:  103.43.148.148

Non-authoritative answer:
aiub.edu        text =

        "v=spf1 include:spf.protection.outlook.com -all"
```

Here is what it means:

- **TXT record:** A type of DNS record that stores arbitrary text, often used for verification or email security.
- **SPF record:** "v=spf1 include:spf.protection.outlook.com -all"

- v=spf1 → Specifies SPF version 1.
- include:spf.protection.outlook.com → Authorizes Outlook/Office 365 mail servers to send emails on behalf of aiub.edu.
- -all → All other servers are **not allowed** to send email for this domain.

Key takeaway:

This TXT record is mainly used to **prevent email spoofing** and ensure emails from aiub.edu are delivered properly.

💧 MX Record Lookup

```
C:\Windows\System32>nslookup -type=MX aiub.edu
Server:  dns1.xpress.ltd
Address:  103.43.148.148

Non-authoritative answer:
aiub.edu        MX preference = 0, mail exchanger = aiub-edu.mail.protection.outlook.com
```

This shows the mail server (aiub-edu.mail.protection.outlook.com) that handles email for aiub.edu with a priority value of 0.

Any email sent to @aiub.edu goes to aiub-edu.mail.protection.outlook.com.

💧 A Record Lookup

```
C:\Windows\System32>nslookup -type=A portal.aiub.edu
Server:  dns1.xpress.ltd
Address:  103.43.148.148

Non-authoritative answer:
Name:    portal.aiub.edu
Address:  45.64.132.70

C:\Windows\System32>ping portal.aiub.edu

Pinging portal.aiub.edu [45.64.132.70] with 32 bytes of data:
```

This shows the IPv4 address (45.64.132.70) for portal.aiub.edu

💣 Common nslookup Command Options:

- -type=A: IPv4 address records
- -type=AAAA: IPv6 address records
- -type=CNAME: Canonical name (alias) records

- -type=MX: Mail exchange records
- -type=TXT: Text records
- -type=NS: Nameserver records

```
C:\Windows\System32>nslookup -type=NS aiub.edu
Server:  dns1.xpress.ltd
Address:  103.43.148.148

Non-authoritative answer:
aiub.edu        nameserver = ns2.aiub.edu
aiub.edu        nameserver = ns.aiub.edu
```

- --type=SOA: Start of authority records

------------------------ X ------------------------