**NAME – ANIKET SHARMA**

**ROLL NO – 102203017**

**GROUP – 3COE-2**

**LAB ASSIGNMENT 3 (UCS534 – Cyber and Network Security)**

**Q1. Aim – To Study the DNS Hierarchy Architecture elaborating each component with the help of an example.**

**Domain Name System (DNS) hierarchy is a structured and distributed system that translates human-readable domain names (like www.example.com) into machine-readable IP addresses (like 192.0.2.1).**

**Each component of the DNS hierarchy is**

**1. Root Level**

* **Component: Root Servers**
* **Example:. (dot) at the end of www.example.com.**

**The root level is the top-most level in the DNS hierarchy and consists of root servers that store the information about the top-level domains (TLDs). There are 13 sets of root servers around the world, labelled A through M, which handle the root zone.**

**When you query for www.example.com, the DNS resolver first contacts one of the root servers to get information about the relevant TLD servers for .com.**

**2. Top-Level Domain (TLD)**

* **Component: TLD Servers**
* **Example: .com in www.example.com.**

**TLDs are the second level in the DNS hierarchy. They categorize domains by common traits, such as the type of organization (.com, .org, .net) or geographic location (.in, .uk, .us).**

**After getting a response from the root server, the DNS resolver queries the TLD server for .com to find out which authoritative name servers hold the information for example.com.**

**3. Second-Level Domain (SLD)**

* **Component: Authoritative Name Servers**
* **Example: example.com in www.example.com.**

**The SLD is the domain name chosen by the organization or individual. It's the main part of the domain name. Authoritative name servers hold the DNS records for domains like example.com, including A records, MX records, and more.**

**When the DNS resolver receives the address of the authoritative name server from the TLD server, it queries the authoritative server to find out the specific IP address for** [**www.example.com**](http://www.example.com)**.**

**4. Subdomain**

* **Component: Subdomains and Hostnames**
* **Example: www in www.example.com.**

**Subdomains are used to organize or categorize different sections of a website. In this example, www is a subdomain of example.com. Subdomains can have their own DNS records separate from the main domain.**

**Finally, the authoritative name server returns the IP address corresponding to www.example.com, allowing the user's browser to connect to the correct server.**

**5. Host (or A Record)**

* **Component: IP Address Mapping**
* **Example: 192.0.2.1 is the IP address for www.example.com.**

**The host level is where the domain name is finally mapped to an IP address. This is done through DNS records like A records (for IPv4) or AAAA records (for IPv6). The DNS resolver returns this IP address to the client, completing the lookup process.**

**Example Walkthrough:**

**For e.g. www.example.com:**

1. **Root Level: Your DNS resolver asks the root server for .com TLD servers.**
2. **TLD Level: The root server directs the resolver to the .com TLD servers.**
3. **Second-Level Domain: The .com TLD server directs the resolver to the authoritative name server for example.com.**
4. **Subdomain Level: The authoritative server has a record for www.example.com and returns the IP address.**
5. **Host Level: The resolver finally returns the IP address (e.g., 192.0.2.1) to your browser, which then connects to the website.**

**Q2. Aim – DNS Cache Poisoning (at Local DNS level)**

**To - Fetch and display entries of your Local DNS server. Perform DNS poisoning by inserting a phished/wrong entry for a domain name/IP address. Check and display the corresponding new redirection.**

**DNS cache poisoning is the act of entering false information into a DNS cache, so that DNS queries return an incorrect response and users are directed to the wrong websites. DNS cache poisoning is also known as 'DNS spoofing.'**[**IP addresses**](https://www.cloudflare.com/learning/dns/glossary/what-is-my-ip-address/)**are the 'phone numbers' of the Internet, enabling web traffic to arrive in the right places. DNS resolver caches are like a directory that lists these phone numbers, and when they store faulty information, traffic goes to the wrong places until the**[**cached**](https://www.cloudflare.com/learning/cdn/what-is-caching/)**information is corrected.**

**Because there is typically no way for DNS resolvers to verify the data in their caches, incorrect DNS information remains in the cache until the**[**time to live (TTL)**](https://www.cloudflare.com/learning/cdn/glossary/time-to-live-ttl/)**expires, or until it is removed manually.**

**How do attackers poison DNS caches?**

**Attackers can poison DNS caches by impersonating**[**DNS nameservers**](https://www.cloudflare.com/learning/dns/dns-server-types/)**, making a request to a DNS resolver, and then forging the reply when the DNS resolver queries a nameserver. This is possible because DNS servers use**[**UDP**](https://www.cloudflare.com/learning/ddos/glossary/user-datagram-protocol-udp/)**instead of**[**TCP**](https://www.cloudflare.com/learning/ddos/glossary/tcp-ip/)**, and because currently there is no verification for DNS information.**

**Instead of using TCP, which requires both communicating parties to perform a 'handshake' to initiate communication, DNS requests and responses use UDP, or the User Datagram Protocol. With UDP, there is no guarantee that a connection is open or that the recipient is ready to receive. UDP is vulnerable to forging for this reason – an attacker can send a message via UDP and pretend it is a response from a legitimate server by forging the header data.**

**If a DNS resolver receives a forged response, it accepts and caches the data uncritically because there is no way to verify if the information is accurate and comes from a legitimate source. DNS was created in the early days of the Internet, when the only parties connected to it were universities and research centres. There was no reason to expect that anyone would try to spread fake DNS information.**

**Despite these major points of vulnerability in the DNS caching process, DNS poisoning attacks are not easy. Because the DNS resolver does actually query the authoritative nameserver, attackers have only a few milliseconds to send the fake reply before the real reply from the authoritative nameserver arrives.**

**Attackers also have to either know or guess a number of factors to carry out DNS spoofing attacks:**

* **Which DNS queries are not cached by the targeted DNS resolver, so that the resolver will query the authoritative nameserver**
* **What**[**port\***](https://www.cloudflare.com/learning/network-layer/what-is-a-computer-port/)**the DNS resolver is using – they used to use the same port for every query, but now they use a different, random port each time**
* **The request ID number**
* **Which authoritative nameserver the query will go to**

**Attackers could also gain access to the DNS resolver in some other way. If a malicious party operates, hacks, or gains physical access to a DNS resolver, they can more easily alter cached data.**

**Perform DNS poisoning by inserting a phished/wrong entry for a domain name/IP address. Check and display the corresponding new redirection.**







