

|  |  |
| --- | --- |
| **All about DNS & DNS tunneling** |  |
| 04 oct 2024 |  |

**What is DNS?**

The Domain Name System, is the phonebook of the internet which translates the human readable website names like example.com to computer readable IP address like 137.465.57.5. This translation is very much important as the computers and networks always communicate with IP addresses. This translation of website names which is human readable words into the machine-friendly IP address is commonly known as the **DNS resolution**.

**What is DNS Server?**

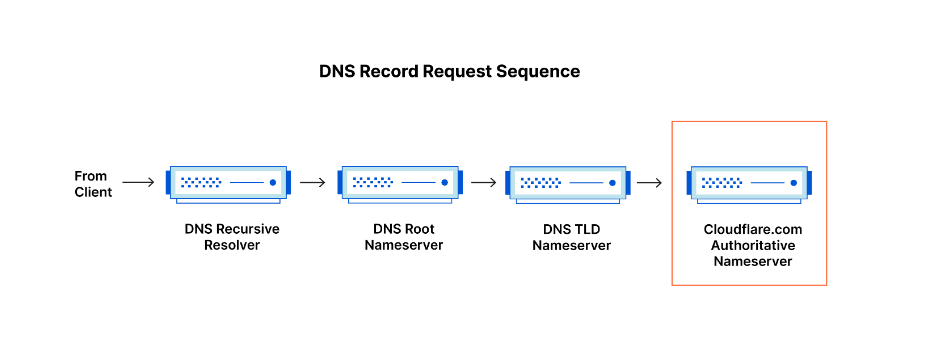
A DNS server is a computer with a database containing the public IP addresses associated with the names of the websites an IP address brings a user to.

In order to understand the process behind the DNS resolution, we should understand the 4 types of hardware which is the key essential for the DNS to work properly.

* DNS recursor: Imagine you ask a librarian to find a book. The librarian doesn’t know where it is, so they check with other people. Similarly, the DNS recursor starts the search by asking other servers to find the IP address of a website.
* Root nameserver: This is like the librarian asking the library's main index where to look. The root nameserver points to the next step, saying "try checking the section for .com websites."
* TLD nameserver: Now, the librarian knows the general section, like the ".com" area in the library. The TLD nameserver gives a more specific direction, narrowing it down to the right part of the website name.
* Authoritative nameserver: This is where the librarian finds the exact book (or in this case, the website’s IP address). The authoritative nameserver has the final piece of information and sends the correct IP back to the DNS recursor (the librarian).

If this is confusing, don’t worry the working of this sets a clear understanding.

**Working of DNS**



**Step 1:**

The client searches for a website in his/her browser, the client machine searches for its local cache if the website is searched before and has the IP address of the website. If false, then the **recursive dns query** is sent to the **dns recursor**.

**Step 2:**

The dns recursor which is also a server, gets the query from the client and searches for its cache for the website IP address. If false then it proceed to next step by communicating to the **Root nameserver** for assistance.

**Step 3:**

The Root nameserver has all the information about the TLD(top level domain) nameserver location example the Root nameservers hold information about the locations of all the TLD nameservers, such as those for .com, .org, .net, and country code TLDs like .uk, .ca, etc.

If the client is requesting a website example.com which has the .com TLD, the root nameserver points out the .com TLD nameserver and passes these information to the dns recursor.

**Step 4:**

With the TLD nameserver information, the dns recursor contacts the specific TLD nameserver asking for the website information. TLD nameservers hold information about the domain names registered under their respective TLD example .com. The TLD nameserver points out the Authoritative nameserver information to the dns recursor.

**Step 5:**

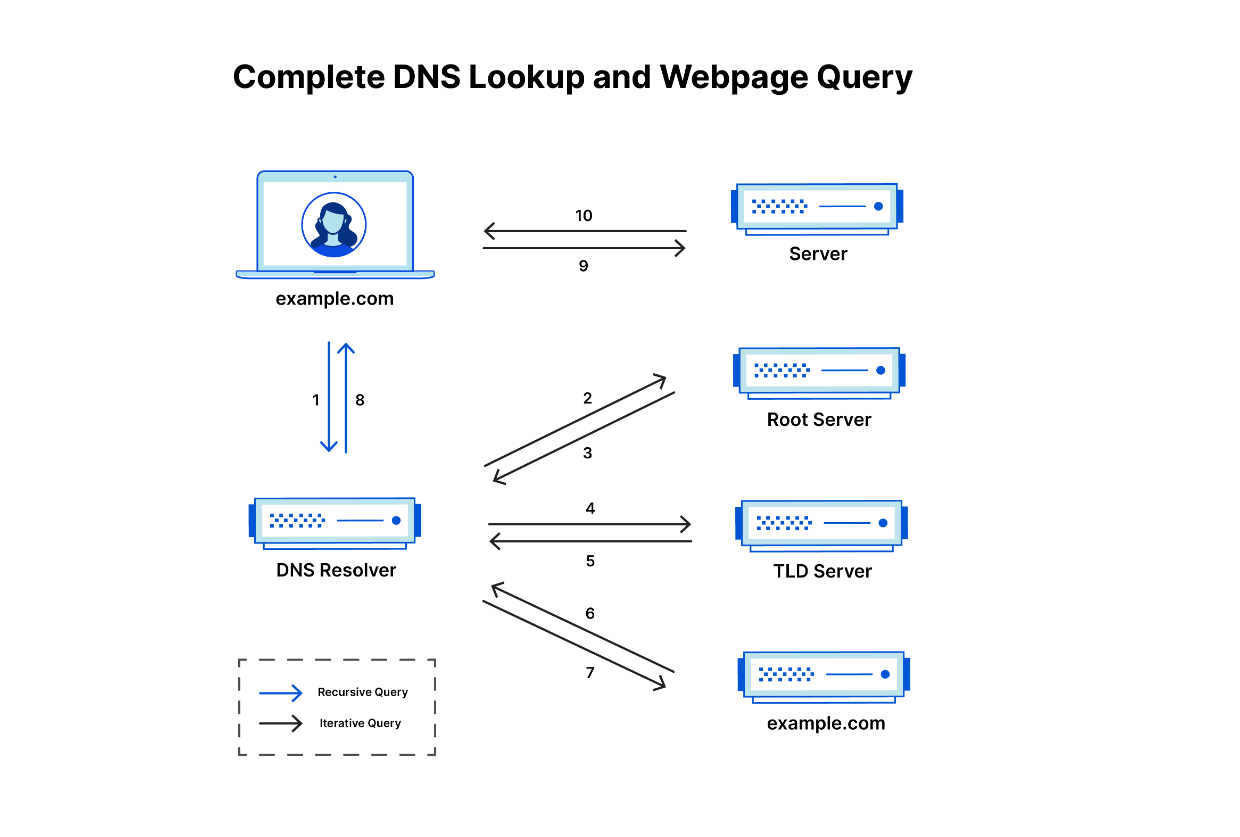
Authoritative nameservers store various DNS records for a domain, including records, Cname, MX records, NS records, etc.

With the Authoritative nameserver information, the dns recursor contacts the authoritative nameserver and gets the example.com website dns records from the authoritative namesever.

**Step 6:**

After getting the DNS records from the authoritative nameserver mainly the IP address, the dns recursor stores the IP in its cache and sends it back to the client.

After receiving the IP the website is loaded for the client and this is where two things happen. One, the IP is saved to the client’s local cache and Once the DNS server finds the correct IP address, browsers take the address and use it to send data to content delivery network (CDN) edge servers or origin servers.



After the clear understanding of DNS and how DNS work gives us the basic understanding of it. Now, there are various attack which compromises the security of the DNS servers like DNS spoofing, DNS amplification, DNS tunneling, DNS hijacking, DNS reflection etc., from the list we are focusing on the DNS tunneling.

**DNS Tunneling**

**DNS tunneling** is a method of sending non-DNS traffic (like commands, files, or other data) over the DNS protocol. However, DNS queries and responses can travel through many networks and are often allowed by firewalls, some people use DNS tunneling to send other types of data secretly.

**Why attackers use DNS tunneling?**

Attackers use DNS tunneling for three major reasons,

* Bypassing firewall
* Data exfiltration
* Remote control

Some networks block certain types of traffic (like HTTP or FTP) but allow DNS queries. Attackers or users might use DNS tunneling to send data in a way that evades these restrictions.

Malicious users can extract data from a network without detection by encoding it in DNS queries, allowing them to send sensitive information to an external server.

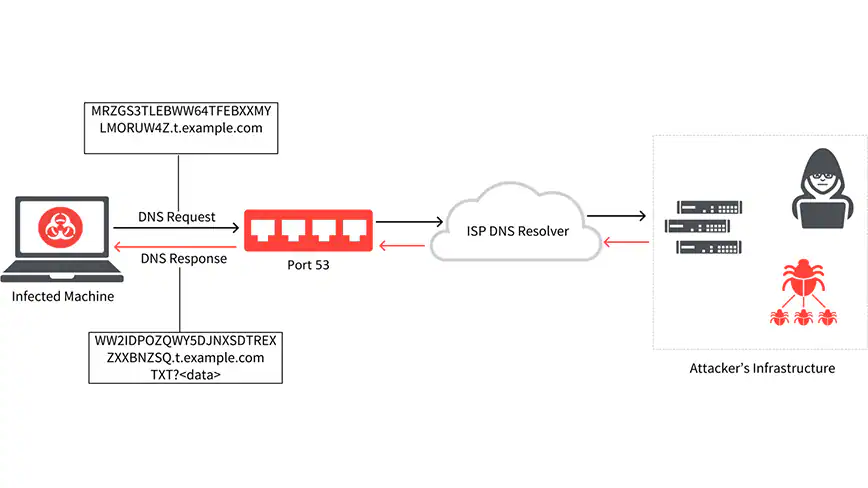
Attackers may use DNS tunneling to control compromised devices remotely, sending commands via DNS queries and receiving responses in DNS replies.

**Working of DNS tunneling**

When the attacker or user plans to make dns tunneling, the user want to first encode the message or imformation using any encrypting algorithms. The encoded message is then formated as dns query most probabily the information will be in the subdomains (coded-txt.example.com).

The dns query is sent to the dns servers. When the server gets the query it can simply ignore it or stores the information. It can also response back which may include hidden informations.

Finally the response reaches the client computer and decodes it to retrieve any additional messages or data.



**Prevention of DNS Tunneling**

* Use DNS filtering to block the unauthorized DNS requests to approve by dns servers
* Regular monitoring and logging, monitoring the traffic makes early detection. Maintaining logs of DNS queries to identify suspicious activity.
* Implement rate limiting on DNS queries to reduce the likelihood of data exfiltration attempts.