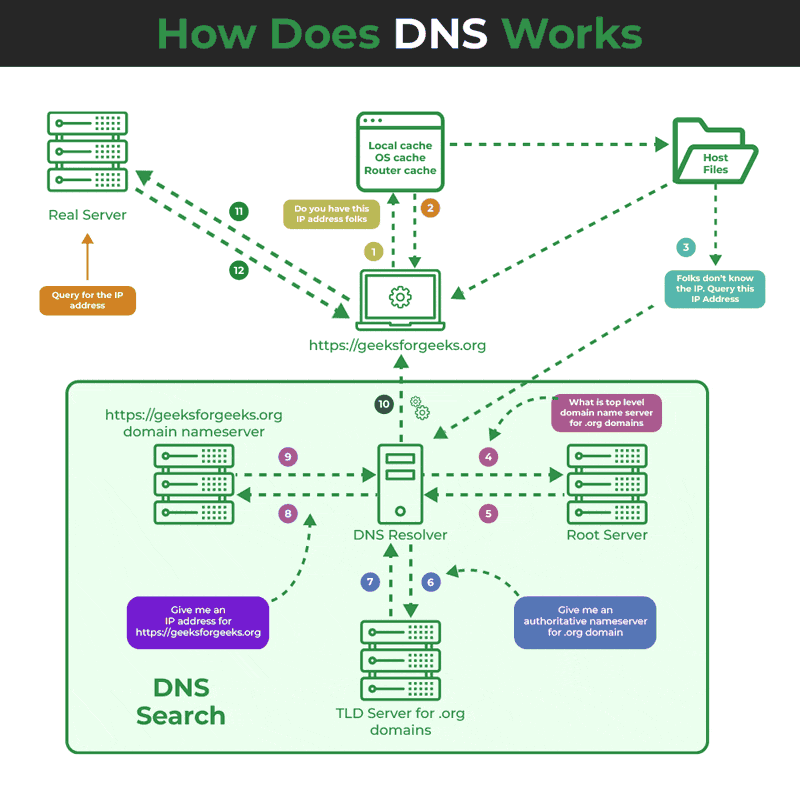
**Domain Name System (DNS)**

DNS is a hierarchical and distributed naming system that translates domain names into IP addresses. When you type a domain name like [www.geeksforgeeks.org](https://www.geeksforgeeks.org/) into your browser, DNS ensures that the request reaches the correct server by resolving the domain to its corresponding IP address.

Without DNS, we’d have to remember the numerical IP address of every website we want to visit, which is highly impractical.

**How Does DNS Work?**

The DNS process can be broken down into several steps, ensuring that users can access websites by simply typing a domain name into their browser.

How DNS Works?

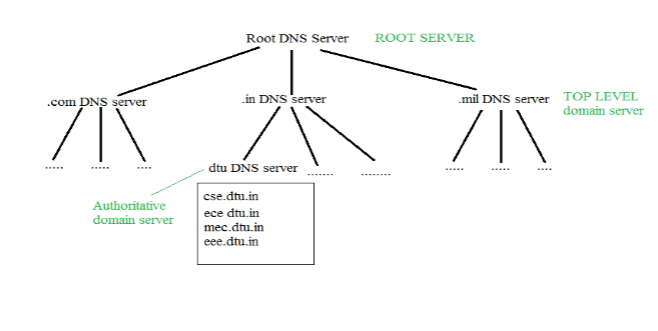
1. **User Input:**You enter a website address (for example, www.geeksforgeeks.org) into your web browser.
2. **Local Cache Check:**Your browser first checks its local cache to see if it has recently looked up the domain. If it finds the corresponding IP address, it uses that directly without querying external servers.
3. **DNS Resolver Query:** If the IP address isn’t in the local cache, your computer sends a request to a DNS resolver. The resolver is typically provided by your Internet Service Provider (ISP) or your network settings.
4. **Root DNS Server:**The resolver sends the request to a root [DNS server](https://www.geeksforgeeks.org/computer-networks/dns-server/). The root server doesn’t know the exact IP address for www.geeksforgeeks.org but knows which Top-Level Domain (TLD) server to query based on the domain’s extension (e.g., .org).
5. **TLD Server:** The TLD server for .org directs the resolver to the authoritative DNS server for geeksforgeeks.org.
6. **Authoritative DNS Server:** This server holds the actual DNS records for geeksforgeeks.org, including the IP address of the website’s server. It sends this IP address back to the resolver.
7. **Final Response:**The DNS resolver sends the IP address to your computer, allowing it to connect to the website’s server and load the page.

**Note:** This entire process happens in milliseconds, enabling a fast and efficient browsing experience.

**Structure of DNS**

DNS operates through a hierarchical structure, ensuring scalability and reliability across the global internet infrastructure. Here's how it’s organized:

* **Root DNS Servers:**These are the highest-level DNS servers and know where to find the TLD servers. They are crucial for directing DNS queries to the correct locations.
* **TLD Servers:**These servers manage domain extensions like .com, .org, .net, .edu, .gov and others. They help route queries to the authoritative DNS servers for specific domains.
* **Authoritative DNS Servers:**These are the servers that store the actual DNS records for domain names. They are responsible for providing the correct IP addresses that allow users to reach websites.



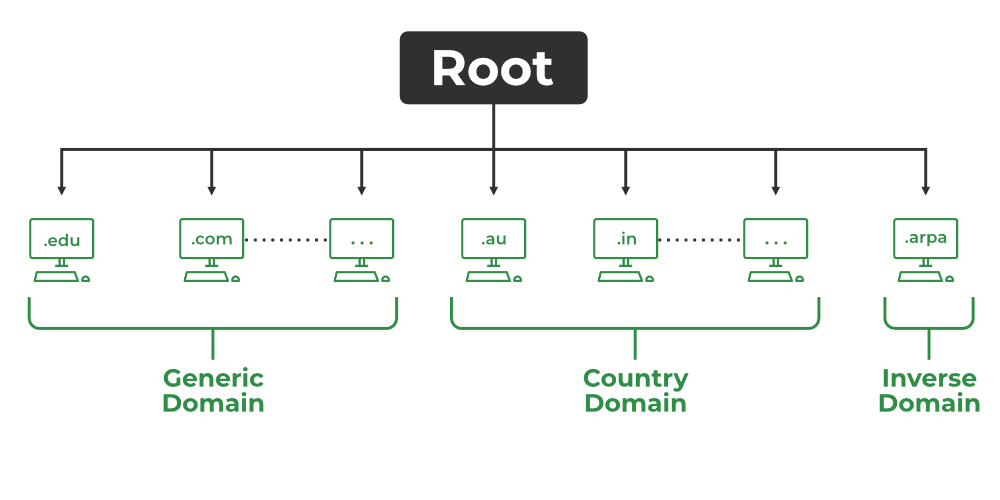
This hierarchical approach allows DNS to handle billions of queries every day, ensuring the stability and scalability of the internet.

**Types of Domains**

DNS helps manage a wide variety of domain types to organize the vast number of websites on the internet. Here are the primary categories:

* **Generic Domains:**These include top-level domains like .com, .org, .net and .edu. These are widely used and recognized across the world.
* **Country Code Domains:**These domains represent specific countries or regions, such as .in for India, .us for the United States, .uk for the United Kingdom and .jp for Japan.
* **Inverse Domains:**Used for reverse DNS lookups, these domains help map IP addresses back to domain names. Reverse DNS lookups are useful for diagnostics and security purposes, ensuring that the source of network traffic is legitimate. So DNS can provide both the mapping for example to find the IP addresses of geeksforgeeks.org then we have to type

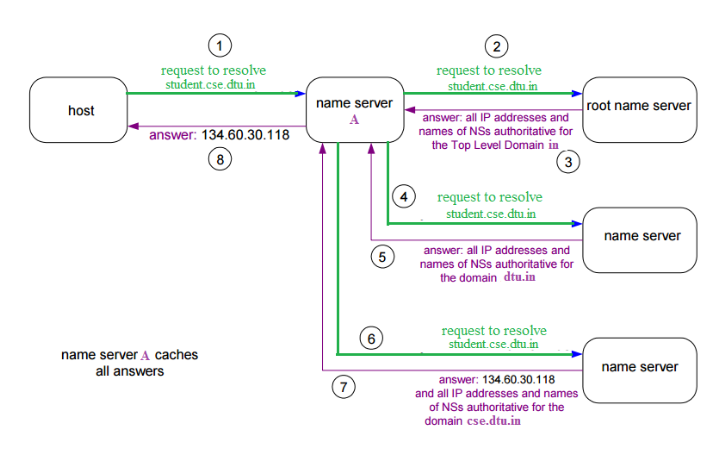
*nslookup* ***www.geeksforgeeks.org***



Understanding these categories is essential for managing domains effectively and recognizing the purpose behind different types of domains.

**Domain Name Server**

The client machine sends a request to the local name server, which, if the root does not find the address in its database, sends a request to the root name server, which in turn, will route the query to a top-level domain (TLD) or authoritative name server. The root name server can also contain some hostname to IP address mappings. The Top-level domain (TLD) server always knows who the authoritative name server is. So finally the IP address is returned to the local name server which in turn returns the IP address to the host.

Domain Name Server

**DNS Lookup**

[DNS Lookup](https://www.geeksforgeeks.org/computer-networks/dns-look-up/), also called DNS Resolution, is the process of translating a human-readable domain name (like www.example.com) into its corresponding IP address (like 192.0.2.1), which computers use to locate and communicate with each other on the internet. It allows users to access websites easily using names instead of remembering numeric IP addresses.

* DNS Lookup starts when a user types a domain name into their browser.
* The query goes through a series of servers: the DNS resolver, Root server, TLD server and authoritative server.
* Each server plays a role in finding the correct IP address for the domain.
* Once the IP address is found, the browser connects to the website’s server and loads the page.

**DNS Resolver**

[DNS Resolver](https://www.geeksforgeeks.org/computer-networks/address-resolution-in-dns-domain-name-server/)is simply called a DNS Client and has the functionality for initiating the process of DNS Lookup which is also called DNS Resolution. By using the DNS Resolver, applications can easily access different websites and services present on the Internet by using domain names that are very much friendly to the user and that also resolves the problem of remembering IP Address.

**Types of DNS Queries**

There are basically three types of DNS Queries that occur in DNS Lookup. These are stated below.

* **Recursive Query:** In this query, if the resolver is unable to find the record, in that case, DNS client wants the DNS Server will respond to the client in any way like with the requested source record or an error message.
* **Iterative Query:** Iterative Query is the query in which DNS Client wants the best answer possible from the DNS Server.
* **Non-Recursive Query:** Non-Recursive Query is the query that occurs when a DNS Resolver queries a DNS Server for some record that has access to it because of the record that exists in its cache.

**DNS Caching and TTL (Time-to-Live)**

[DNS caching](https://www.geeksforgeeks.org/computer-networks/what-is-dns-caching/) is a mechanism that stores DNS records locally to avoid querying external DNS servers repeatedly for the same information. This speeds up the browsing experience and reduces network traffic.

TTL (Time-to-Live) is the amount of time that a DNS record is cached before it expires. When the TTL expires, the cache is cleared and a fresh DNS query must be made. The TTL is defined by the authoritative DNS server, which can adjust this based on the nature of the domain.

**For example**, if the TTL for www.geeksforgeeks.org is set to 3600 seconds (1 hour), then the DNS record will be stored in the cache for one hour before it expires and requires a new lookup.

By understanding how DNS caching and TTL work, students can learn how to optimize website performance and troubleshoot issues related to stale or outdated DNS records.

**DNS Security and DNSSEC (DNS Security Extensions)**

Although DNS is essential for smooth internet navigation, it can be vulnerable to security threats. One common risk is DNS cache poisoning, where malicious actors inject harmful DNS records into caches, redirecting users to fraudulent websites.

DNS Security Extensions (DNSSEC) is a protocol designed to address these security concerns. It adds cryptographic signatures to DNS records, allowing resolvers to verify the authenticity and integrity of DNS responses. DNSSEC ensures that the information a user receives from a DNS query has not been tampered with.

By understanding DNSSEC, students can help secure websites against DNS-based attacks, ensuring safe and authentic browsing experiences.

**Reverse DNS Lookup**

Reverse DNS Lookup is the process of mapping an IP address back to its corresponding domain name. This is the opposite of the typical DNS lookup, where a domain name is resolved to an IP address. Reverse DNS is commonly used for:

* **Network Diagnostics:**System administrators use reverse DNS to determine the domain name associated with a specific IP address. This helps identify the source of network traffic.
* **Email Security:**Many email servers perform reverse DNS lookups to verify that incoming emails are coming from legitimate sources. This helps prevent spam and fraud.

A reverse DNS lookup is typically used in conjunction with standard DNS to ensure a complete and accurate mapping of network resources.

**DNS Record Types (A, CNAME, MX, TXT, etc.)**

DNS records are essential for defining how domain names are used and how services are configured. Here are some of the most commonly used DNS record types:

* **A Record:** This record maps a domain name to an IPv4 address (e.g., geeksforgeeks.org to 185.199.109.153). This is the most common DNS record used to point a domain to its website's IP address.
* **CNAME Record:**The Canonical Name (CNAME) record allows you to alias one domain name to another. For example, [www.geeksforgeeks.org](https://www.geeksforgeeks.org/) can be an alias for geeksforgeeks.org.
* **MX Record:** The Mail Exchange (MX) record defines which mail servers are responsible for receiving emails for a domain. This is crucial for setting up email services.
* **TXT Record:** The Text (TXT) record stores text-based information. It is commonly used to verify domain ownership and to implement email security protocols like SPF (Sender Policy Framework) and DKIM (DomainKeys Identified Mail).