

To answer how a DNS server works, let's see what DNS (Domain name system) is? (Source: Wikipedia): The **Domain Name System (DNS)** is a **hierarchical** decentralized naming system for computers, services, or any resource connected to the **Internet** or a private network. It associates various information with **domain names** assigned to each of the participating entities. Most prominently, it translates more readily memorized domain names to the numerical **IP addresses** needed for the purpose of locating and identifying computer services and devices with the underlying network protocols. By providing a worldwide, distributed **directory service**.

It is clear that DNS achieves this "DNS name translation (resolution)" using many distributed servers each of which can play different responsibilities but irrespective of responsibility type each one can be called a DNS server. Considering an end user perspective, a DNS server then can be visualized as - DNS name resolution served by a **single** server, though it is completely distributed throughout the world on millions of machines, managed by millions of people, and yet it behaves like a single, integrated database/directory for mapping name to address.

Considering an end user perspective, for a DNS server,

1. Input is a query (a user friendly domain name of a website)
2. Output is a Machine readable IP address (address of the website over the internet)

How this is achieved (how does this server work)? - Via DNS lookup. To explain this we can change the end user perspective of single server and start visualizing it as distributed servers (as mentioned before, each of them can be called a DNS server) working together. Following are the different server types and their workings (excluding name resolution which may happen via browser, OS and/or router cache in the beginning itself) –

1. **Recursive Resolver (Or Recursive DNS server)** : These are the first set of servers in DNS which get contacted for the name resolution purpose (can be operated by local ISP/Wireless carrier or a third party provider)
 - If it has the domain name and IP address in its database (already cached and not expired - TTL logic), it resolves the name itself.
 - If it doesn't have the domain name and IP address in its database, then it knows which other DNS servers it needs to ask answer to the original query - Root Servers, TDL, SOA (explained next). It is the recursive server responsibility to delegate responses from one server to another server till it gets the result.
 - If it has to contact another DNS server, it caches the lookup results for a limited time so it can quickly resolve subsequent requests to the same domain name.
 - If it has no luck finding the domain name after a reasonable search, it returns an error indicating that the name is invalid or doesn't exist.
2. **The root servers / root nameserver**: The first type of DNS server the recursive resolver talks to. Root server knows information about the Top level domain (server). By looking into the hierarchical name structure of a domain name (names are separated by dot (.)), the right most token helps Root server to identify suitable TDL (e.g. an appropriate server which can handle .com domain name queries), from where query can be answered. In short Recursive resolver queries a root server for TDL servers.
3. **TOP level domain name server (TDL)**: Each TOP level domain (TLD) DNS name server stores the address information (This information is basically address of SOA servers and not the address itself.) for the second level domains (e.g. google.com) with in the top level domain (e.g. .com). Each TLD has their own set of nameservers. There can be more levels (sub-domains) based on name hierarchy and for each level there could be a server which can provide the name resolution information. When query reaches the TLD server, the TLD server answers with the IP address of the **domain's** name server (SOA), which will provide the next piece of puzzle. Again, this is not the answer but the reference to someone (SOA) who can have the answer.
4. **The Domain's name server / Authoritative DNS server / start of authority (SOA)**: A DNS server that manages a specific domain is called the **Start of authority (SOA)** for that domain. The recursive resolver sends the query to the domain's name server. The DNS server knows the IP address in the form of a record (called **DNS record**) for the full domain and returns it to the recursive resolver. The recursive resolver gets

this DNS record and cache it along with its TTL (time to live) for record expiration purpose. Over time, the results from looking up hosts at the SOA will propagate to other DNS servers, which in turn propagate to other DNS servers, and so on across the Internet.

DNS lookups don't have to start at a root name server, resolver can contact a root name server as a last resort to help track down the SOA for a domain.

Source:

https://en.wikipedia.org/wiki/Domain_Name_System

<http://dyn.com/blog/dns-why-its-important-how-it-works/>

https://www.verisign.com/en_US/website-presence/online/how-dns-works/index.xhtml

<http://computer.howstuffworks.com/dns.htm>