* **Go through of how DNS works**

When you visit a domain such as dyn.com, your computer follows a series of steps to turn the human-readable web address into a machine-readable IP address. This happens every time you use a domain name, whether you are viewing websites, sending email or listening to internet radio stations such as Pandora.

**Step 1: Request information**

The process begins when you ask your computer to resolve a hostname, such as visiting https://dyn.com. The first place your computer looks for the corresponding IP address is its local DNS cache, which stores information that your computer has recently retrieved. If your computer doesn’t already know the answer, it needs to perform a DNS query to find out.

**Step 2: Ask the recursive DNS servers**

If the information is not stored locally, your computer queries (contacts) the recursive DNS servers (resolvers) from your internet service provider (ISP). These specialized computers perform the legwork of a DNS query on your behalf. Resolvers have their own caches, and given that many of the ISP’s customers are using the same resolvers, there is a reasonable chance that popular domains will already be cached. If this is the case for our example, dyn.com, the process usually ends here and the information is returned to the user.

Just about every ISP runs their own resolvers, yet those aren’t necessarily what you could be using. Some companies and perhaps even technically sophisticated home users could run their own resolvers on site. Additionally, there are several very popular open resolvers available, including Google Public DNS, OpenDNS, Dyn Recursive DNS, and Quad9.

**Step 3: Ask the root name servers**

If the recursive servers don’t have the answer, they query the root name servers. A name server is a computer that answers questions about domain names, such as IP addresses. These 13 servers act as a kind of telephone switchboard for DNS. They don’t know the answer, but they can direct DNS queries to someone that knows where to find it.

**Step 4: Ask the TLD name servers**

The root name servers will look at the first part of our request, reading from right to left — www.dyn.com — and in our case, direct our query to the top-level domain (TLD) name servers for .com. Each TLD, such as those for .com, .org, and .us, has its own set of name servers, which act like a receptionist for each TLD. These servers don’t have the information we need, but they can refer us directly to the servers that do have the information.

**Step 5: Ask the authoritative DNS servers**

The TLD name servers review the next part of our request — www.dyn.com — and direct our query to the name servers responsible for this specific domain. These authoritative name servers are responsible for knowing all the information about a specific domain, which is stored in DNS records. There are many types of records, which each contain a different kind of information.

In this example, we want to know the IP address for www.dyn.com, so we ask the authoritative name server for the address record (A record). Some authoritative name servers have intelligence that can analyze an incoming DNS query and return a response that is more performant for the user that originated the query.

**Step 6: Retrieve the record**

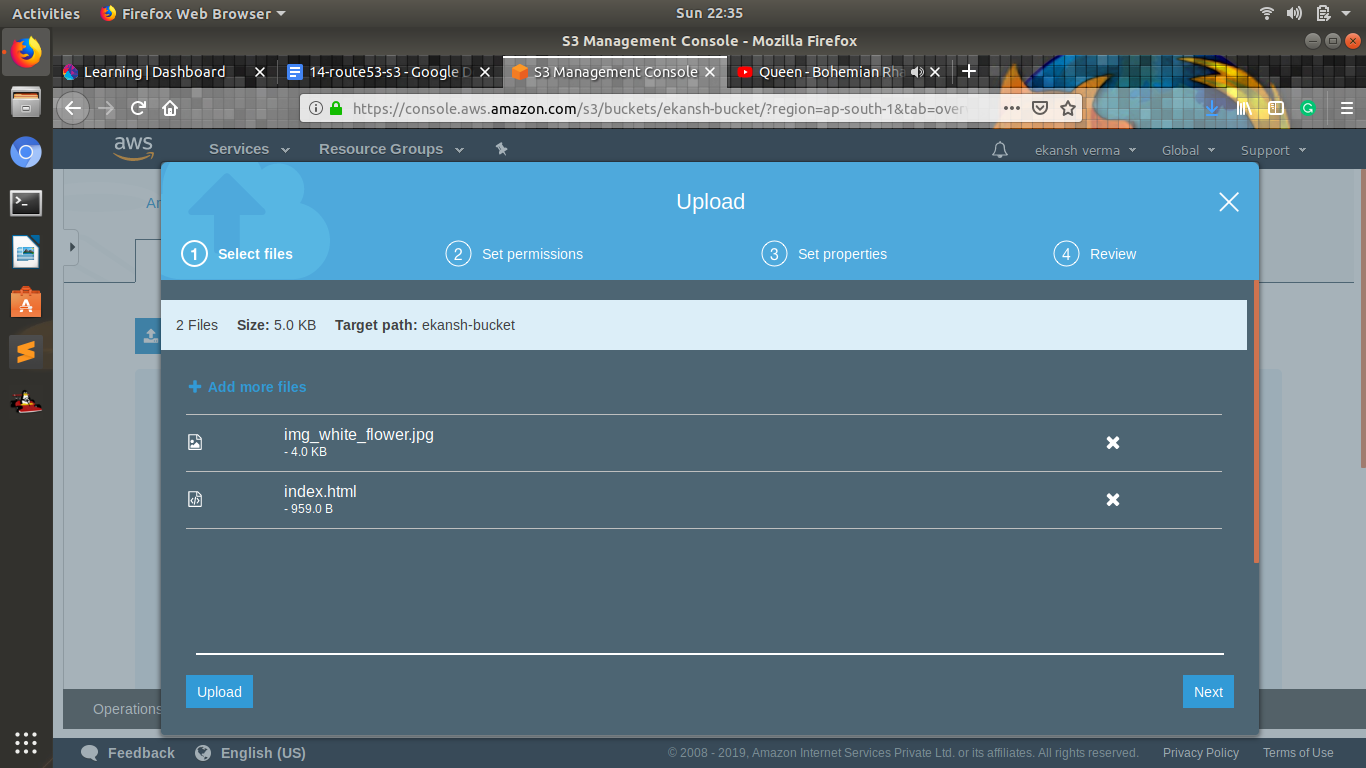
The recursive server retrieves the A record for dyn.com from the authoritative name servers and stores the record in its local cache. If anyone else requests the host record for dyn.com, the recursive server will already have the answer and will not need to go through the lookup process again. All records have a time-to-live value, which is like an expiration date. After a while, the recursive server will need to ask for a new copy of the record to make sure the information doesn’t become out-of-date.

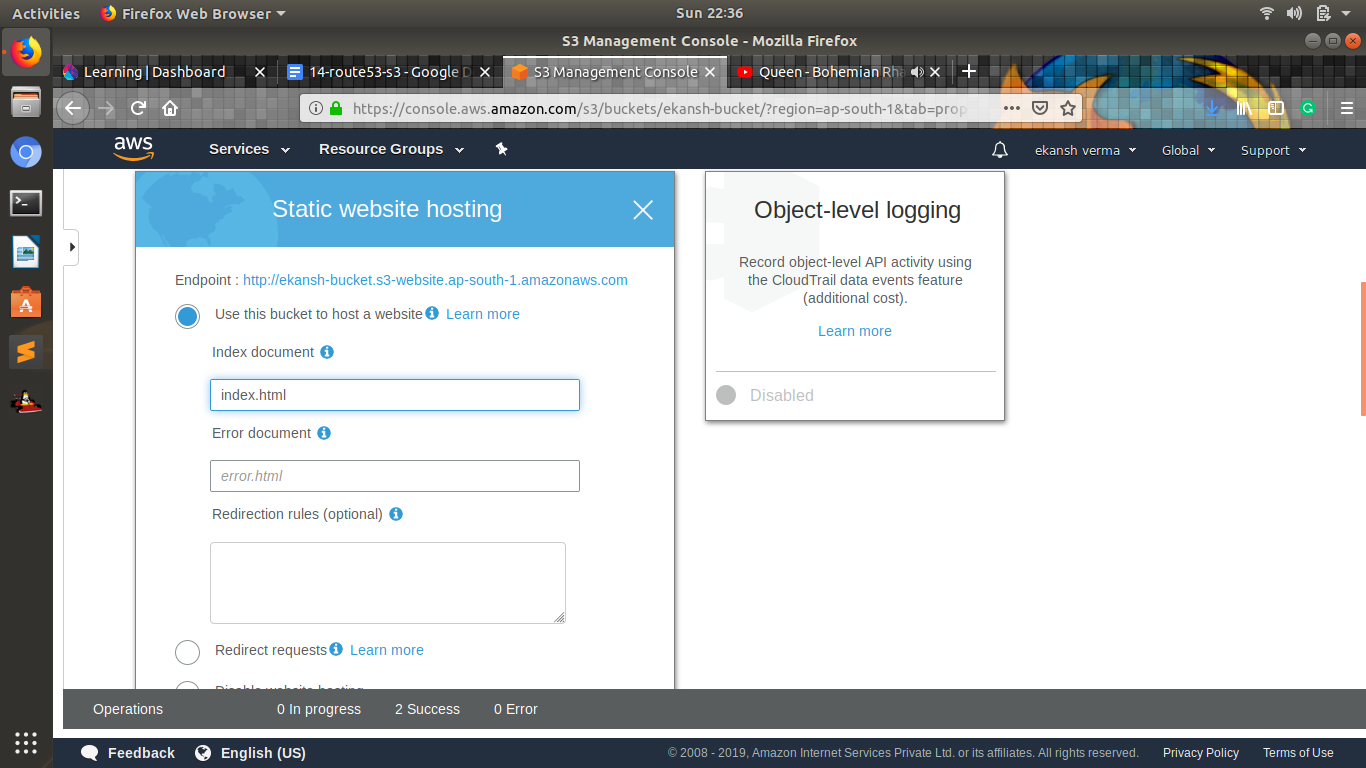
**Step 7: Receive the answer**

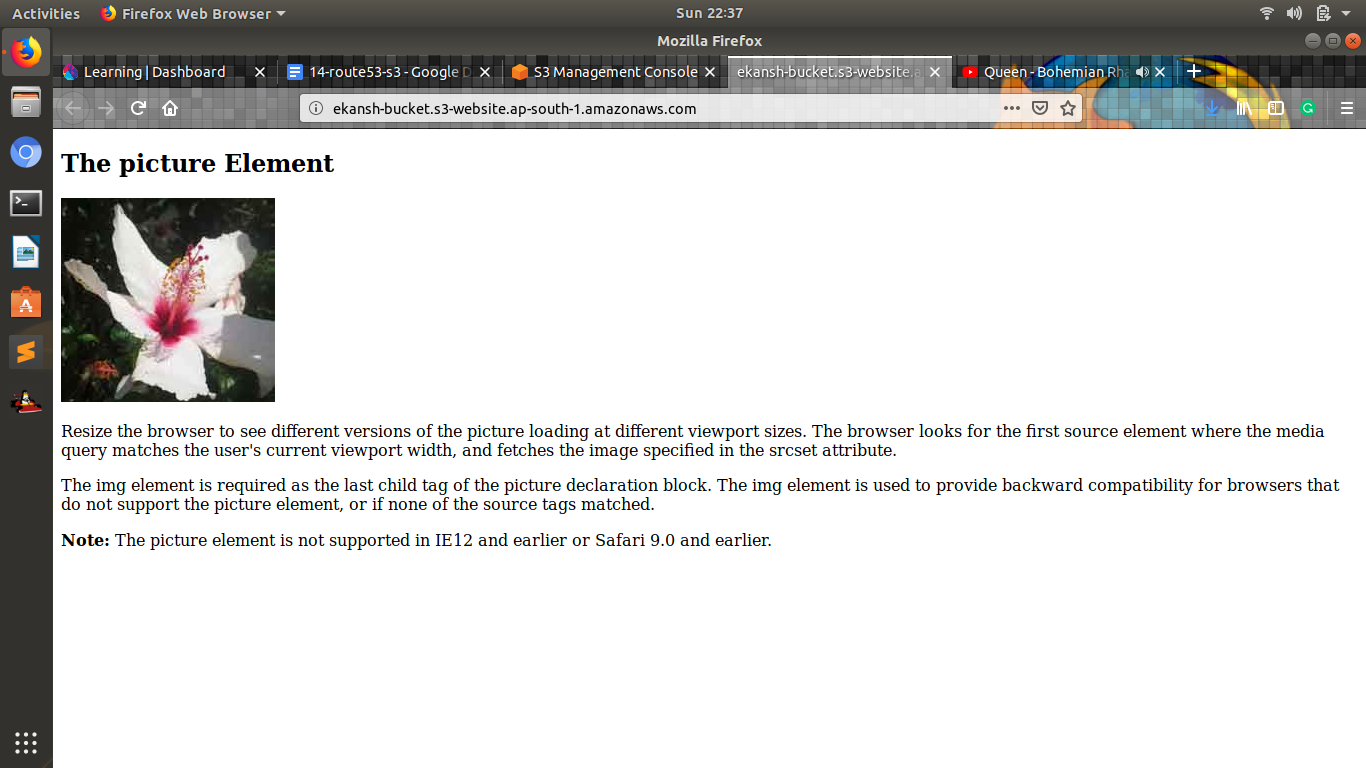
Armed with the answer, recursive server returns the A record back to your computer. Your computer stores the record in its cache, reads the IP address from the record, then passes this information to your browser. The browser then opens a connection to the webserver and receives the website.

This entire process, from start to finish, takes only milliseconds to complete.

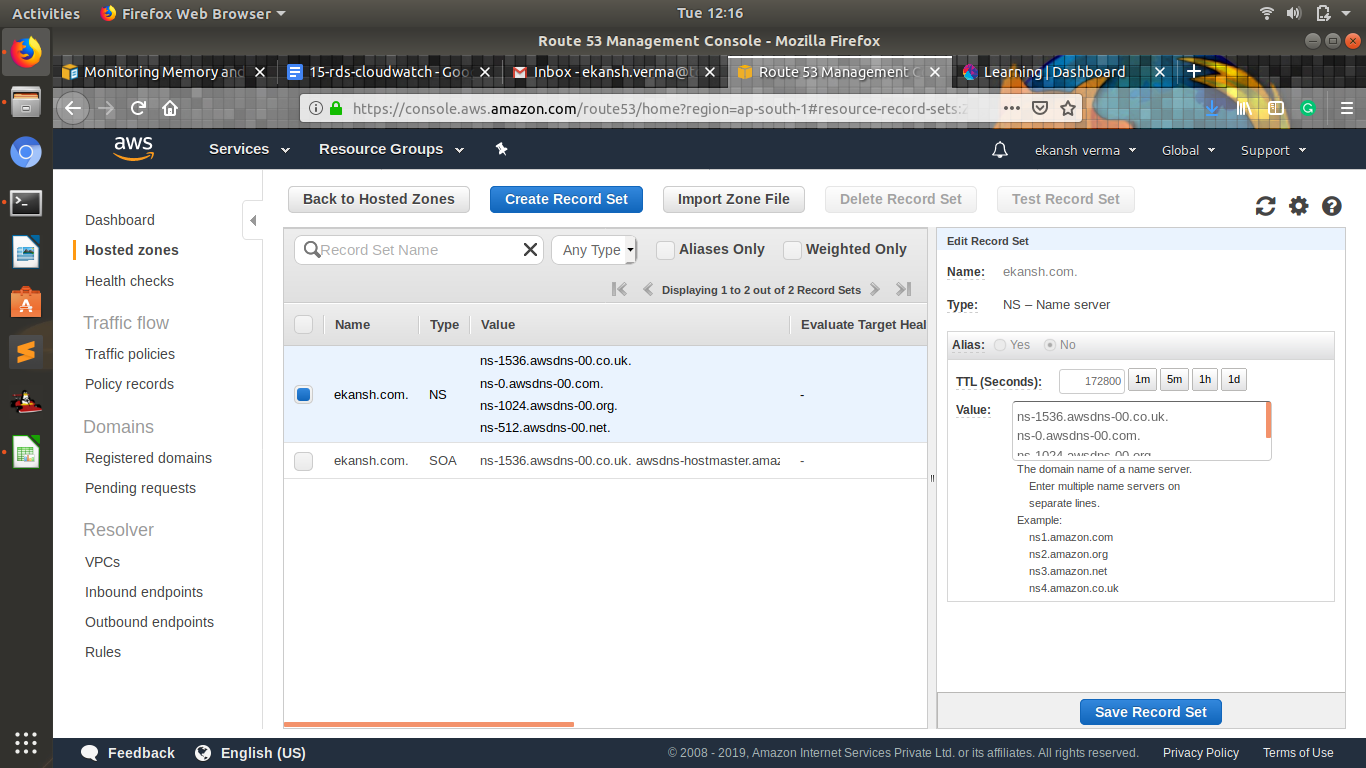
* **Create a static website using S3**







* **Create a private hosted zone in Route53**



* **Create CF distribution having S3 static website as an origin**

