**How a name search works**

Let's take look at a simplified example of what happens when a name request for a web page is made by a client service on your computer. For this example, I will use [www.opensource.com](http://www.opensource.com/) as the website I want to view in my browser. I also assume that there is a local name server on the network, as is the case with my own network.

1. First, I type in the URL or select a bookmark containing that URL. In this case, the URL is [www.opensource.com](http://www.opensource.com/).

2. The browser client, whether it is Opera, Firefox, Chrome, Lynx, Links, or any other browser, sends the request to the operating system.

3. The operating system first checks the **/etc/hosts** file to see if the URL or hostname is there. If so the IP address of that entry is returned to the browser. If not, I proceed to the next step. In this case, I assume that it is not.

4. The URL is then sent to the first name server specified in **/etc/resolv.conf**. In this case, the IP address of the first name server is my own internal name server. For this example, my name server does not have the IP address for [www.opensource.com](http://www.opensource.com/) cached and must look further afield. Let's go on to the next step.

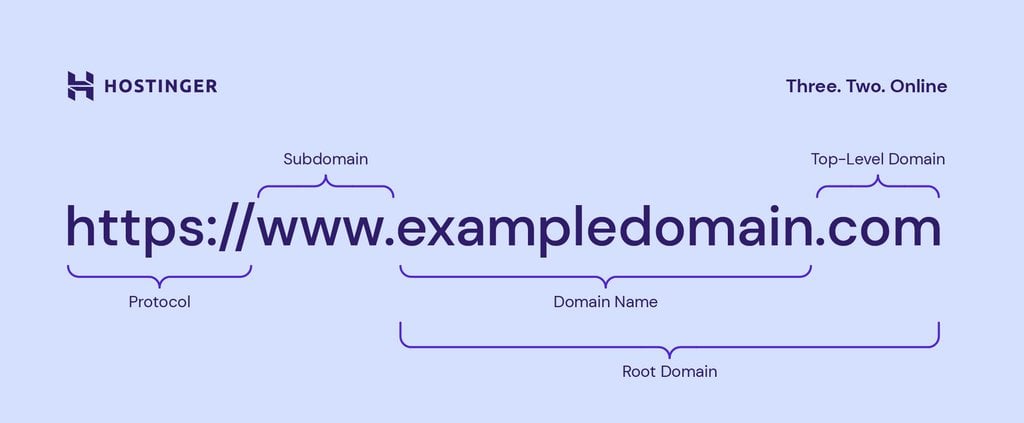
5. The local name server sends the request to a remote name server. This can be one of two destination types, one type of which is a forwarder. A forwarder is simply another name server, such as the ones at your ISP, or a public name server, such as Google at 8.8.8.8 or 8.8.4.4. The other destination type is that of the top-level [root name servers](https://en.wikipedia.org/wiki/Root_name_server). The root servers don't usually respond with the desired target IP address or [www.opensource.com](http://www.opensource.com/), they respond with the authoritative name server for that domain. The authoritative name servers are the only ones that have the authority to maintain and modify name data for a domain.

The local name server is configured to use the root name servers so the root name server for the .com top-level domain returns the IP Address of the [authoritative name server](https://en.wikipedia.org/wiki/Name_server#Authoritative_name_server) for [www.opensource.com](http://www.opensource.com/). That IP address could be for any one of the three (at the time of this writing) name servers, ns1.redhat.com, ns2.redhat.com, or ns3.redhat.com.

6. The local name server then sends the query to the authoritative name server, which returns the IP address for [www.opensource.com](http://www.opensource.com/).

7. The browser uses the IP address for [www.opensource.com](http://www.opensource.com/) to send a request for a web page, which is downloaded to my browser.

One of the important side effects of this name search is that the results are cached for a period of time by my local name server. That means that the next time I, or anyone on my network, wants to access Opensource.com, the IP Address is probably already stored locally, which prevents doing a remote lookup.



**The DNS database**

The DNS system is dependent upon its database to perform lookups on hostnames to locate the correct IP address. The DNS database is a general-purpose distributed, hierarchical, replicated database. It also defines the style of hostname used on the internet, properly called a FQDN (Fully Qualified Domain Name).

FQDNs consist of complete hostnames such as hornet.example.com and test1.example.com. FQDNs break down into three parts.

1. The TLDN ([Top-Level Domain Names](https://en.wikipedia.org/wiki/Top-level_domain)), such as .com, .net, .biz, .org, .info, .edu, and so on, provide the last segment of a FQDN. All TLDNs are managed on the root name servers. Aside from country top-level domains such as .us, .uk, and so on, there were originally only a few main top-level domains. As of February 2017, there are 1528 top-level domains.

2. The second level domain name is always immediately to the left of the top-level domain when specifying a hostname or URL, so names like Redhat.com, Opensource.com, Getfedora.org, and example.com provide the organizational address portion of the FQDN.

3. The third level of the FQDN is the hostname portion of the name, so the FQDN of a specific host in a network would be something like host1.example.com.

