

Many DNS issues result from gaps in our knowledge – typically because we do not work with DNS all that much. So I thought it would be good to give you an overview of what happens from the beginning. The premise here is you want to implement a new DNS domain name. Let's pick something to experiment with, `yada123.com`, first we need to see if the domain name is available and if so, we need to register it. For this example we are going to use GoDaddy, but any registrar should be fine. Go to www.godaddy.com. Find the search field on the form, GoDaddy's home page is a bit cluttered but the search is there. Type in `yada123.com` and hit enter or click go. Congratulations we see that `yada123.com` is available. We need to buy the domain name. Complete the purchase process and once done, you will have a new domain name.

Now what actually happened here? Once you paid for your domain. GoDaddy added a record to its database, and being that it is a registrar, it sends that information to the root servers for the domain you registered, in this example, the `{dot}com` domain. The root servers are a collection of servers distributed all over the world that keep the initial pointers to the sub domains for which it is responsible. For example, there are root servers which hold the pointers to the servers responsible for the next level zones. When you read a domain name you read from right to left, and all DNS zones begin with a period, which indicates the root, sometimes people do not type the period when referring to domains but it is implicitly there. So for example, we look at `yada123{dot}com{dot}` we see that we can start at the `{dot}` zone, the root, and ask those servers “where do I go to get information on the `{dot}com` zone?”, the root servers will return a list of server which can be asked about `{dot}com` zones. These are the servers that GoDaddy has to notify that it has a new domain that has been registered and that the nameservers for that domain are `NS57.DOMAINCONTROL.COM` and `NS58.DOMAINCONTROL.COM`, for this example.

Ok, so here is the deal, you open a browser right now and type in www.yada123.com, what happens? Your browser checks its internal DNS cache to see if there is an entry for `yada123.com`, since there is not, the browser then asks the local resolver if it has an entry for `yada123.com`, the local resolver says no, but says please hold, it (the local resolver) then asks the DNS server it has been configured with if it knows anything about `yada123.com`, that DNS server checks its cache, finding nothing, it asks the root servers for the `{dot}com` servers, it gets the answer then asks the `{dot}com` servers what they know about the `yada123{dot}com` zone, if the changes have propagated to the server you asked it should return the SOA record (Start of Authority) for `yada123{dot}com`, which is `NS57.DOMAINCONTROL.COM` and `NS58.DOMAINCONTROL.COM`. Ok now we know which server we have to ask about the `www` record for `yada123{dot}com`. Next, the DNS server opens a connection to one of those two servers (`NS57` or `NS58`) and asks for the `www` record in zone `yada123{dot}com` – what is the answer? If the record is an A record it will return an IP address. If it is a CNAME (canonical name) it will return another record to lookup, ultimately resulting in an IP address. Now wait,

what is the IP address that we are going to get in this specific case? When we registered with GoDaddy, they created a placeholder for the domain and it will direct browsers to a “parking” page hosted by GoDaddy.

So we can begin to see some of the pitfalls. There is a gap between the time you register and the time it takes GoDaddy to submit that zone to the zones root servers. Also, when you then go into GoDaddy site and change your nameservers (aka SOA) to some other server that will take some time as well. Further, if you tried to browse to your site, yada123{dot}com, your browser will cache the last answer it got, your resolver will cache the last answer it got and your DNS server will cache the last answer it got for some amount of time. So once you make a change on GoDaddy, don't expect to see it for some time. For example, GoDaddy, by default has a TTL (time-to-live) on their name servers (NS) records of 1 hour. So it can take up to an hour for that TTL to go by causing the cache to expire and causing a new lookup to occur. When you need to know what the world thinks you can run our DNSreport tool and find out what is happening right now (One of the things that DNSreport does is never caches data so that we don't have a delay problem).