# DNS Exercise 1.1

## Configure the resolver to be used by your server

Check that `/etc/resolv.conf` contains the address of a resolver that will work. You can specify more than one if you like, like this:

```
search sse.ws.afnog.org
nameserver 196.200.223.10
nameserver 196.200.223.12
```

## Generate some test DNS queries using "dig"

Check that you have "dig" installed. Debian does not install a lot of things by default, so you need to install things that are missing.

```
dig
```

```
-bash: dig: command not found
```

```
apt-get install -y dnsutils
```

```
Reading package lists··· Done Building dependency tree Reading state information··· Done Suggested packages: rblcheck The following NEW packages will be installed: dnsutils 0 upgraded, 1 newly installed, 0 to remove and 6 not upgraded. Need to get 0 B/125 kB of archives. After this operation, 379 kB of additional disk space will be used. debconf: delaying package configuration, since apt-utils is not installed Selecting previously unselected package dnsutils. (Reading database ··· 16415 files and directories currently installed.) Preparing to unpack ···/dnsutils_1%3a9.9.5.dfsg-9+deb8u11_i386.deb ··· Unpacking dnsutils (1:9.9.5.dfsg-9+deb8u11) ··· Setting up dnsutils (1:9.9.5.dfsg-9+deb8u11) ··· root@pc39:~#
```

Run each command below, look for the "ANSWER SECTION" and write down the result. Make a note of the TTL as well. Repeat the command. Is the TTL the same as in the first try? Are the responses authoritative?

```
COMMAND                          RESULT      TTL (1st)   TTL (2nd)
================================ ==========  ==========  ==========

dig www.tiscali.co.uk. a         __________  __________  __________

dig afnog.org. mx                __________  __________  __________

dig www.afrinic.net. aaaa        __________  __________  __________

dig psg.com. aaaa                __________  __________  __________

dig somnog.so a   __________  __________  __________

dig <domain of your choice> mx   __________  __________  __________

dig tiscali.co.uk. txt           __________  __________  __________

dig ripe.net. txt                __________  __________  __________

dig afnog.org. txt               __________  __________  __________

dig geek.tiscali.co.uk. a        __________  __________  __________
```

Now send some queries to another caching server. How long did it take each answer to be received?

```
COMMAND                                                  RESULT
=======================================================  ==========


 dig @8.8.8.8 psg.com. a                                  __________


dig @rip.psg.com. yahoo.com. a                           __________


dig @zoe.dns.gh. www.afrinic.net. aaaa                   __________


dig @<a-server-of-yours> <domain-of-yours> a             __________
```

# Reverse DNS lookups

Now try some reverse DNS lookups. Remember to reverse the four parts of the IP address, add '**.in-addr.arpa.**', and ask for a **PTR** resource record. For example, for 216.235.14.38:

dig 38.14.235.216.in-addr.arpa. ptr

Repeat for an IP address of your choice.

Now try the short form of dig using the '-x' flag for reverse lookups:

```
dig -x 216.235.14.38

dig -x 2001:4900:1:392::38

dig @<server-of-your-choice> -x
```

<ip-address-of-your-choice>

# Use tcpdump to show DNS traffic

Just like before, we need to make sure tcpdump is installed.

root@pc39:~# tcpdump -bash: tcpdump: command not found root@pc39:~# apt-get install -y tcpdump Reading package lists… Done Building dependency tree Reading state information… Done The following NEW packages will be installed: tcpdump 0 upgraded, 1 newly installed, 0 to remove and 6 not upgraded. Need to get 0 B/419 kB of archives. After this operation, 1,128 kB of additional disk space will be used. debconf: delaying package configuration, since apt-utils is not installed Selecting previously unselected package tcpdump. (Reading database … 16412 files and directories currently installed.) Preparing to unpack …/tcpdump_4.9.0-1~deb8u1_i386.deb … Unpacking tcpdump (4.9.0-1~deb8u1) … Setting up tcpdump (4.9.0-1~deb8u1) … root@pc39:~#

tcpdump should now be installed.

In a separate window, run the following command (you must be 'root')

```
tcpdump -n -s 0 -v udp port 53
```

This shows all packets going in and out of your machine for UDP port 53 (DNS). Now go to another window and repeat some of the 'dig' queries from earlier. Look at the output of tcpdump, check the source and destination IP address of each packet

```
-n            Prevents tcpdump doing reverse DNS lookups on the
          packets it receives, which would generate additional
          (confusing) DNS traffic

-s 0          Read the entire packet (otherwise tcpdump only reads
          the headers)

-i eth0       Which interface to listen on (use ifconfig to determine
          the name of your ethernet interface) ```

udp port 53   A filter which matches only packets to/from UDP port 53
```