DNS Exercise 1.1
==========

1. Verify the resolv.conf configuration on your workstation

cat /etc/resolv.conf

The output should look like

```
nameserver 10.10.0.254
domain ws.nsrc.org
search ws.nsrc.org
```

2. Issue the following DNS queries using 'dig'

2a. 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 nsrc.org. a			
# dig www.tiscali.co.uk. a			
# dig afnog.org. mx			
# dig www.afrinic.net. aaaa			
# dig psg.com. aaaa			
# dig <domain of your choice> 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 _____ _____ _____

2b. 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 @nsrc.org google.com. a	_____
# dig @zoe.dns.gh. www.afrinic.net. aaaa	_____
# dig @<a-server-of-yours> <domain-of-yours> a	_____

3. 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 10.10.0.250)
dig 250.0.10.10.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 196.1.95.15

dig -x 2001:42d0::200:80:1
dig -x 2001:468:d01:103::80df:9d13

dig @<server-of-your-choice> -x <ip-address-of-your-choice>

4. Use tcpdump to show DNS traffic

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

tcpdump -n -s 1500 -i eth0 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 1500
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