DNS Exercise 1

1. Configure the caching nameserver and resolver on your workstation

If you haven't gotten around to doing it during the presentation, enable your caching nameserver and configure your resolver configuration:

a) edit the file /etc/rc.conf and add:

```
named_enable="YES"
```

b) start named, the name server process

```
/etc/rc.d/named start
```

c) test that resolution works with dig:

```
dig @127.0.0.1 www.afnog.org
```

d) if all works, change your /etc/resolv.conf file so that it now contains:

```
search sae.ws.afnog.org
nameserver 127.0.0.1
```

This will have the effect that you are now using your own nameserver to resolve queries on the Internet.

2. Test that DNS works

Ping other PCs in the room (pcX.sae.ws.afnog.org), where X is 100 - 120

If in doubt, read the ping manpage (man ping)

3. Issue DNS queries using 'dig'

3a. Run each command, look for the ANSWER section and write down the result. Make a note the TTL as well.

Repeat the command. Is the TTL the same?

Are the responses Authoritative?

	RESULT 1	RESULT 2
# dig www.tiscali.co.uk. a		
# dig afnog.org. mx		
# dig news.bbc.co.uk. a		
# dig <domain of your choice> a		
# dig NonExistentDomain.ma any		
# dig tiscali.co.uk. txt		
# dig ripe.net. txt		
# dig geek.tiscali.co.uk. a		
# dig www.afrinic.net aaaa		
# dig ipv6.google.com aaaa		

3b. Now send some queries to another caching server. How long did it take each answer to be received?

```
# dig @196.200.223.1 news.bbc.co.uk. a

# dig @nsrc.org yahoo.com. a

# dig @<a server of your choice> <domain of your choice> a
```

4. Reverse DNS lookups

Now try some reverse DNS lookups.

```
       # dig -x 196.200.223.X
```

... where X is an IP address (1..254)

Repeat for an IP address of your choice.

Now try to lookup:

```
       # dig X.223.200.196.IN-ADDR.ARPA PTR
```

... where X is ***the same IP address as the one you used*** with dig -x.

What do you notice ?

5. Use tcpdump to show DNS traffic

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

```
# tcpdump -n -s 1500 udp and 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

Explanation:

-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)

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