Building a DNS cache

1. Check you have the correct packages installed

```
# rpm -qa | grep bind
redhat-config-bind-1.9.0-13
bind-utils-9.2.1-16
ypbind-1.11-4
bind-9.2.1-16        <-- This is the package we need
# rpm -qa | grep caching
caching-nameserver-7.2-7
# rpm -qi caching-nameserver    (gives a description of the package)
# rpm -ql caching-nameserver    (shows which files it contains)
```

2. Start the cache and check it is running

```
# /etc/rc.d/init.d/named start
# ps auxwww | grep named
# tail /var/log/messages
Check for successful startup, no error messages
```

3. Reconfigure your resolver to use your own cache only

Edit `/etc/resolv.conf` as follows:

```
search espe.edu.ec
nameserver 127.0.0.1
#nameserver 192.188.58.126
#nameserver 192.188.58.2
```

Remove any existing 'nameserver' lines, or comment them out by inserting '#' at the front as shown above.

4. Send some queries

Issue a query. Make a note of whether the response has the 'aa' flag set. Look at the answer section, note the TTL of the answer. Note how long the query took to process.

Then repeat the exact same query, and note the information again.

```
# dig yahoo.com.        Does it have the 'aa' flag?     ______
                        What is the TTL of the answer?  ______ seconds
                        How long is the Query Time?     ______ milliseconds

# dig yahoo.com.        Does it have the 'aa' flag?     ______
                        What is the TTL of the answer?  ______ seconds
                        How long is the Query Time?     ______ milliseconds
```

Repeat it a third time. Can you explain the differences?

Try sending some queries to your neighbour's cache. (If this fails, it may be a problem with IP firewalling)

5. Watch the cache in operation

You can take a snapshot of the cache contents like this:

```
# /usr/sbin/rndc dumpdb
# less /var/named/named_dump.db
```

(Don’t do this on a busy cache - you will generate a huge dump file!)

You can watch the cache making queries to the outside world using ’tcpdump’ in a different window

```
# tcpdump -n -s1500 -i eth0 udp port 53
```

While this is running, in the first window flush your cache (so it forgets all existing data)

```
# rndc flush
# dig yahoo.com.      -- and watch tcpdump output. What do you see?

# dig yahoo.com.      -- watch tcpdump again. This time?
```

6. Tightening up the configuration

(If you have extra time)

Following the examples on the presentation, create an acl which restricts access to your cache to your machine only. Get someone else to try to resolve names using your cache. Remember:

rndc reload
    to make your modified configuration active
tail /var/log/messages
    to check for errors in your configuration