**Lab 6 (plus HW)**

Oakland University / CSI3660

Fall 2018

**Due November 20th@ 11:59pm**

This lab is intended to provide you experience with:

1. Creating a cache-only DNS server (for quick lookup)

**LAB PROCEDURE**

One point of note, make sure that you **directly type everything…copying and pasting from Word/PDF tends to introduce odd errors!!!**

As usual, ensure that you take the appropriate screenshot as you go for your report, where the requested screenshot is marked in **bold and red**.

1. **Install and configure a DNS cache server**

While we’re not running full DNS servers, it may sometimes be helpful to run a DNS cache server. This speeds up the lookup time of resolving addresses, making browsing faster. The obvious downside is that if the server’s IP address changes, then the DNS cache will need to be updated as well.

For this lab, you might find it easier to login as root and edit all the commands with administrative privileges:

$ su -

The first thing we need to do is install the BIND utilities to enable the creation of DNS server entries.

# yum install bind bind-utils

Next, we need to edit a few configuration files. First, open up /etc/named.conf:

Look for the following lines and make the changes shown in GREEN:

listen-on port 53 { 127.0.0.1; any; };

allow-query { localhost; any; };

allow-query-cache { localhost; any; };

This enables the DNS server to listen on port 53 (we’ll open that port later).

Now, we need to make sure that /etc/named.conf is owned by root:named

# ls –l /etc/named.conf

If the file is not owned by root:named, make it so (you should NOT need to do this!):

# chownroot:named /etc/named.conf

Now we need to make sure SELinux (to be discussed when we talk about security) allows us to run this server. First, look at what access rights the DNS files have:

# ls -lZ /etc/named.conf

# ls -lZ /etc/named.rfc1912.zones

First, install the package to allow us to easily change SELinux contexts:

# yum install policycoreutils-python

Let’s make sure SELinux plays nicely:

# semanage fcontext -a -t named\_conf\_t /etc/named.conf

# semanage fcontext -a -t named\_conf\_t /etc/named.rfc1912.zones

Lastly, let’s make sure we didn’t mess up our configuration file:

# named-checkconf /etc/named.conf

Now start the service!

# systemctl start named

We need to open port 53 in the firewall to allow the DNS queries, so let’s go ahead and do that now.

# firewall-cmd --permanent --add-port 53/tcp

Now save and exit the file, then restart the firewall:

# firewall-cmd --reload

One thing we can do to secure our system a little bit is to keep the DNS server in chroot jail (making sure that any changes are limited to where DNS \*think\* the root file system is located).

So, let’s do that.

# yum install bind-chroot

And make a symbolic link to enable the jail (may exist, FYI):

# ln -s /etc/named.conf /var/named/chroot/etc/named.conf

And restart the service

# systemctl restart named

Nearly done, now it’s time to configure the ‘client’ machine (you’re using your machine as both server and client…not optimal but for the purposes of this lab it’s fine):

And now update/etc/resolv.conf with the green text**:**

***MAKE SURE YOU INSERT THIS BEFORE THE OTHER NAMESERVERS FOR IT TO TAKE EFFECT!***

**# Generated by NetworkManager**

**domain clust.secs.oakland.edu**

**search clust.secs.oakland.edu secs.oakland.edu sys.oakland.edu Oakland.edu**

nameserver <YOUR IP ADDRESS>

nameserver 141.210.8.150

nameserver 141.210.25.3

You can now exit root (just type exit). Now let’s see it working in action, using dig(make sure to run the command twice … feel free to use whichever URL you want):

*For this section, if you mess up and forget to take a screenshot, just use another domain OR clear your DNS cache using (sudo service named restart):*

$ dig www.reddit.com

$ dig www.reddit.com

If you look at the Query time, you should see a difference in how long it took to resolve the address.

**Take a screenshot of your console output, showing the difference in query time:**

Now, check to make sure the cache server is working:

$ nslookup www.reddit.com

You should see your VM’s IP address in the Server field!**Take a screenshot showing your VM’s IP address is correctly being used as the DNS server.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**SUBMISSION REQUIREMENTS / HOMEWORK:**

There are only two screenshots necessary from the lab and can be shared if you’re working in a group. **The homework must be done separately, and is a mix of practical/theoretical questions from recent classes.**

1. (20 points) Copy and paste the **two screenshots**from your lab:
2. (4 points) What is the purpose of running a DNS cache server, and what risk do you run from having a cache server?

*For Questions (3) and (4), assume we have an IP address of 141.100.25.4 and a subnet mask of 255.255.0.0*

1. (2 points) What is the network portion of the IP address above?
2. (2 points) What is the host portion of the IP address above?
3. (2 points) Assume a subnet mask of 255.255.0.0. What happens when a packet is sent to 192.168.255.255?
4. (2 points) What is the purpose of **chroot**?
5. (6 points) What are the differences between the 3 NTP strata?
6. (2 points) What is the current status of your term project?