

TCOM5350

Domain Name Service

Prof. Weiser
v.0.5

Introduction:

This lab implements the results of your homework assignment. Because different section of the lab will have different configurations to achieve, you **will not** be led through the exercise step-by-step, but instead, will be given some hints and direction and be allowed to use the built-in menu system to accomplish the goals.

Objectives:

- 1) To implement in hardware a logical design
- 2) To get some hands-on router experience, mucking around inside NetBuilder II's
- 3) Apply concepts from TCOM5123.

Prelab:

- 1) Read this handout.
- 2) Read Weiser, IP Subnet Primer

Lecture:

The instructor will present a 20 - 60 minute lecture covering selected aspects of this lab, and the goals that you should be striving to meet during the actual experiments.

Rules of Engagement:

You will need to wait for direction from the Professor or TA, indicating which portion of this exercise you are to complete. Every group will have a piece of the puzzle to fit in, and we will endeavor to keep the network functioning at all times during the transition. To do this, we will not only need to implement the design, but consider the appropriate order for the changes.

First Things First:

Let's take a look at the files that make DNS work:

*There are several basic text files that provide information to the DNS daemon. By default, Linux, starts a simple caching DNS daemon with some very basic information. For the most part a server's DNS knows about itself and how to find the root nameservers. In effect, when you point a client at the unmodified DNS, if the target host name is not the server itself, the DNS then queries directly to the root nameservers, down the tree, and finally returns the information that was determined from the authoritative server. The good thing is that the information is then **cached** in the DNS. By doing this, a site that is frequented by the users of a local DNS will not need to be looked up very frequently.*

Take a look at the following files' contents, and note what you believe each does:

`/etc/named.conf`

`/etc/hosts`

`/var/named/named.ca`

`/var/named/named.local`

Establishing a New Zone

*Domains are delegated by NIC as a result of a request. In order to qualify for management of a domain, at least two **authoritative** name servers must be registered. The locations of these nameservers are recorded in each of the **root** nameservers. The locations of the root nameservers is public information and is recorded in all non-root DNS servers. For instance, OSU maintains nameservers at 139.78.100.1 and 139.78.200.1. Each of the root nameservers points any queries ending in okstate.edu to these OSU nameservers. As authoritative servers, they must have information about all valid hosts directly, or be able to indicate another nameserver that does have the information.*

*Although most organizations managing a domain will only need one zone, larger, distributed organizations may prefer to delegate control of the portions of the domain to other name servers. Each of these "areas" of the total host name space within the domain is called a **zone**. Individual zones may be managed by different servers and a given server may be **master** for some zones and **slaves** for others.*

*Because the DNS that you establish will not be registered with either of the University's authoritative servers, we will create a fictitious domain that will only show up if a host uses your DNS for a lookup. We will pretend that you have been assigned the domain **benchX.cxm**, where **X** is the letter of room to which you're assigned. For instance the server in 104D will service the domain **benchd.cxm**.*

First, you will need to point your DNS daemon to the specific file where the individual host names reside. **Edit** `/etc/named.conf` by adding the following text to the end of the file:

```
zone "benchX.cxm" {  
    type master;  
    file "named.benchX.cxm";  
};
```

This addition will tell the daemon that if anyone querying this server is looking for addresses ending in `benchX.cxm`, this server has authoritative information and the details can be found in the file `named.benchX.cxm`. **master** means that this is the server whose information will always be considered to be the most current. There is only one master per zone, but there may be several **slaves** which read the data from the master.

Creating the Zone Data File:

Zone data may be stored in any accessible location on any file system. The location of those files can be found at the top of the `/etc/named.conf` file in the **options** section. All zone files should be placed in that directory. If you are having a hard time seeing the top of a file when you list out the contents, you can **pipe** most commands that send information to the screen to the **more** program by adding to the end of the command "`| more`" (no quotes). **more** simply takes the information it receives and displays it on screen-fuls. To get the next screen, press the spacebar.

Based upon the **options** information and the lines that you added to the `named.conf` file, you'll need to create a new zone file. One good way to do this is to start with a file that already functions and modify it for the new zone.

```
cd /var/named  
cp ./named.local ./named.benchX.cxm
```

Now you can **load up this new file into the vi editor**.

Notice that the information is in a few different sections as follows:

Start of Authority (SOA)

Names the zone, nameserver, and contact information of the responsible party, as well as specifies default information that applies to the entire site.

Name Server (NS)

Specifies the authoritative nameservers for the domain

Address Record (A)

Translations from hostnames to IP addresses

Mail Exchanger (MX)

Enables specification of the hosts that are in charge of receiving mail from the outside

Canonical Name (CNAME)

Aliases that point to address records so that a single host may be known by different hostnames.

Documentation Entries (RP and TXT)

Contact information and comments

Pointer Record (PTR)

Reverse resolution record, so that someone can look up a host name from an IP address

The very first character in the file is an @ sign. Anywhere in the file this sign appears is the same as including the entire domain name that appears in the named.conf file.

Replace **localhost.** with **benchX.cxm.** on the first line to specify the domain for which this file is authoritative. On this line and others where an address ends in a period, be sure your changes also end in a period. This specifies the address is absolute, rather than relative to the domain represented by @.

Replace **root.localhost.** with your e-mail address. Rather than using the @ sign (which is reserved for another purpose), use a "."

Arbitrarily assign the names host1, host2, host3, and host4 to the systems in your room. On the line with NS, replace **localhost.** with the name you selected for the host on which the name server resides.

Add an "A" record for each of your hosts in the following format:

```
hostname IN A IP.add.res.s
```

We will also make a web alias that points to the server:

```
www IN CNAME hostname
```

where hostname is the hostname that you assigned in the A record and want to also be referred to as www.benchX.cxm

None of this information is current, because it is read when the daemon is started. To have this take effect, you'll need to restart the daemon with the command,

```
/etc/rc.d/init.d/named restart
```

Test your configuration to make sure it all works, including all hosts and aliases served by this DNS and any other hostname on the Internet

Get the TA or Professor to initial in the box to the right, indicating that you have successfully configured a basic DNS for a fictitious domain, including alias records.

Now we'll go a little deeper. After each of the following three areas, restart the daemon, test and trouble-shoot.

Add a mail record

This will allow you to receive mail at login@bench0.cxm, rather than login@hostY.bench0.cxm

Edit your zone file to include the following line immediately after the NameServer line:

```
IN          MX 10          mailserverHost
```

*Only use the unqualified host name (ie dev156). The rest of the address will be resolved by the **A** record.*

Add a forwarder

A forwarders statement can point to specific other name servers that may be able to resolve names that cannot be resolved locally, but may be able to be resolved without going to the root server. We'll use forwarders to each of the other groups' master nameserver.

Edit the options line of the /etc/named.conf, adding the following statement

```
forwarders { xxx.xxx.xxx.xxx; yyy.yyy.yyy.yyy; };
```

including each of the IP's representing the other nameservers within the braces

Operate slaves

Slaves are also authoritative and serve either as a backup for the master, or can localize data from a frequently accessed domain. Slaves may be created solely from the slave DNS, without permission or configuration from the master. The retrieval of the current file, however, is either triggered by a timer expiration in the slave, or manually by the slave's administrator. If the master is configured, it can NOTIFY the slave anytime an update occurs, guaranteeing almost identical data in both servers.

Add the following lines to /etc/named.conf:

```
zone domain.cxm" {  
    type slave;  
    file "slave.domain.cxm";  
    masters {xxx.xxx.xxx.xxx};  
};
```

where domain.cxm is the name of the domain for which this is a slave, and xxx.xxx.xxx.xxx is the IP address of the master DNS for that domain.

Get the TA or Professor to initial in the box to the right, indicating that you now understand many of the additional complexities involved in administering a DNS.

