David Lambert

July 11, 2012

Assignment 5 - CNT4603 Langley

DNS Server Configuration with BIND

This assignment started off with a choice: the choice of which DNS server to use. Out of DJBDNS, BIND, MaraDNS, and DNSMASQ, I chose to go with BIND. After a little research of the options, BIND seemed like it was the most documented on the Internet, and this proved useful throughout the project (Google is my friend).

The goal of this assignment was to install a DNS server on the CentOS virtual machine and use it as the nameserver for both the CentOS machine and the Debian machine. This involved opening UDP access to port 53 on the CentOS machine, installing and configuring the DNS server on the CentOS machine, and configuring both machines to resolve DNS queries to the CentOS machine.

To start, I knew I needed to install BIND on the CentOS machine. This was accomplished with `yum install bind`. I also needed some ancillary utilities to help me get things working as they should - namely `dig` and `nslookup` (both a part of the `bind-utils` RPM package).

After satisfying the requisites, there wasn't anything else to do but dive right in with configuration of BIND. The pertinent configuration file, `named.conf`, was installed right into the `/etc` directory. A quick consultation of Google showed me that I didn't need the bulk of the lines of content in the `named.conf` file (most of them were commented out as options anyway). I ended up configuring a simple `named.conf` file with an "options" section and two "zone" sections: one called "cslab08.internal" and the other being the reverse records zone section called "10.168.192.in-addr.arpa". Both specified their respective configuration files that I created next.

The zone configuration files were created in the `/var/named` directory. These two files were where the grunt of the effort took place. It was a little tricky to figure out exactly how to structure the files, but once the initial learning curve of writing the first zone file was behind me, the reverse zone file went much quicker. The contents of these files basically included the associations between IP addresses and names to be resolved to, as specified in the assignment instructions.

Now was time for some testing. If the Debian machine were to use the CentOS machine to resolve domain names, it would have to be told to do so. This was done simply enough by editing the `/etc/resolv.conf` file on the Debian machine. In this file, the "nameserver" parameter was changed to point to the CentOS machine's IP address instead of Google's public DNS server (8.8.8.8).

Finally it was time to consult dig by executing `dig cslab08.internal` on the Debian machine. Unfortunately the results were not as desired: it could not connect to the CentOS server. Since I was still able to ping the CentOS machine from the Debian machine, I decided it was most likely a firewall problem.

After much frustration of trying to configure `iptables`, the only solution to get things working in the time allotted was to deactivate the firewall. Obviously not the most secure solution, but even after `nmap` showing that port 53 was opened, the Debian machine still would not resolve anything properly. After deactivation of the firewall with an `iptables` flush, both `dig` and `nslookup` confirmed that everything was working as it should.

In the end, this was a very intense project, but a very helpful one. For as long as I can remember, it has been a personal challenge of mine to learn how to set up and configure a DNS server and now I can finally say I've taken the first steps towards knowing how to do so.