CONFIGURING A DNS SLAVE
AUTHOR - Nic Maurel

Something that was plaguing my mind recently was trying to add a little redundancy to my network by adding in a secondary DNS server using BIND 9. In the event that my primary dns server would fall down the lookups on the network would not be affected as slave would then take over. Lets take a quick look at a primary dns configuration for a domain - *bobsyouruncle.com* on a *192.168.0.0/24 network*. Unfortunately this setup is only for the following:

```
Primary Server         - 192.168.0.2     -> Redhat 9 (BIND 9)
Secondary Server     - 192.168.0.3    -> Redhat 9 (BIND 9)
```

Master DNS Setup
Lets edit the */etc/named.conf*

```
# vi /etc/named/conf

// the blackslashes are soley for comments and ease of use in your /etc/named.conf
// MASTER NAME SERVER EXAMPLE
options {
  directory "/var/named";
  // version statement for security to avoid hacking known weaknesses
  version "get lost";
};
// required zone for recursive queries
zone "." {
  type hint;
  file "root.servers";
};
zone "bobsyouruncle.com"{
  type master;
  file "bobsyouruncle.domain";
// NOTE THIS OPTION IS FOR YOUR SLAVE SERVERS
  allow-transfer {192.168.0.3;};
};
// reverse lookups for class C 192.168.0.0 networks
zone "0.168.192.IN-ADDR.ARPA"{
  type master;
  file "bobsyouruncle.domain.rev";
// NOTE THIS OPTION IS FOR YOUR SLAVE SERVERS
 allow-transfer {192.168.0.3;};
};
```

See now that wasn't that hard. Now lets configure the slave.

Slave DNS Setup
The thing to note about slave DNS servers is that they are schizophrenic as they can be slaves of some zones and masters of others. This makes distributing your zones and load very easy.

```
// SLAVE NAME SERVER EXAMPLE
options {
  directory "/var/named";
  // version statement for security to avoid hacking known weaknesses
  version "go away"
};

// zone for recursive queries
// NOTE that this stays the same
zone "." {
  type hint;
  file "root.servers";
};
// see notes below
zone "bobsyouruncle.com"{
  type slave;
  file "bobsyouruncle.domain.bkp";
  masters {192.168.0.2;};
};
// reverse lookups for class C 192.168.0.0 network
//Note: I transfer the reverse lookup as well
zone "0.168.192.IN-ADDR.ARPA" IN {
  type slave;
  file "bobsyouruncle.domain.rev.bkp";
  masters {192.168.0.2;};
};
```

You will notice I use a *.bkp* extension. This is not a syntax standard, it's just a convention that I use to avoid confusion with what server I am working on, there are many other naming conventions you can use eg. *sec.bobsyouruncle.domain* or *slave.bobsyouruncle.domain*. Another thing to note is if you want your slave server to do lookups outside of your network then you must allow *port 53 tcp* and *udp* through your firewall. That's pretty much it, bobs your uncle, start named and watch those zones transfer themselves into */var/named* (or the directory specified)

Checking the Setup
If you are getting stuck just open up another console and type:

```
#tail -f /var/log/messages
```

that will provide an output to help see if the named daemon started correctly. You should see something similar to this:

```
Oct 24 18:09:00 bob named[4751]: starting BIND 9.2.1 -u named
Oct 24 18:09:00 bob named[4751]: using 1 CPU
Oct 24 18:09:00 bob named[4751]: loading configuration from '/etc/named.conf'
Oct 24 18:09:00 bob named: named startup succeeded
Oct 24 18:09:00 bob named[4751]: no IPv6 interfaces found
Oct 24 18:09:00 bob named[4751]: listening on IPv4 interface lo, 127.0.0.1#53
Oct 24 18:09:00 bob named[4751]: listening on IPv4 interface eth0, 192.168.0.3#53
Oct 24 18:09:00 bob named[4751]: command channel listening on 127.0.0.1#953
Oct 24 18:09:00 bob named[4751]: running <---- this one is the most important appart from you zones
transfering
Oct 24 18:09:04 bob named[4751]: zone bobsyouruncle.com/IN: transfered serial 2000113071
Oct 24 18:09:04 bob named[4751]: transfer of 'bobsyouruncle.com/IN' from 192.168.0.2#53: end of transfer
Oct 24 18:14:45 bob named[4751]: zone 0.168.192.IN-ADDR.ARPA/IN: transfered serial 2000101071
Oct 24 18:14:45 bob named[4751]: transfer of '0.168.192.IN-ADDR.ARPA/IN' from 192.168.0.2#53: end of
transfer
```

There are two other command's that might help with troubleshooting are :

```
named-checkconf
named-checkzone
```

The first checks your syntax in */etc/named.conf*, while the second checks your syntax in */var/named/bobsyouruncle.domain*

Final Words
Building redundancy into your network as a network is very important, especially with services like DNS, as users today rely a lot more on "friendly names" then they do on IP addresses. It is especially not nice when the server goes down and you only have one DNS server. The real test begins when you drop your primary DNS server and test whether the lookups work. I hope you've had as much fun as I have but more importantly learnt something!