Master_DNS and Slave Configuration

by karlo

Scenario:

This configuration includes configuring a MASTER_DNS (192.168.2.10) and a SLAVE_DNS (192.168.2.10).
So remember in case you want to configure only a MASTER_DNS remove any reference to SLAVE_DNS.
Linux Distributions that this should work. Centos 5.x, Red Hat 5.x and any other based on these like Fedora.

Domain Name:	training.com	
MASTER_DNS	masterdns.training.com	192.168.2.10
SLAVE_DNS:	secdns.training.com	192.168.2.10
client1	client1.training.com	192.168.2.20
cloneserver	cloneserver.training.com	192.168.2.30

Commands used for managing named.

```
# service named status        // status for named
# service named start
# service named stop
# service named restart
# service named reload        // to reload changes to named.conf or zones
# service named configtest  // to test named.conf sintaxis
```

Argument	Description
flush	Flushes the server's cache.
halt	Stops *named* immediately.
querylog	Enables / disables query logging.
reconfig	Reloads configuration file and any new zones.
refresh	Schedules maintenance for the specified zone.
reload	Reloads the configuration file and one or all zones.
restart	Restarts *named*.
stats	Writes server statistics to the log file.
status	Displays server status.
stop	Saves any pending updates and stops *named*.
trace / notrace	Enables / disables debugging level.

Important:

When there is only a master_DNS Server Name Resolution works fine but an important reminder is when working with Master/Slave Configuration is:
When you make any change to zone files under /var/named/chroot/var/named/data, for example adding a new server so you have to increase serial number by one. So when you reload or restart Master_DNS it will replicate those changes to Slave_DNS since it will check its serial number and if it is equal it won't do anything but if it's higher it will get the zones updated from Master_DNS. DO NOT FORGET

MASTER_DNS Configuration

MASTERDNS


```
# /var/named/chroot/etc/named.conf

options
{
    listen-on { 192.168.2.10; };
    directory "/var/named";
    dump-file         "data/cache_dump.db";
    statistics-file     "data/named_stats.txt";
    memstatistics-file     "data/named_mem_stats.txt";
    dnssec-enable yes;
    recursion yes;
    allow-query { any; };
    allow-notify { 192.168.2.10; 192.168.2.11; };
    version "In House DNS Test";
};

logging
{
        channel default_debug {
                file "data/named.run";
                severity dynamic;
        };
};

    zone "training.com" {
        type master;
        file "data/training.com.db";
    };

    zone "2.168.192.in-addr.arpa" {
        type master;
        file "data/2.168.192.zone.db";
        };

    zone "." {
    type hint;
    file "data/named.ca";
};
```

Understanding BIND main configuration file

- **listen-on { 192.168.2.10; }:** Listen on 192.168.2.10 IPv4 address.
- **directory "/var/named":** BIND directory to store logs and zone data
- **dump-file "data/cache_dump.db":** - The pathname of the file the server dumps the database to when instructed to do so with rndc dumpdb command.
- **statistics-file "data/named_stats.txt":** - The pathname of the file the server appends statistics to when instructed to do so using rndc stats.
- **memstatistics-file "data/named_mem_stats.txt":** - The pathname of the file the server writes memory usage statistics to on exit.

- **dnssec-enable yes:** - Enable DNSSEC support in named.
- **allow-query { any; }:** Specifies which hosts are allowed to query this nameserver. By default, all hosts are allowed to query. An access control list, or collection of IP addresses or networks, may be used here to allow only particular hosts to query the nameserver.
- **allow-notify { 192.168.2.10; 192.168.2.11; }:** Specifies which hosts are allowed to notify this server, a slave, of zone changes in addition to the zone masters.
- **version "In House DNS Test":** Set BIND version number. This is security measure for Bind not to reveal its version number.
- **logging { ... }:** - BIND provides various fine tuning options for server to log messages. The severity clause works like syslog "priorities", except that they can also be used if you are writing straight to a file rather than using syslog. Channels with dynamic severity use the server's global debug level to determine what messages to print.

16.2.3. Comment Tags

The following is a list of valid comment tags used within **named.conf**:

- **//** — When placed at the beginning of a line, that line is ignored by **named**.
- **#** — When placed at the beginning of a line, that line is ignored by **named**.
- **/*** and ***/** — When text is enclosed in these tags, the block of text is ignored by **named**.

Zone structure named.conf

```
zone <zone-name> <zone-class> {
        <zone-options>;
        [<zone-options>; ...]
};
```

; is used for comments

Example

```
zone "training.com" {
    type master;
    file "data/training.com.db";
};
```

type: Defines the type of zone.

Below is a list of valid options:

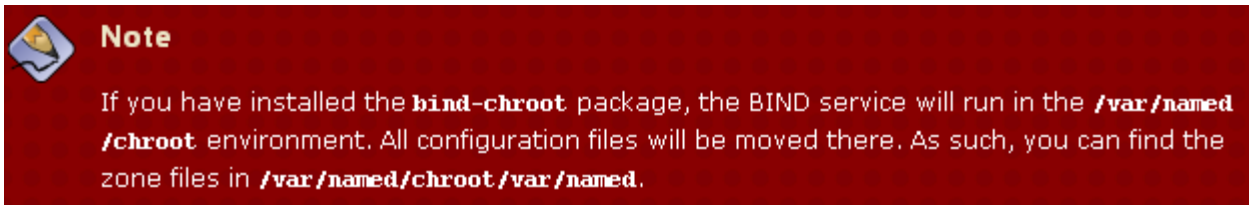
- **delegation-only** — Enforces the delegation status of infrastructure zones such as COM, NET, or ORG. Any answer that is received without an explicit or implicit delegation is treated as NXDOMAIN. This option is only applicable in TLDs or root zone files used in recursive or caching implementations.
- **forward** — Forwards all requests for information about this zone to other nameservers.
- **hint** — A special type of zone used to point to the root nameservers which resolve queries when a zone is not otherwise known. No configuration beyond the default is necessary with a hint zone.

- **master** — Designates the nameserver as authoritative for this zone. A zone should be set as the master if the zone's configuration files reside on the system.
- **slave** — Designates the nameserver as a slave server for this zone. Also specifies the IP address of the master nameserver for the zone.

file: Specifies the name of the file in the named working directory that contains the zone's configuration data.

Zone Files

These are the files define in named.conf using zone parameter: **`training.com`** and **`2.168.192.in-addr.arpa.`**



training.com.dc

```
$TTL    86400
@       IN      SOA     masterdns.training.com root.training.com. ( ; Defining SOA
                        2009111929  ; serial
                        21600       ; refresh after 6 hours
                        14400       ; retry after one hour
                        3600000     ; expire after 1 week
                        86400 )     ; minimum TTL of 1 day

                IN       NS  masterdns.training.com. ; Define NameServer Record:  IN  NS  <FQDN>
                IN       NS  secdns.training.com.    ; Define NameServer Record # 2 Slave DNS
                IN       A   192.168.2.10            ; A Record: Name -> IP
masterdns       IN       A   192.168.2.10            ; A Record: Name -> IP
secdns          IN       A   192.168.2.11            ; ""
cloneserver     IN       A   192.168.2.30            ; ""
```

2.168.192.zone.db

```
$TTL    86400
@       IN      SOA     masterdns.training.com root.training.com.   (
                        2009111930  ; serial
                        21600       ; refresh after 6 hours
                        14400       ; retry after one hour
                        3600000     ; expire after 1 week
                        86400 )     ; minimum TTL of 1 day

         IN    NS    masterdns.training.com. ; Define NameServer Record:  IN  NS  <FQDN>
         IN    NS    secdns.training.com.    ; Define NameServer Record # 2 Slave DNS
10       IN    PTR   training.com.           ; IP -> Names Main Domain Name
11       IN    PTR   secdns.training.com.    ; IP -> Names (FQDN)
30       IN    PTR   cloneserver.training.com. ; IP -> Names (FQDN)
```

Records:

A: This refers to the Address record, which specifies an IP address to assign to a name, as in this example:

```
<host>  IN      A       <IP-address>
```

If the <host> value is omitted, then an A record points to a default IP address for the top of the namespace. This system is the target for all non-FQDN requests.

Consider the following A record examples for the example.com zone file:

```
server1 IN      A       10.0.1.3
                IN      A       10.0.1.5
```

Requests for example.com are pointed to 10.0.1.3 or 10.0.1.5.

NS: This refers to the NameServer record, which announces the authoritative nameservers for a particular zone.

The following illustrates the layout of an NS record:

```
        IN      NS      <nameserver-name>
```

Here, <nameserver-name> should be an FQDN.

Next, two nameservers are listed as authoritative for the domain. It is not important whether these nameservers are slaves or if one is a master; they are both still considered authoritative.

```
        IN      NS      dns1.example.com.
        IN      NS      dns2.example.com.
```

PTR: This refers to the PoinTeR record, which is designed to point to another part of the namespace.
PTR records are primarily used for reverse name resolution, as they point IP addresses back to a particular name.

Important:

Like I said before
When makin a change to zone files increase serial number by one so Master_DNS will replicate the zone changes to the Slave_DNS

SLAVE_DNS Configuration

```
# /var/named/chroot/etc/named.conf
```

```
options
{
    listen-on { 192.168.2.11; };
    directory "/var/named";
    dump-file          "data/cache_dump.db";
    statistics-file     "data/named_stats.txt";
    memstatistics-file     "data/named_mem_stats.txt";
    dnssec-enable yes;
```

```
    recursion yes;
    allow-query { any; };
    allow-notify { 192.168.2.10; 192.168.2.11; };
    version "In House DNS Test Slave DNS";
};


logging
{
        channel default_debug {
                file "data/named.run";
                severity dynamic;
        };
};

    zone "training.com" {
        type slave;
        file "data/sec.training.com.db";
        masters { 192.168.2.10; };
    };

    zone "2.168.192.in-addr.arpa" {
        type slave;
        file "data/sec.2.168.192.zone.db";
        masters { 192.168.2.10; };
        };

    zone "." {
    type hint;
    file "data/named.ca";
    };
```

Details:

For Slave_DNS configuration what you have to change is the zone

```
type Slave;
masters { masterdns_IP; }
```

You can also change the filename for the zone

```
zone "training.com" {
        type slave;
        file "data/sec.training.com.db";
        masters { 192.168.2.10; };
    };
```

After configuring the named.conf for the Slave_DNS you have to start it it will automatically request the zones files from Master_DNS in case this does not happen have you have to go to Master_DNS and run reload command.

Client Side

vi /etc/resolv.conf

search training.com
nameserver 192.168.2.10
nameserver 192.168.2.11

Use commands to check Name Resolution

```
# ping hostname
# host hostname / IP
# dig hostname/IP
# nslookup hostname / IP
```

For checking High Availabity run

```
# watch –n nslookup hostname
```

```
Every 5.0s: nslookup cloneserver.training.com

Server:         192.168.2.10
Address:        192.168.2.10#53

Name:   cloneserver.training.com
Address: 192.168.2.30
```

Stop the Master_DNS named service or reboot the server so you will see how slave manages/handles the DNS queries.

```
Every 5.0s: nslookup cloneserver.training.com

Server:         192.168.2.11
Address:        192.168.2.11#53

Name:   cloneserver.training.com
Address: 192.168.2.30
```

Sources:

http://www.lamolabs.org/blog/282/how-to-setup-a-dns-server-on-centos-5/
http://www.centos.org/docs/5/html/Deployment_Guide-en-US/ch-bind.html
RHEL / CentOS Bind Tutorial - nixcraft.com