

CIA Practicum Assignments

Domain Name System (1)

E.P. Schatborn N.P.H. Sijm A. van Inge

7 September 2012

Abstract

The Domain Name System (DNS) is a hierarchical, distributed database, which is mostly used for matching IP-adresses to hostnames. DNS information is also used for mail routing and other internet applications. The practicum this week is about compiling, installing, configuring and testing DNS on your experimentation machines. You will also implement a delegation for your DNS. To this end your are provided with your own subdomain in the domain `practicum.os3.nl`.

1 Downloading and installing BIND

The most well known implementation of a Domain Name Server is called Berkeley Internet Name Domain (BIND). The first version of BIND was written in Berkeley by a group of students, and was paid for by DARPA. Currently the development of BIND is supported by the Internet Systems Consortium (ISC). The sources for latest version (BIND 9.9.1 as of today) can be downloaded from their website: www.isc.org.

1.1 Validating the download

The www.isc.org website provides several signature files apart from the BIND tarball. These can be used to check if you have downloaded the version they intended to distribute.

1. Why is it wise to use a signature to check your download?

Download the BIND 9.9.1 tarball and check its validity using one of the signatures.

2. Which signature is the best one to use? Why?

1.2 Installation documentation

Apart from the source code the BIND distribution contains documentation about BIND and DNS. The README file contains instructions on installation, and in the doc/ directory you can find the Administrator Reference Manual (ARM) and other relevant documents including RFCs. A good RFC to start with is RFC 1034: Domain Names - Concepts and Facilities.

1.3 Compiling BIND

Compiling and installing BIND is simple and consists of the usual sequence of commands on most systems; `./configure`, `make` and `make install`. First make sure your installation does not contain a previous version of BIND, that can really mess things up.

Compile and then install BIND in the directory `/usr/local/`. Make sure BIND will look for its configuration files in `/etc/` and have BIND write its state information such as the `named.pid` file in `/var/run`. You can find more information in the README file.

BIND will install a number of tools in `/usr/local/bin`, the daemon executables in `/usr/local/sbin` and the header files in `/usr/local/include`.

2 Configuring and testing BIND

Compiling and installing BIND is relatively simple, but configuring a BIND server is not trivial. To keep things simple we will start with a caching only nameserver. This type of name server does not control any zone data.

3. Why are caching only nameservers still useful?

2.1 ROOT servers

Our server needs a file containing references to the DNS root servers, the root hint file. The root hint file contains a list of root servers that our server uses to retrieve a more recent list of root servers. This file can be downloaded from `ftp://ftp.rs.internic.net/domain`.

2.2 localhost

The server also needs a file in the working directory called `named.local` to resolve the loopback address 127.0.0.1 to `localhost`. See figure 1 for an example configuration file for a caching only nameserver.

The file `named.local` is the zone file for the domain `0.0.127-in-addr.arpa`. It takes care of the reverse mapping of the loopback address to `localhost`. Read RFC 1033 for more information about zone files. In it the fields

```

// Working directory
options {
    directory "/etc/named";
};

// Caching only DNS server
zone "." {
    type hint;
    file "named.cache";
};

// Provide a reverse mapping for the loopback address 127.0.0.1
zone "0.0.127.in-addr.arpa" {
    type master;
    file "named.local";
    notify no;
};

```

Figure 1: An example configuration file for a caching only nameserver.

<origin>, <person> and <server> are explained. See figure 2 for an example `named.local` file.

2.3 `named.conf`

The configuration file you will have to create is called `named.conf` and should be stored in `/etc/`. It contains general options for `named` as well as references to the DNS database files and the local zone.

Create a simple configuration file for a caching only nameserver.

2.4 Testing...

You can check the syntax of your configuration file by using the `named-checkconf` program. The program only returns a result value, so you'll have to make the result visible yourself.

4. Why does the program only return a result value?

One way to see the result value is as follows:

```
if named-checkconf; then echo t; else echo f; fi
```

5. Can you think of a better way to make the result visible?

```

$TTL 86400
@ IN SOA <origin> <person> (
    1          ; serial
    360000    ; refresh every 100 hours
    3600      ; retry after 1 hour
    3600000   ; expire after 1000 hours
    3600      ; negative cache is 1 hour
)

                IN NS     <server>.
0                IN PTR   loopback.
1                IN PTR   localhost.

```

Figure 2: An example `named.local` file for the domain `0.0.127-in-addr.arpa`

3 Running and improving BIND

You can now start the `named` daemon by hand using `named -g -d2`. It will start the daemon with debug level 2 (read the `named` manual page for more information). It would be better to configure the nameserver to write debug information to a log file.

Configure `named` to write debug information to a log file.

It is also better to use the name server control tool `rndc` to start and stop the server. You'll have to adapt the configuration to enable it, this is described in chapter 3 of the ARM.

Configure `named` to be able to use `rndc`

6. What other commands/functions does `rndc` provide?

To use your own name server you will need to adapt `resolv.conf`.

7. What do you need to put in `resolv.conf` to use your own name server?

Now use the tools and scripts provided with your BIND distribution to test your nameserver.

4 Delegation

Now that you have checked that your configuration works correctly, you can set it up to serve your own subdomain of `os3.nl`. Do the following:

- Use the subdomain `city.practicum.os3.nl`, which is already delegated to your experimentation computer (`city.studlab.os3.nl`). Now create a forward mapping zone file for your domain. It must contain the following resource records:

- 2 MX records. Make sure that mail for your domain is delivered to your own computer. We will use the second MX record later on.
- 4 A records. Use your imagination...
- 2 CNAME records.

RFC 1178 provides useful tips for choosing names. It is also simply fun to read. Use the mentioned RFCs (1034 and 1033) for information about zone files and examples.

- Add a reference to your zone file to `named.conf`. Restart or reload the nameserver and test your configuration using the BIND tools.
 8. If Niels had not yet implemented the delegation, what information would you need to give him so that he can implement it?
 9. Describe the official delegation proces for a domain in the netherlands (requesting / assignment / denial / testing / etcetera). What organisations are involved (other than you and your registrar)?
 10. What important requisite is not yet met for your subdomain?