Assignment #3: DNS(1)

Emil Sharifulllin, Innopolis University

September 13, 2016

1 Introduction

The nameserver was deployed at Ubuntu:16.04 docker container:

```
$ cat /etc/*release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=16.04
DISTRIB_CODENAME=xenial
DISTRIB_DESCRIPTION="Ubuntu 16.04.1 LTS"
NAME="Ubuntu"
VERSION="16.04.1 LTS (Xenial Xerus)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 16.04.1 LTS"
VERSION_ID="16.04"
HOME_URL="http://www.ubuntu.com/"
SUPPORT_URL="http://help.ubuntu.com/"
BUG_REPORT_URL="http://bugs.launchpad.net/ubuntu/"
UBUNTU_CODENAME=xenial
```

2 Downloading and Installing a Caching Nameserver

Nameserver can be downloaded from site http://www.isc.org by following command:

```
$ wget -O bind.tar.gz https://www.isc.org/downloads/file/bind-9-10-4-↩
    p2/?version=tar-gz
```

To unarchieve tarbal we can use following command:

```
$ tar -xzvf bind.tar.gz
```

2.1 Validating the Download

Download can ve validated by GPG with ASC file. To download this file and check package we can use following bash command.

```
$ wget -O bind.tar.gz.asc ftp://ftp.isc.org/isc/bind9/9.10.4-P2/bind↩
    -9.10.4-P2.tar.gz.asc
$ gpg --verify bind.tar.gz.asc bind.tar.gz
gpg: Signature made Mon Jul 18 22:59:45 2016 UTC using RSA key ID 911↩
    A4C02
gpg: Good signature from "Internet Systems Consortium, Inc. (Signing ↩
    key, 2015-2016) <codesign@isc.org>"
```

Why is it wise to use a signature to check your download?
Signing is very important because after checking signature you can be sure that certain tarbal is right and no one can spoof this tarbal.

Which kind of signature is the best one to use? Why?
PGP signing is the safest way to check package validity because probability of collision of PGP key is lower than probability of collision of SHA key.

2.2 Installation Documentation

Documentation can be found in doc/ directory. README file contains installation guides and is located at root installation directory.

2.3 Compiling

To compile bind you need to rum following commands:

```
$ ./configure
$ make
$ make install
```

3 Configuring and Testing

After installing DNS server we will start to configure it.

Why are caching-only name servers still useful?
It can store resourse Records during theirs TTL to decrease DNS resolving time and decrease network flood. It can reduce the time needed to internet communication.

3.1 Main Configuration

To configure BIND as a caching server it is necessary to create file /etc/named.conf . To enable remote control to nameserver we need to generate keys for RNDC.

```
$ rndc-confgen
# Start of rndc.conf
key "rndc-key" {
    algorithm hmac-md5;
    secret "VERY SECRET";
};

options {
    default-key "rndc-key";
    default-server 127.0.0.1;
    default-port 953;
};
```

3.2 Root Servers

BIND needs list of root servers to work. this list can be downloaded from ftp://ftp.rs. internic.net/domain. In named.conf we need to define path to root severs cache file.

```
zone "." {
    type hint;
    file "named.root";
};
```

3.3 Resolving localhost

To enable reverse mapping for loopback address 127.0.0.1 we need to add zone file.

```
$TTL 86400
@ IN SOA localhost. admin.localhost. (
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  ns.st10.os3.su.
0   IN  PTR loopback.
1   IN  PTR localhost.
```

Now that you know all the elements of the main configuration, create a simple named.conf or unbound.conf for a caching-only name server. Show the configuration file in your re-

port.

```
//Define a access list to limit recursion later
acl localnet {
    127.0.0.1/32;
};

controls {
    inet 127.0.0.1 port 953
        allow { 127.0.0.1; } keys { "rndc-key"; };
};

key "rndc-key" {
    algorithm hmac-md5;
    secret "xxsGwSWnOTIOvyIbdFjtAQ==";
};


// Working directory and limit recursion
options {
    directory "/etc/bind";
    allow-recursion {
        localnet;
    };
};

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

zone "st10.os3.su." {
    type master;
    file "st10.os3.su.zone";
};

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

3.4 Testing

To test configuration files we can use two checkconf and checkzone programs

```
$ named-checkconf
$ echo $?
0
$ named-checkzone localhost /etc/bind/local.zone
zone localhost/IN: loaded serial 1
OK
```

3.4.1 Testing of cache server

```
$ dig google.com @127.0.0.1
...
;; Query time: 318 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
...

$ dig google.com @127.0.0.1
...
;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
...
```

Why do the programs return a result value?

Returning result value is very useful and can be used in bash scripts.

4 Running and Improving the Name Server

Show the changes you made to your configuration to allow remote control

To allow rndc tool I added to named.conf following lines:

Listing 1: named.conf

```
controls {
    inet 127.0.0.1 port 953
        allow { 127.0.0.1; } keys { "rndc-key"; };
};

key "rndc-key" {
    algorithm hmac-md5;
    secret "xxsGwSWnOTIOvyIbdFjtAQ==";
};
```

Listing 2: rndc.conf

```
key "rndc-key" {
    algorithm hmac-md5;
    secret "xxsGwSWnOTIOvyIbdFjtAQ==";
};

options {
    default-key "rndc-key";
    default-server 127.0.0.1;
    default-port 953;
};
```

What other commands/functions does rndc/unbound-control provide? rndc allows you to control nameserver without stopping and restarting nameserver daemon.

What do you need to put in resolv.conf (and/or other files) to use your own name server?
Adding namesevers to resolv.conf is the bad way because after restarting network manager will restore previous values of resolv.conf. To configure Ubuntu to use my own nameserver I need to add following strings to /etc/resolvconf/resolv.conf.d/base and then run sudo resolvconf -u

```
nameserver 127.0.0.1
```

5 Configuring an Authoritative Nameserver

Show the forward mapping zone file in your log.
My zone file contain following information:

```
$TTL 86400
@ IN SOA ns10.os3.su. admin.st10.os3.su. (
2016082900  ; serial
360000  ; refresh every 100 hours
3600    ; retry after 1 hour
3600000 ; expire after 1000 hours
3600    ; negative cache is 1 hour
)


@       IN  NS      ns10.os3.su.
@       IN  A       188.130.155.43
@       IN  MX  10  mail
```

```
st11    IN  NS      ns11.os3.su.
ns      IN  A       188.130.155.43
www     IN  A       188.130.155.43
mail    IN  A       188.130.155.43
web     IN  CNAME   www
mob     IN  CNAME   www
ns1     IN  CNAME   ns
ns2     IN  CNAME   ns
```