



Domain Name System

KTH NS-Lab

Short XEN version

Group Nr	
Name1	
Name2	
Date	
Grade	
Instructor's Signature	

Table of Contents

1 Goals.....	3
2 Connecting to the lab.....	3
3 Using dig to explore the DNS system.....	4
4 Setting up BIND.....	5
5 Accepting the zone delegation.....	5
5.1 Getting the files right.....	5
5.2 Edit the zone file.....	6
6 Adding records to your zone.....	6
7 Creating a sub-zone.....	7
8 Delegating a sub-zone.....	7
9 Masters and slaves, replication.....	8
10 Verification of zones.....	9
References.....	9

1 Goals

This lab is an introduction to the Domain Name System. The goal is to give you knowledge so that you can configure, setup and troubleshoot DNS systems. More specifically, you will get a basic understanding of the dns system BIND and the dns lookup tool DIG.

The lab will also introduce the various types of data stored in DNS, the caching system and master-slave relationships for redundant servers.

The lab assumes basic knowledge of UNIX style systems, including how logging is done and how services are used. It is also absolutely necessary to be able to edit text files using a standard text editor.

2 Connecting to the lab

The following information will be given to you at the start of the lab.

Your two-digit hex identifier (x): 72 (working with 71) (provided by lab assistants)

Your XEN host instance: dns-72.xen.netlab.csc.kth.se (fill in x)

Your DNS zone: 72.experiment.xen.netlab.csc.kth.se (fill in x)

Your partner's DNS zone: 71.experiment.xen.netlab.csc.kth.se (fill in y of another group)

Login: student

Password: m0rris (provided by lab assistants)

Now you can log in on your xen instance from your client computer:

```
ssh student@dns-<x>.xen.netlab.csc.kth.se
```

which is a virtual computer running OpenSUSE Linux.

As soon as you have logged in, use the `sudo bash` command to enter root privileges (you need to enter the password again).

The IPv4 address of your computer is the one you will use for all your servers in the lab.

IPv4 address: 193.11.23.35 (find out using `ifconfig`)

You may have problem logging in on the xen clients using some client software. For example, xterm on Sun clients seem to function badly. Use "terminal" instead.

In the xen computers you will be editing files. You will have to master a text editor. There are several text editors available, including emacs, vi, nano. The most intuitive is nano. If you do not know how to handle a text editor this lab is not for you, the lab assistants will not help you to master a text editor. Emacs has a nice DNS mode by default which updates serial numbers automatically. But if you have a numeric hex identifier (eg 21) emacs thinks your domain name is a serial number and changes that. Change to text-mode (M-x text-mode).

3 Using dig to explore the DNS system

Dig is a tool for sending queries to DNS servers and view the results. Typical output from Dig looks like this:

```
# dig www.kth.se
; <<>> DiG 9.3.4 <<>> www.kth.se
;; global options: printcmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 53537
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 0

;; QUESTION SECTION:
;www.kth.se.                IN      A

;; ANSWER SECTION:
www.kth.se.                 60      IN      CNAME   lvs-vip-
6.sys.kth.se.               7200    IN      A       130.237.32.143
lvs-vip-6.sys.kth.se.

;; AUTHORITY SECTION:
sys.kth.se.                 7200    IN      NS      a.ns.kth.se.
sys.kth.se.                 7200    IN      NS      b.ns.kth.se.

;; Query time: 3 msec
;; SERVER: 192.71.24.1#53(192.71.24.1)
;; WHEN: Wed Nov 19 18:08:18 2008
;; MSG SIZE rcvd: 107
```

Make sure you understand the status and flags fields. They are as important as the records returned. Reference the questions above.

Dig is a powerful tool to explore the DNS system. Use Dig to answer the following questions:

What is the IPv4 address of netlab.csc.kth.se?

192.71.24.1

What is the IPv6 address of ipv6.google.com?

2a00:1450:500f:802::1008

What data is stored in the TXT record of experiment.xen.netlab.csc.kth.se?

Fuzzy Kittens Rules!

Use the trace flag to do an iterative lookup for xen.netlab.csc.kth.se. List the servers queried and what record is used from each server:

```
- Root server («.» ) NS
- c.root-servers.net NS
- c.ns.se NS
- a.ns.kth.se NS
- ns2.nada.kth.se SOA
```

Dig can also do reverse DNS, mapping IP addresses to DNS name. This is done by doing a query for the PTR records of a specially formed DNS name. Use dig with the trace flag to get the DNS name of 130.237.32.143. Use the returned data to illustrate how reverse lookups are done in DNS.

dig -x 130.234.32.143 +trace -> polyfront-1-old.sys.kth.se.

You can request to ask a specific nameserver in DIG. How is this done?

dig name@servername

How do you query for a zone transfer? Try to get all records from experiment.xen.netlab.csc.kth.se. **dig experiment.xen.netlab.csc.kth.se. AXFR**

125 records !

4 Setting up BIND

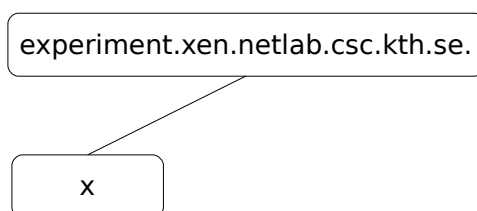
Bind is a daemon program (actually called “named”), which means it runs in the background most of the time. It will not give you any direct output but syslogs its output. Interaction with bind is done through the system log, zone files, the configuration files and the rndc client.

To check that bind is running, log in to the computer and type
`# service bind9 status`

You configure bind by editing /etc/bind/named.conf.local
 Operation of named can be controlled by the program rndc (eg rndc reload after zone changes). Always check the syslog for errors after changes (rndc might report success even if parts of the configuration contain errors):
`# tail /var/log/syslog`

IIS has a tool that lets you verify if a DNS zone is working correctly. This tool can be accessed on <http://dnscheck.iis.se/>. The goal of the complete lab is to pass all its tests. But you can use it intermediately as a debugging tool.

5 Accepting the zone delegation



A central issue with DNS is that of delegation. In order to be tied to the global DNS tree, someone else needs to delegate a zone to you. You have been given the DNS name <x>.experiment.xen.netlab.csc.kth.se. This means that experiment.xen.netlab.csc.kth.se has delegated the <x> zone to you. This has already been setup prior to the lab by assigning ns.<x>.experiment.xen.netlab.csc.kth.se as your nameserver. In the delegation there is also a secondary nameserver allocated, but you can ignore this until Section 11.

Verify the delegation from experiment.xen.netlab.kth.se using dig! How is this done?

By using +trace : dig +trace experiment.xen.netlab.kth.se

Your task is to accept this delegation by configuring a DNS server to answer for zone <x>. You do this by creating a zone file containing a SOA record matching the zone and adding your zone to the bind configuration file.

5.1 Getting the files right

Add a new zone entry to the /etc/bind/named.conf.local file. Don't make any other changes. Create a new zone file in /var/lib/bind/. When you make changes to /etc/bind/named.conf.local and your zone file, reload named and *always* check for any error messages:
`# rndc reload`

```
# tail /var/log/syslog
```

Verify that the correct zone file is found and loaded.

5.2 Edit the zone file

To get a working zone file you need to specify (at least) the following records:

- Specify \$TTL using a low value
- The \$ORIGIN macro should be the absolute name of your zone (trailing dot)
- A SOA record (see examples in the preparations). Use a low refresh value.
- An NS entry pointing at the nameserver for the zone. This should be the specific host ns.<x>.experiment.xen.netlab.csc.kth.se.
- An A record matching the name-server's IP address

You will know that you have the correct *syntax* of your zone file when tail /var/log/syslog says that the file is loaded correctly with a new serial number.

Then you can verify that your server is part of the DNS tree by using dig. For example:

```
dig @<your ip address> ns.<x>.experiment.xen.netlab.csc.kth.se
dig <x>.experiment.xen.netlab.csc.kth.se
dig ns.<x>.experiment.xen.netlab.csc.kth.se
dig ns.<x>.experiment.xen.netlab.csc.kth.se +trace
```

Things to think about when editing the zone file:

- The zone file is quite picky about spaces and newlines. Write as it looks in the examples.
- Increment the serial number each time you save the zone file. Emacs can do this for you automatically (but you may wish to switch to text-mode)
- *Avoid* blank names (lines starting with space): they represent repetitions. If you move a line with a blank name, it may get a different name, making the order significant.
- Make rndc reload after every change and check tail /var/log/syslog!
- Do you use the correct IPv4 address?
- Does /etc/bind/named.conf.local really contain the name of the correct zone file?
- Are you using the correct zone name (e.g. a1.experiment.xen.netlab.csc.kth.se)?
- Do you have an A record for your NS record?
- Sometimes (when nothing else helps) you just have to restart named using “service bind9 restart” and check with “rndc status”. This happens now and then if named got into a strange state.

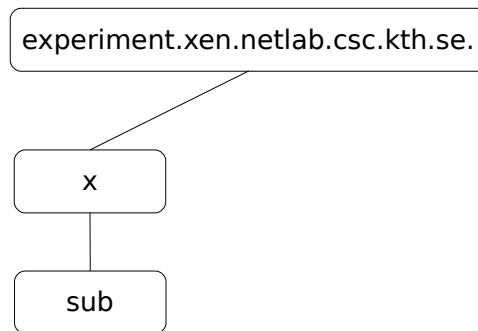
6 Adding records to your zone

A zone is rather uninteresting unless there is some data in it. Therefore you should add the following records to your zone:

- A CNAME for your server
- A TXT record with your name or some other unique text which you can use when debugging.

Check the new records with dig.

7 Creating a sub-zone



It is common for the same DNS server to answer for several zones. In this case you will create a subzone named `sub.<x>.experiment.xen.netlab.csc.kth.se.` You do this simply by creating a new zone file and adding the zone to your bind config. Make sure you add NS and A records to the zone.

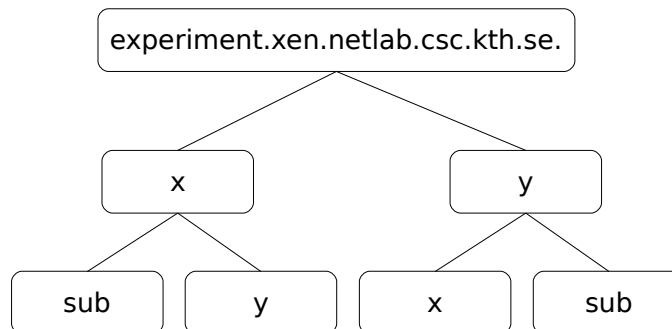
Note that you do not need to define glue records for the delegation to work. Verify that you can reach the zone.

Define a different TXT file for the sub-zone.

Show with dig your working zones, for example your TXT fields.

Milestone 1: Zones Signature: _____

8 Delegating a sub-zone



Frequently you wish to delegate a sub-zone to another server (and administrator). For example `csc.kth.se.` delegates `netlab.csc.kth.se.` to the lab. You should now do a delegation to one of the other groups. Assuming the other group is `y` you should delegate `<y>.<x>.experiment.xen.netlab.csc.kth.se` to the other group.

For the delegation to work correctly you need to add two records (NS and A) to your zone file. These are called glue records.

Why are glue records needed?

Glue records are required when you wish to set the name servers of a domain name to a hostname under the domain name itself.

When you delegate, you should assume that the nameserver in the delegated zone is called ns.<y>.<x>.experiment.xen.netlab.csc.kth.se.

Which are your glue records?

71 IN NS ns.71

ns.71 IN A 193.11.23.34

The glue records breaks one important principle in DNS. What is that?

We are breaking the tree : when we try to reach x or y we are going on the wrong way ! x should be on the other part of the tree circular dependency !

Once you have done the delegation, make sure your partner group delegates <x>.<y>.experiment.netlab.csc.kth.se to you in return.

For your server to answer for the zone you must accept the delegation by creating a Zone file matching <x>.<y>.experiment.netlab.csc.kth.se. Add NS and A records to this file.

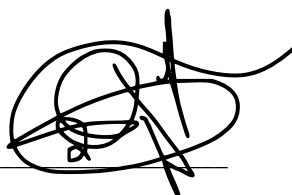
Also add a TXT records to so you can demonstrate that it works correctly.

To complete this milestone you should have:

- 1: dig +trace'es showing your zones working**
- 2: dig's illustrating the data in your zones**

Milestone 2: Sub-zone

Signature: _____



9 Masters and slaves, replication

At this point your DNS infrastructure is very vulnerable as the failure of a single server would make it all stop working. The way DNS solves this problem is through replication of the data over several redundant servers.

Replication requires several steps:

1. All nameservers for a zone must be listed in the zonefile (and in the delegation)
2. The master must send notify messages and accept zone transfers to the slaves
3. The slave must accept notify messages, initiate zone transfers and respond for the zone.

The parent zone, experiment.xen.netlab.csc.kth.se has been setup to return two separate nameservers for your zone: Your nameserver (ns.<x>) and your partner's nameserver (ns.<y>). This means that your zone currently has a secondary nameserver which does not respond correctly.

To replicate your zone to your partners zone you must:

1. List his nameserver among the authoritative nameservers in your zonefile for <x>
2. Add a "also-notify" entry to your bind configuration for the zone
3. Ask your neighbour to add a slave entry for the zone, with your server as the master.

To act as a slave for your neighbour you must:

1. Configure named to be a slave of <y>. If you do not specify a directory for the file, it will be stored in /var/cache/bind/ by default.

Use `rndc` (or restart `bind`) to get `bind` to notify your slave of the zone and initiate a zone transfer. You can check the `syslog` to see if the zone transfer was successful. After that you can use `"dig <x> @ns.<y>"` to check that your neighbour answers correctly for your zone.

10 Verification of zones

The final proof of your DNS setup is to pass the IIS test at <http://dnscheck.iis.se/>. Use this tool to verify all your zones and correct any errors that show up.

Note that DNSCheck will cache your results for ca 5 minutes, so if you run the tool again on the same zone you will just get the same report again.

Milestone 3: Working, redundant DNS! Signature: _____

To complete this milestone you should have

1. `dig +trace` results illustrating how the two nameservers answers to queries and
2. DNSCheck results for all your zones, without any errors

References

- [1] Forouzan, "TCP/IP protocol Suite", Chapter 17 Domain name System (DNS)
- [2] P. Mockapetris, "RFC 1034", Domain Names - Concepts and Facilities, IETF, 1987
- [3] P. Mockapetris, "RFC 1035", Domain Names - Implementations and Specifications, IETF, 1987
- [4] Bind 9 administrator reference manual on the web. Chapter 3: Nameserver configuration provides some examples of zone files and how to use `dig` and `rndc`.