# Lab 3

This lab has two parts. In the first part, you will configure a local DNS server using the mininet VM. In the second part, you will get familiar with socket programming in C by slightly modifying a server program and a client program.

## Part 1

In this part, you will configure a local DNS server using the mininet virtual machine. This DNS server will be able to resolve local DNS queries and forward other DNS queries to the school's DNS server.

1) Power on your mininet virtual machine and make a SSH connection to it. (Refer to the first lab if necessary).
2) Then update your VM's software by running the following command:

```
mininet@mininet-vm:~$ sudo apt-get update
```

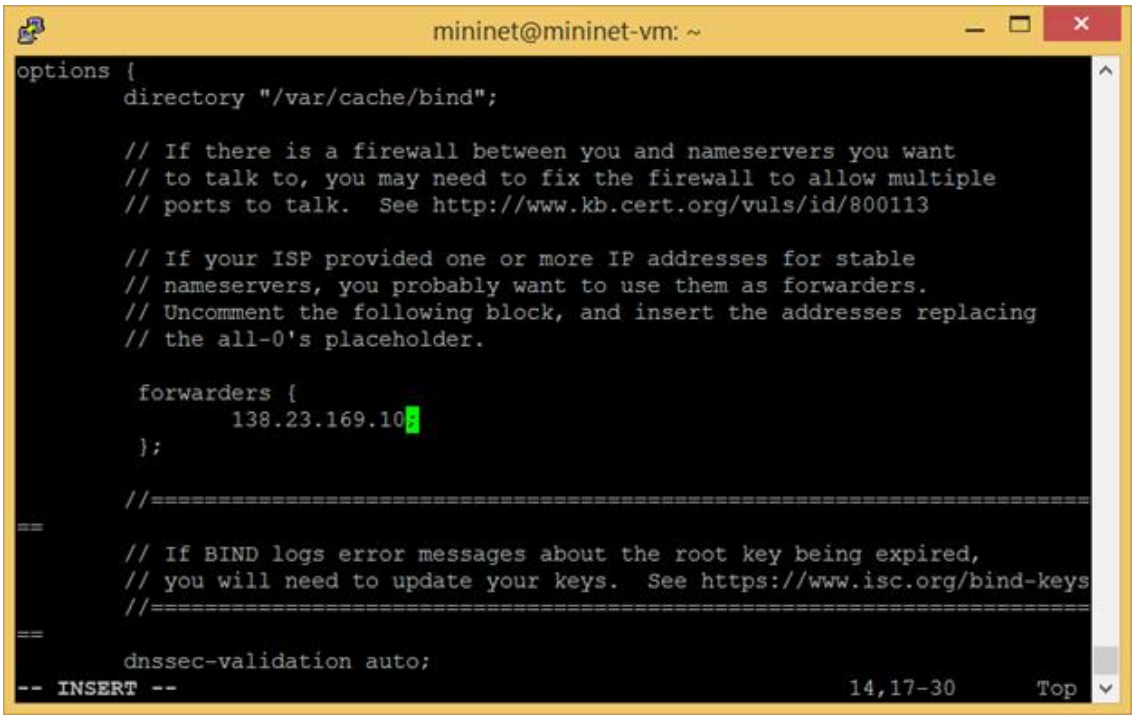
3) Get the bind9 DNS software by using the following command:

```
mininet@mininet-vm:~$ sudo apt-get install bind9
```

4) To configure bind9, we should edit "/etc/bind/named.conf.options"

```
 7  sudo vim /etc/bind/named.conf.options
```

5) In this file, uncomment the forwarders part, and add the IP address of our department's DNS server to it. The queries will be forwarded to this DNS server. You can find an example of the updated file below. NOTE: instead of the address "138.23.169.10", you need to enter "169.235.30.11".



6) Save the changes in the file and exit.
7) Now we edit the “/etc/bind/named.conf.local” file:

```
8  sudo vim /etc/bind/named.conf.local
```

8) We need to add a new zone to this file. The name for this zone will be “yourFirstName.CS164”, The following example is for a student with “Ali” as his first name.

```
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
zone "ali.cs164" {
        type master;
        file "/etc/bind/db.ali.cs164";
};
~
```

9) Now make a copy of the local DB file by using the following command (make sure to insert your name as appropriate):

```
sudo cp /etc/bind/db.local /etc/bind/db.ali.cs164
```

10) Before editing this file, you will need the hostname of your VM. You can find it by using the following command. It will probably be “mininet-vm”.

```
hostname
```

11) Now, edit the file we copied:

```
sudo vim /etc/bind/db.ali.cs164
```

12) Edit this file in a similar way to the following image (mininet-vm was our hostname, ali.cs164 was our defined zone, and ali.mohammadkhan.net is representing the Email address of administrator. PLEASE NOTICE the DOTs at the end of the URLs.

```
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     mininet-vm.ali.cs164. ali.mohammadkhan.net. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      mininet-vm.ali.cs164.
@       IN      A       127.0.0.1
@       IN      AAAA    ::1

mininet-vm      IN      A       10.0.2.15

server  IN      CNAME   mininet-vm.ali.cs164.
client  IN      CNAME   mininet-vm.ali.cs164.
~
~
```

13) Restart the bind service with this command:

```
sudo service bind9 restart
```

```
mininet@mininet-vm:~$ sudo service bind9 restart
 * Stopping domain name service... bind9
waiting for pid 22207 to die
                                                                    [ OK ]
 * Starting domain name service... bind9                           [ OK ]
```

14) If you don’t see two “OK” signs, you have made a mistake doing the previous steps. The following commands may help you to debug the problem:

```
mininet@mininet-vm:~$ sudo named-checkconf
mininet@mininet-vm:~$ named-checkzone ali.cs164 /etc/bind/db.ali.cs164
zone ali.cs164/IN: loaded serial 2
OK
```

15) After getting both “OK” signs in restarting bind9, you can double check it by sending the following DNS queries to the local DNS server.

```
mininet@mininet-vm:~$ nslookup www.google.com localhost
Server:         localhost
Address:        127.0.0.1#53

Non-authoritative answer:
Name:   www.google.com
Address: 216.58.216.36

mininet@mininet-vm:~$ nslookup ali.cs164 localhost
Server:         localhost
Address:        127.0.0.1#53

Name:   ali.cs164
Address: 127.0.0.1

mininet@mininet-vm:~$ nslookup server.ali.cs164 localhost
Server:         localhost
Address:        127.0.0.1#53

server.ali.cs164        canonical name = mininet-vm.ali.cs164.
Name:   mininet-vm.ali.cs164
Address: 10.0.2.15
```

16) To change the default DNS server of the system temporarily, we can edit the /etc/resolve.conf file and replace the current DNS servers with 127.0.0.1.

```
sudo vim /etc/resolv.conf
```

```
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 127.0.0.1
search cs.ucr.edu
~
~
~
```

17) Now run previous commands without specifying local DNS server to make sure that this change is applied:

```
mininet@mininet-vm:~$ nslookup www.google.com
Server:         127.0.0.1
Address:        127.0.0.1#53

Non-authoritative answer:
Name:   www.google.com
Address: 216.58.216.36

mininet@mininet-vm:~$ nslookup ali.cs164
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   ali.cs164
Address: 127.0.0.1

mininet@mininet-vm:~$ nslookup server.ali.cs164
Server:         127.0.0.1
Address:        127.0.0.1#53

server.ali.cs164        canonical name = mininet-vm.ali.cs164.
Name:   mininet-vm.ali.cs164
Address: 10.0.2.15
```

**Demo:** Your TA should be able to run the last series of queries (like in the above figure) and get the correct responses. You should be able to explain the meaning of any configuration changes you have made, from the first step to the last one.

# Part 2

In this part you are going to get familiar with socket programming in C. To make it easier, a client program and a server program are provided for you. (If you haven't already, you can download the source code from iLearn in the same place you found this lab.)

1) Use scp or "copy and paste" to move server.c and client.c to your mininet virtual machine. (In the example of scp below, amoha006 is my username and sierra-14 is my machine name and server.c was located in ~ on my machine.)

```
mininet@mininet-vm:~$ scp amoha006@sierra-14.cs.ucr.edu:server.c server.c
```

2) Compile, run, and investigate the code of the server and client to get familiar with it. (the command "*gcc server.c -o server*" compiles the source code "*server.c*" into an executable program called "server", which you can run by typing the command "*./server*").
3) Make changes in the client and server file so that it will work in the following way:

a) Client sends a message (like “Hi”) to the server based on user input (either as a command line argument or in response to a prompt from the client program) an server prints that message on his terminal.
b) Client uses the server’s name, server.yourFirstname.cs164, as the hostname which it uses to make the connection. In the example client and server files, the connection is made based on the IP address of the server and not the server’s name, so you have to update these files to satisfy this requirement. HINT: For mapping the name to IP address, you can use ‘gethostbyname’ function (an example is provided for you in gethostbynameExample.c).

**Demo:** A working client and server. Also you should be able to explain the role of each line in the client and server program.

Good Luck!