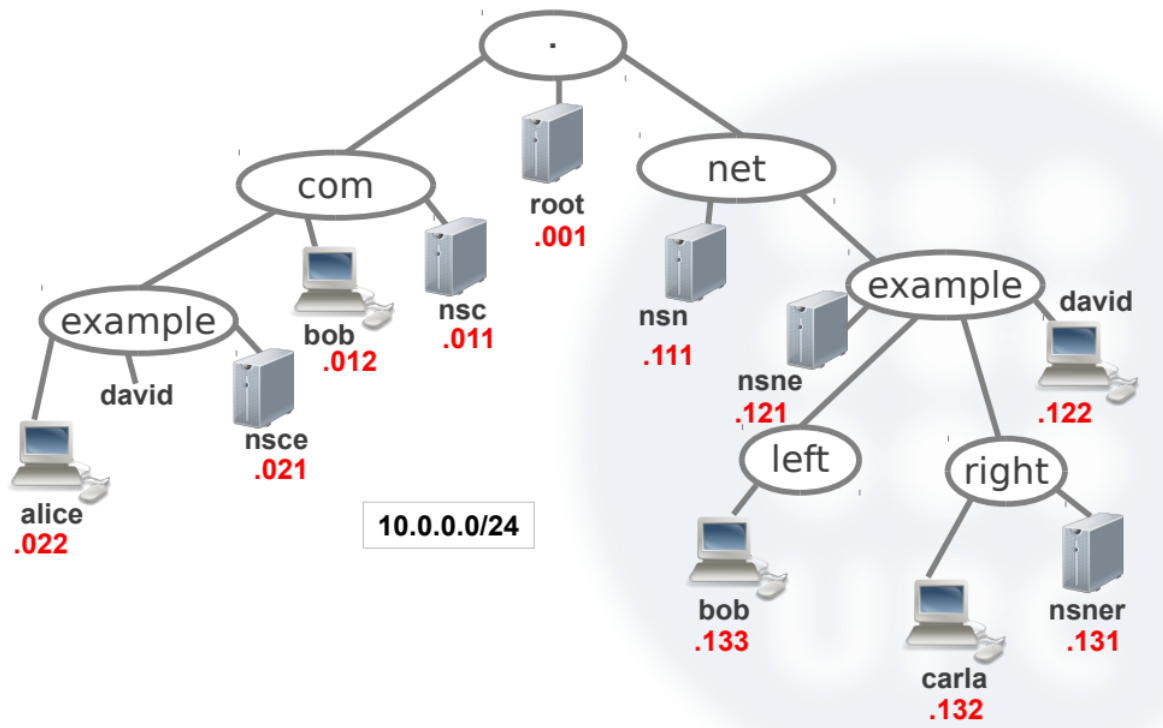


## P4: DNS



**Exercise 1.1**– In this exercise, we are going to study the DNS service using the scenario of Figure 1.15. Before the practice you must answer some questions. To answer consider that ROOT-SERVER is the master of the reverse zone and that it delegates .com to nsc.com and .net to nsn.net. Consider also that nsc.com delegates example.com to nsce.example.com and that nsn.net delegates example.net to nsne.example.net. Finally, consider that nsne.example.net must be configured with a single zone for the zone example.net (single configuration file) and that it must delegate right.example.net to nsner.right.example.net. According to the previous considerations about our DNS tree, explain in which server we should find the following resource records (RR):

- An A record for peter.example.com  
**nsce**
- An A record for peter.left.example.net  
**nsne**
- An A record for peter.right.example.net  
**nsner**
- The SOA record of right.example.net  
**nsner**
- A PTR record for peter.example.com //all the ptr are located in root server  
**ROOT-SERVER**
- A MX record for right.example.net  
**nsner**
- A MX record for left.example.net

```
nsne,  
• A NS record for right.example.net  
nsne,nsner
```

```
//en terminal  
simctl dns-basic sh  
start  
exec initial
```

1. Get a console at alice and looking at the configuration explain which is the name server used by this host.

```
//en alice  
host alice
```

```
alice:~# host alice  
alice.example.com has address 10.0.0.22  
alice:~#
```

```
dig alice
```

```
alice:~# dig alice  
  
; <>> DiG 9.6-ESV-R4 <>> alice  
;; global options: +cmd  
;; Got answer:  
;; ->HEADER<- opcode: QUERY, status: NXDOMAIN, id: 47453  
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0  
  
;; QUESTION SECTION:  
;alice.                IN      A  
  
;; AUTHORITY SECTION:  
;                        0      IN      SOA      ROOT-SERVER. admin-mail.ROOT-SERVER.  
2006031201 28800 14400 3600000 0  
  
;; Query time: 44 msec  
;; SERVER: 10.0.0.21#53(10.0.0.21)  
;; WHEN: Sat Oct 22 11:36:22 2022  
;; MSG SIZE rcvd: 80
```

the server used by host is nsce (10.0.0.21)

2. Identify which is the server of the zone example.com and describe the configuration of this zone.

```
//from nsce  
cat /etc/bind/named.conf  
cat /etc/bind/db.com.example
```

```

nsce:~# cat /etc/bind/db.com.example
; /etc/bind/db.com.example
$ORIGIN example.com.
$TTL      60000
@          IN      SOA     nsce  admin-mail.nsce (
2006031201 ; serial
28 ; refresh
14 ; retry
3600000 ; expire
20 ; 20 secs of negative cache ttl
)
@          IN      NS      nsce      ; unqualified name
nsce       IN      A       10.0.0.21
david      IN      CNAME   david.example.net.
@          IN      MX      10 mailserver1
@          IN      MX      20 mailserver2.example.com.
alice      IN      A       10.0.0.22
mailserver1 IN      A       10.0.0.25
mailserver2 IN      A       10.0.0.26

```

the server of the zone example.com with alice is nsce. Alice and nsce are the devices form this zone and david is an alias for the david.example.net.

\$ORIGIN example.com, is used to add unqualified names(those which aren't FQDN,(ended without .)). i.e : nsce = nsce.example.com

@ = same as origin

CNAME= alias

MX= mail servers of the domain

3. In nsce using the command netstat, identify the name of the DNS server process.

//en nsce

netstat -ulpn -# u cz now dns is udp

```

nsce:~# netstat -ulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
udp        0      0 0.0.0.0:2048            0.0.0.0:*               *          946/rpc.statd
udp        0      0 127.0.0.1:53            0.0.0.0:*               *          1167/named
udp        0      0 10.0.0.21:53            0.0.0.0:*               *          1167/named
udp        0      0 0.0.0.0:698             0.0.0.0:*               *          946/rpc.statd
udp        0      0 0.0.0.0:111             0.0.0.0:*               *          935/portmap

```

DNS port id UDP 53 so we can see the dns server process are named with PID 1167.

4. In this exercise, we analyze a simple query from alice. In first place, reset the name servers processes and then, capture with wireshark tap0 and explain the output of the following command:

//en terminal

exec resetbind

```

dns-basic-> exec resetbind
Virtual machines group: nsner nsne nsn nsce nsc joker root
OK The command has been started successfully.
OK The command has been started successfully.
OK The command has been started successfully.
OK The command has been started successfully.
OK The command has been started successfully.
OK The command has been started successfully.
OK The command has been started successfully.
Total time elapsed: 1 seconds

```

//en alice  
dig alice.example.com

```

alice:~# dig alice.example.com

; <<>> DiG 9.6-ESV-R4 <<>> alice.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 263
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;alice.example.com.                IN      A

;; ANSWER SECTION:
alice.example.com.        60000    IN      A      10.0.0.22

;; AUTHORITY SECTION:
example.com.              60000    IN      NS      nsce.example.com.

;; ADDITIONAL SECTION:
nsce.example.com.         60000    IN      A      10.0.0.21

;; Query time: 31 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Fri Oct 7 19:08:53 2022
;; MSG SIZE rcvd: 86

```

flags qr= query, aa= authoritative, rd= recursion desired, ra=(recursion available)

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.22
2	0.000482930	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
3	0.000801553	10.0.0.22	10.0.0.21	DNS	77	Standard query 0x0107 A alice.example.com
4	0.002846980	10.0.0.21	10.0.0.22	DNS	128	Standard query response 0x0107 A alice.example.com A 10.0.0.22 NS nsce.example.com A 10.0.0.21
5	0.011001154	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	Who has 10.0.0.22? Tell 10.0.0.21
6	0.011278612	fe:fd:00:00:08:01	fe:fd:00:00:04:01	ARP	42	10.0.0.22 is at fe:fd:00:00:08:01

5. Using dig, try to resolve the IP address of joker.example.com. Did you find any resolution for this name? Discuss the results.

//from alice  
dig joker.example.com

```

alice:~# dig joker.example.com

; <<>> DiG 9.6-ESV-R4 <<>> joker.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 754
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;joker.example.com.                IN      A

;; AUTHORITY SECTION:
example.com.              20      IN      SOA      nsce.example.com. admin-mail.nsce.example.com. 2006031201 28 14 3600000 20

;; Query time: 35 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Fri Oct 7 19:41:31 2022
;; MSG SIZE rcvd: 87

```

The server messages us NXDOMAIN, which means the translation for jokers is not found.

We can also observe the SOA and conclude that the nsce has the maximum information of joker and the TTL is 20.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.22
2	0.000428573	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
3	0.000750148	10.0.0.22	10.0.0.21	DNS	77	Standard query 0x02f2 A joker.example.com
4	0.001899479	10.0.0.21	10.0.0.22	DNS	129	Standard query response 0x02f2 No such name A joker.example.com SOA nsce.example.com
5	0.001283125	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	Who has 10.0.0.22? Tell 10.0.0.21
6	0.002285345	fe:fd:00:00:08:01	fe:fd:00:00:04:01	ARP	42	10.0.0.22 is at fe:fd:00:00:08:01

**There is no resolution for this name because in example.com. There is no joker. We can see the dns query and query response frame.**

6. Add the adequate RR in the appropriate server to map the name `joker.example.com` to the IP address `10.0.0.201`. Reset the name server to load the configuration and explain how you test your configuration with `dig` and `ping`.

**//from nsce**

**cd /etc/bind**

**cat name.conf**

```
zone "example.com" {
    type master;
    file "/etc/bind/db.com.example";
};
```

**nano db.com.example**

```

nsce.4153.0
File Edit View Search Terminal Help
GNU nano 2.0.7 File: db.com.example
; /etc/bind/db.com.example
$ORIGIN example.com.
$TTL 60000
@      IN      SOA     nsce  admin-mail.nsce (
2006031201 ; serial
28 ; refresh
14 ; retry
3600000 ; expire
20 ; 20 secs of negative cache ttl
)
@      IN      NS      nsce      ; unqualified name
nsce   IN      A       10.0.0.21
david  IN      CNAME   david.example.net.
@      IN      MX      10 mailserver1
@      IN      MX      20 mailserver2.example.com.
alice  IN      A       10.0.0.22
mailserver1 IN A       10.0.0.25
mailserver2 IN A       10.0.0.26
joker  IN      A       10.0.0.21
[ Wrote 19 lines ]

```

joker has ip 10.0.0.201

//from terminal  
**exec resetbind**  
**exec start**

//from alice  
**dig joker.example.com.**

```

alice:~# dig joker.example.com.

; <<> DiG 9.6-ESV-R4 <<> joker.example.com.
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 62057
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;joker.example.com.      IN      A

;; ANSWER SECTION:
joker.example.com.      60000   IN      A       10.0.0.21

;; AUTHORITY SECTION:
example.com.            60000   IN      NS      nsce.example.com.

;; ADDITIONAL SECTION:
nsce.example.com.       60000   IN      A       10.0.0.21

;; Query time: 27 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Fri Oct 14 19:04:51 2022
;; MSG SIZE rcvd: 86

```

Capturing from SimNet0						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.22
2	0.000267182	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
3	0.000441537	10.0.0.22	10.0.0.21	DNS	77	Standard query 0x6434 A joker.example.com
4	0.001721110	10.0.0.21	10.0.0.22	DNS	123	Standard query response 0x6434 A joker.example.com A 10.0.0.21 NS nsce.example.com A 10.0.0.21
5	0.007380542	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	Who has 10.0.0.22? Tell 10.0.0.21
6	0.007565415	fe:fd:00:00:08:01	fe:fd:00:00:04:01	ARP	42	10.0.0.22 is at fe:fd:00:00:08:01

Now there is a resolution for the joker.example.com

//from alice  
**ping -c 1 joker.example.com.**

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.0.22	10.0.0.201	ICMP	98	Echo (ping) request id=0xa04, seq=1/256, ttl=64 (reply in 4)
2	0.010631782	fe:fd:00:00:02:01	Broadcast	ARP	42	Who has 10.0.0.22? Tell 10.0.0.201
3	0.010827434	fe:fd:00:00:08:01	fe:fd:00:00:02:01	ARP	42	10.0.0.22 is at fe:fd:00:00:08:01
4	0.010958899	10.0.0.201	10.0.0.22	ICMP	98	Echo (ping) reply id=0xa04, seq=1/256, ttl=64 (request in 1)
5	111.340658757	fe:fd:00:00:08:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.22
6	111.340930061	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
7	111.341169776	10.0.0.22	10.0.0.21	DNS	77	Standard query 0x4495 A joker.example.com
8	111.341519577	10.0.0.21	10.0.0.22	DNS	128	Standard query response 0x4495 A joker.example.com A 10.0.0.21 NS nsce.example.com A 10.0.0.21
9	111.342709804	10.0.0.22	10.0.0.21	ICMP	98	Echo (ping) request id=0xa04, seq=1/256, ttl=64 (reply in 10)
10	111.342844789	10.0.0.21	10.0.0.22	ICMP	98	Echo (ping) reply id=0xa04, seq=1/256, ttl=64 (request in 9)
11	111.343389883	10.0.0.22	10.0.0.21	DNS	82	Standard query 0xea63 PTR 21.0.0.10.in-addr.arpa
12	111.373728482	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 10.0.0.1? Tell 10.0.0.21
13	111.373973329	fe:fd:00:00:01:01	fe:fd:00:00:04:01	ARP	42	10.0.0.1 is at fe:fd:00:00:01:01
14	111.376604429	10.0.0.21	10.0.0.1	DNS	93	Standard query 0xf264 PTR 21.0.0.10.in-addr.arpa OPT
15	111.378016330	10.0.0.1	10.0.0.21	DNS	164	Standard query response 0xf264 PTR 21.0.0.10.in-addr.arpa PTR nsce.example.com NS ROOT-SERVER A 10.0.0.1 OPT
16	111.378412590	10.0.0.21	10.0.0.1	DNS	70	Standard query 0x506d NS <Root> OPT
17	111.378755423	10.0.0.1	10.0.0.21	DNS	110	Standard query response 0x506d NS <Root> NS ROOT-SERVER A 10.0.0.1 OPT
18	111.380626875	10.0.0.21	10.0.0.22	DNS	153	Standard query response 0xea63 PTR 21.0.0.10.in-addr.arpa PTR nsce.example.com NS ROOT-SERVER A 10.0.0.1
19	116.351671728	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	Who has 10.0.0.22? Tell 10.0.0.21
20	116.351818091	fe:fd:00:00:08:01	fe:fd:00:00:04:01	ARP	42	10.0.0.22 is at fe:fd:00:00:08:01
21	116.381744251	fe:fd:00:00:01:01	fe:fd:00:00:04:01	ARP	42	Who has 10.0.0.21? Tell 10.0.0.1
22	116.381828747	fe:fd:00:00:04:01	fe:fd:00:00:01:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01

With dig we obtain the info of joker and we can do some requests. (con commando de dig -...)  
with ping we reach joker.

7. Which IP address will be contacted by a mail server if it has to send an e-mail to john@example.com.

//from nsce

cat /etc/bind/db.com.example

here we obtain the SOA file where we can see the 1st mailserver1 has ip 10.0.0.25

8. Try the following command: alice:~# dig -t MX example.com

```
alice:~# dig -t MX example.com

;>>> Dig 9.6-ESV-R4 <>> -t MX example.com
;; global options: +cmd
;; Got answer:
;;->HEADER<<- opcode: QUERY, status: NOERROR, id: 4310
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 1, ADDITIONAL: 3

;; QUESTION SECTION:
;example.com.                IN      MX

;; ANSWER SECTION:
example.com.                 60000   IN      MX      10 mailserver1.example.com.
example.com.                 60000   IN      MX      20 mailserver2.example.com.

;; AUTHORITY SECTION:
example.com.                 60000   IN      NS      nsce.example.com.

;; ADDITIONAL SECTION:
mailserver1.example.com.    60000   IN      A       10.0.0.25
mailserver2.example.com.    60000   IN      A       10.0.0.26
nsce.example.com.           60000   IN      A       10.0.0.21

;; Query time: 6 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Sat Oct 15 13:38:16 2022
;; MSG SIZE rcvd: 152
```

-t =type

Its give us information about which servers controls the MX(mail eXchanger)

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.0.22	10.0.0.21	DNS	71	Standard query 0x10d5 Mx example.com
2	0.000428974	10.0.0.21	10.0.0.22	DNS	194	Standard query response 0x10d5 Mx example.com MX 10 mailserver1.example.com MX 20 mailserver2.example.com NS nsce.example.com A 10.0.0.25 A 10.0.0.26 A 10.0.0.21

**Exercise 1.2–** Using the scenario dns-basic, in this exercise we analyze the recursive queries and the caching strategy of DNS.

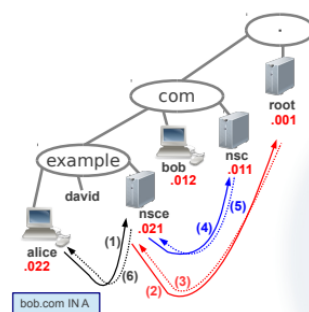
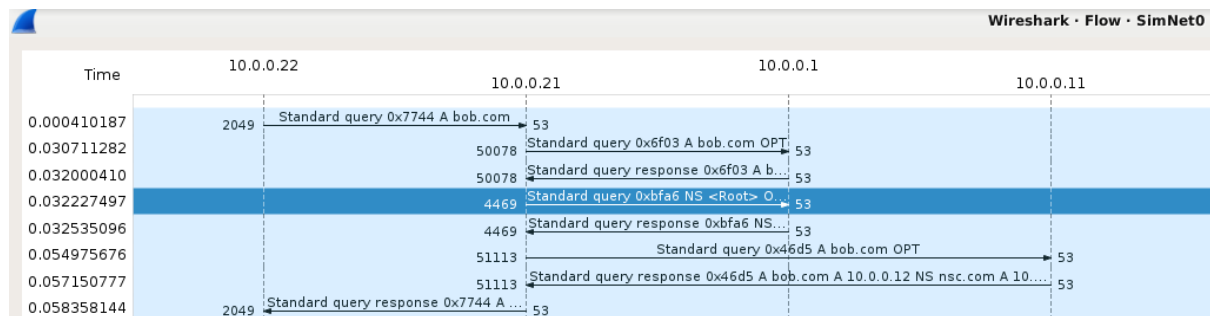
1. In this exercise, we analyze a recursive query from alice. To do so, reset the name servers processes of the scenario, capture with wireshark tap0 and explain the flow of DNS messages captured when executing the following command line: alice:~# dig bob.com

//from terminal

exec resetbind

```
//from alice
dig bob.com
```

Capturing from SimNet0						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.22
2	0.000262282	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
3	0.000410187	10.0.0.22	10.0.0.21	DNS	67	Standard query 0x7744 A bob.com
4	0.030219876	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 10.0.0.1? Tell 10.0.0.21
5	0.030461006	fe:fd:00:00:01:01	fe:fd:00:00:04:01	ARP	42	10.0.0.1 is at fe:fd:00:00:01:01
6	0.030711282	10.0.0.21	10.0.0.1	DNS	78	Standard query 0x6f03 A bob.com OPT
7	0.032000410	10.0.0.1	10.0.0.21	DNS	112	Standard query response 0x6f03 A bob.com NS nsc.com A 10.0.0.11 OPT
8	0.03227497	10.0.0.21	10.0.0.1	DNS	70	Standard query 0xbfa6 NS <Root> OPT
9	0.032535096	10.0.0.1	10.0.0.21	DNS	110	Standard query response 0xbfa6 NS <Root> NS ROOT-SERVER A 10.0.0.1 OPT
10	0.054468407	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 10.0.0.11? Tell 10.0.0.21
11	0.054740964	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	10.0.0.11 is at fe:fd:00:00:03:01
12	0.054975676	10.0.0.21	10.0.0.11	DNS	78	Standard query 0x46d5 A bob.com OPT
13	0.057150777	10.0.0.11	10.0.0.21	DNS	128	Standard query response 0x46d5 A bob.com A 10.0.0.12 NS nsc.com A 10.0.0.11 OPT
14	0.058358144	10.0.0.21	10.0.0.22	DNS	101	Standard query response 0x7744 A bob.com A 10.0.0.12 NS nsc.com
15	0.053451511	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	Who has 10.0.0.21? Tell 10.0.0.11
16	0.053651326	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
17	0.066914326	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	Who has 10.0.0.22? Tell 10.0.0.21
18	0.067029806	fe:fd:00:00:08:01	fe:fd:00:00:04:01	ARP	42	10.0.0.22 is at fe:fd:00:00:08:01



The flow of dns is following

- 1) **alice** sends a recursive query for the A register of bob.com to **nsce**.
- 2) **nsce** sends an iterative query for the A record of bob.com to **root**.
- 3) **root** sends query response for the A record of bob.com which is in nsc.com.
- 4) **nsce** also asks for the NS record of the **root**.
- 5) **root** provides its NS/A and the NS/A records of **nsc**, which is who serves .com
- 6) Then, **nsce** sends an iterative query to **nsc** for A record of bob.com
- 7) Then, **nsc** sends the A register of bob.com to **nsce**
- 8) Finally, **nsce** sends the A register of bob.com to **alice**

2. We analyze DNS caching in this exercise. To do so, reset the name servers processes of the scenario, capture with wireshark tap0 and explain the flow of DNS messages captured when executing the following command line:

```
//terminal
```

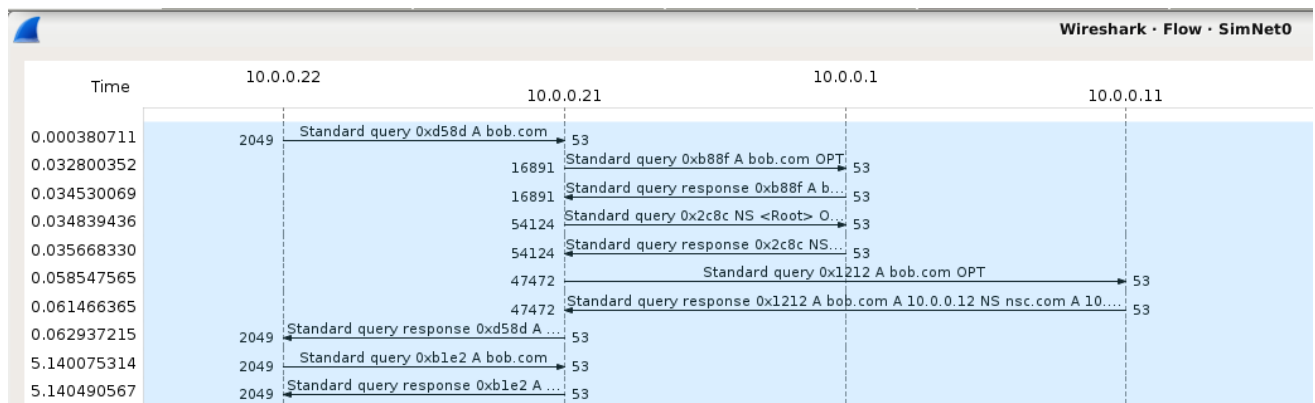
```
exec resetbind
```

```
dig bob.com ; sleep 5 ; dig bob.com
```



\*SimNet0

FileEditViewGoCaptureAnalyzeStatisticsTelephonyWirelessToolsHelp



We got here same flow as in previous exercise for the first dig, but for the second dig the query is directly answered by bob its because the ttl is not expired.

```

alice:~# dig bob.com

; <<>> DiG 9.6-ESV-R4 <<>> bob.com
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 47055
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;bob.com.                        IN      A

;; ANSWER SECTION:
bob.com.      30      IN      A      10.0.0.12

```

bob has ttl 30 seg.

3. Continuing with the analysis of DNS caching, reset the name servers processes of the scenario, capture with wireshark tap0 and explain the flow of DNS messages captured when executing the following command line:

//terminal

exec resetbind

//from alice

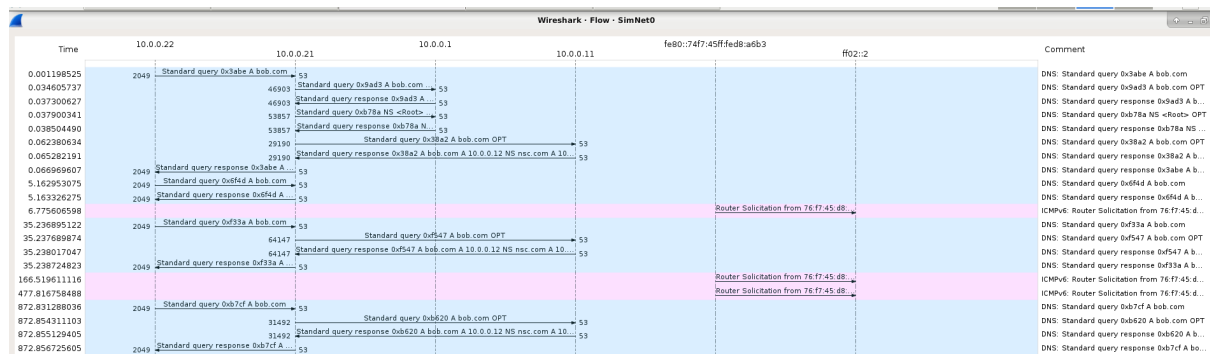
dig bob.com ; sleep 5 ; dig bob.com ; sleep 30 ; dig bob.com

Capturing from SimNet0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl>F

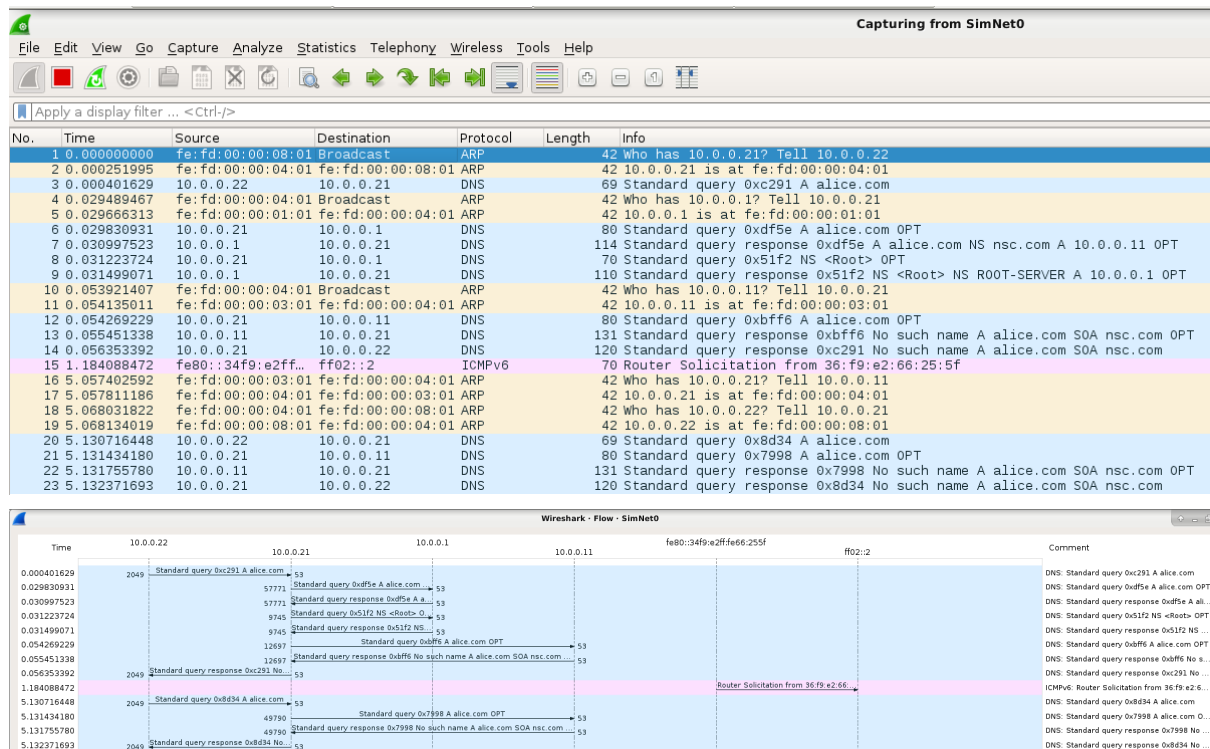
No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000000	fe:fd:00:00:08:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.22
2	0.000248786	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
3	0.000394933	10.0.0.22	10.0.0.21	DNS	67	Standard query 0x5144 A bob.com
4	0.021424367	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 10.0.0.11? Tell 10.0.0.21
5	0.021809419	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	10.0.0.11 is at fe:fd:00:00:03:01
6	0.022063718	10.0.0.21	10.0.0.11	DNS	78	Standard query 0x5868 A bob.com OPT
7	0.022565815	10.0.0.11	10.0.0.21	DNS	128	Standard query response 0x5868 A bob.com A 10.0.0.12 NS nsc.com A 10.0.0.11 OPT
8	0.023488425	10.0.0.21	10.0.0.22	DNS	101	Standard query response 0x5144 A bob.com A 10.0.0.12 NS nsc.com
9	0.028356116	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	Who has 10.0.0.22? Tell 10.0.0.21
10	0.028551606	fe:fd:00:00:08:01	fe:fd:00:00:04:01	ARP	42	10.0.0.22 is at fe:fd:00:00:08:01
11	0.039010792	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	Who has 10.0.0.21? Tell 10.0.0.11
12	0.039390728	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
13	0.098918453	10.0.0.22	10.0.0.21	DNS	67	Standard query 0xd5f7 A bob.com
14	0.099300456	10.0.0.21	10.0.0.22	DNS	101	Standard query response 0xd5f7 A bob.com A 10.0.0.12 NS nsc.com
15	0.164691926	10.0.0.22	10.0.0.21	DNS	67	Standard query 0x9a0f A bob.com
16	0.165737866	10.0.0.21	10.0.0.11	DNS	78	Standard query 0x5fba A bob.com OPT
17	0.166071047	10.0.0.11	10.0.0.21	DNS	128	Standard query response 0x5fba A bob.com A 10.0.0.12 NS nsc.com A 10.0.0.11 OPT
18	0.166755246	10.0.0.21	10.0.0.22	DNS	101	Standard query response 0x9a0f A bob.com A 10.0.0.12 NS nsc.com
19	0.138456541	fe:fd:00:00:08:01	fe:fd:00:00:04:01	ARP	42	Who has 10.0.0.21? Tell 10.0.0.22
20	0.138881256	fe:fd:00:00:04:01	fe:fd:00:00:08:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
21	0.159627524	fe:fd:00:00:03:01	fe:fd:00:00:04:01	ARP	42	Who has 10.0.0.21? Tell 10.0.0.11
22	0.160048848	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42	10.0.0.21 is at fe:fd:00:00:03:01



As we saw before, bob has ttl 30s so after that time the route that before was established is deleted and then, nsce has to send dns query to root again and do the same process of establishing a connection as in previous exercises.

4. Continuing with the analysis of DNS caching, reset the name servers processes of the scenario, capture with wireshark tap0 and explain the flow of DNS messages captured when executing the following command line:

```
//terminal
exec resetbind
//from alice
dig alice.com ; sleep 5 ; dig alice.com
```



The flow is similar as before, but now in query response we obtain SOA form nsc because alice.com is not FQDN and it doesn't exist. We also can see, the second dig hasn't reached to the root although error and its because the negative TTL is not defined.

5. Set the negative cache TTL to 10 in the SOA of nsc. Reset the name servers processes of the scenario, capture with wireshark tap0 and explain the flow of DNS messages captured when executing the following command line:

```
//from nsc
nano /etc/bind/db.com
```

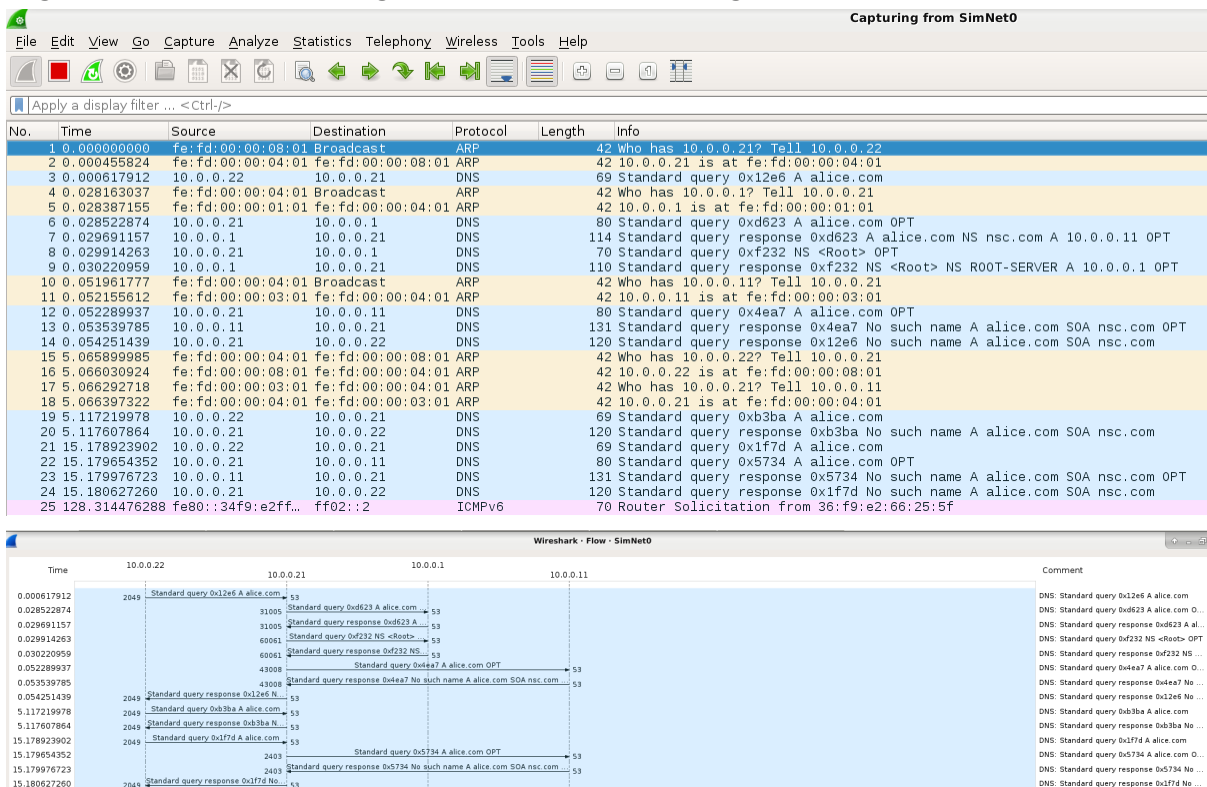
```

nsc.0
File Edit View Search Terminal Help
GNU nano 2.0.7 File: /etc/bind/db.com
$TTL 60000
com. IN SOA nsc.com. admin-mail.nsc.com. (
    2006031201 ; serial
    28800 ; refresh
    14400 ; retry
    3600000 ; expire
    10 ; negative cache ttl

```

//terminal  
exec resetbind

//from alice  
dig alice.com ; sleep 5 ; dig alice.com ; sleep 10 ; dig alice.com



First dig follows the same path as always, alice->nsce->root->nsce->nsc->nsce->alice  
The second one doesn't go through root because it's under 10s.  
The third dig goes from beginning because the ttl is 0 and the flow established before is deleted.

**Exercise 1.3– (\*)** Using the scenario dns-basic, you have to configure the name servers and zones of .net.

1. In your configuration consider that nsne must be configured with a single zone for example.net (single configuration file) and that it must delegate right.example.net to nsner.

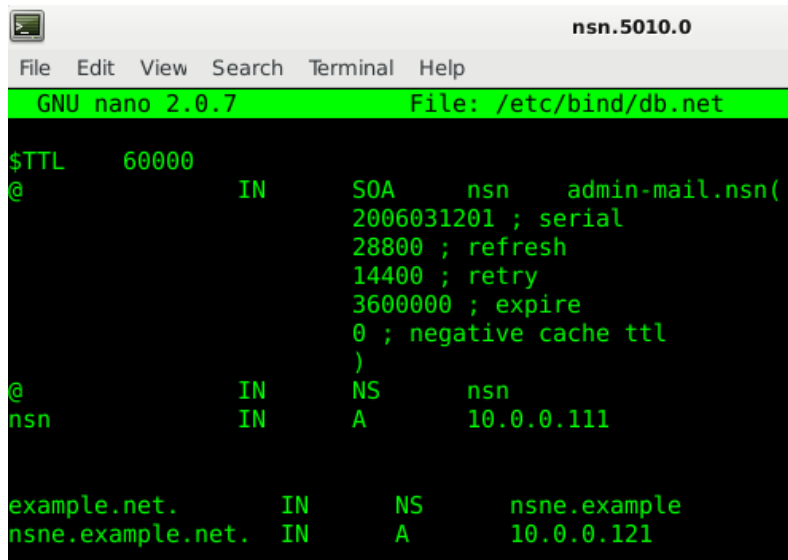
Modify the configuration files of nsn, nsne, and nsner appropriately and describe and test your configuration.

**//terminal**

**exec resetbind**

**//from nsn**

**nano /etc/bind/db.net**



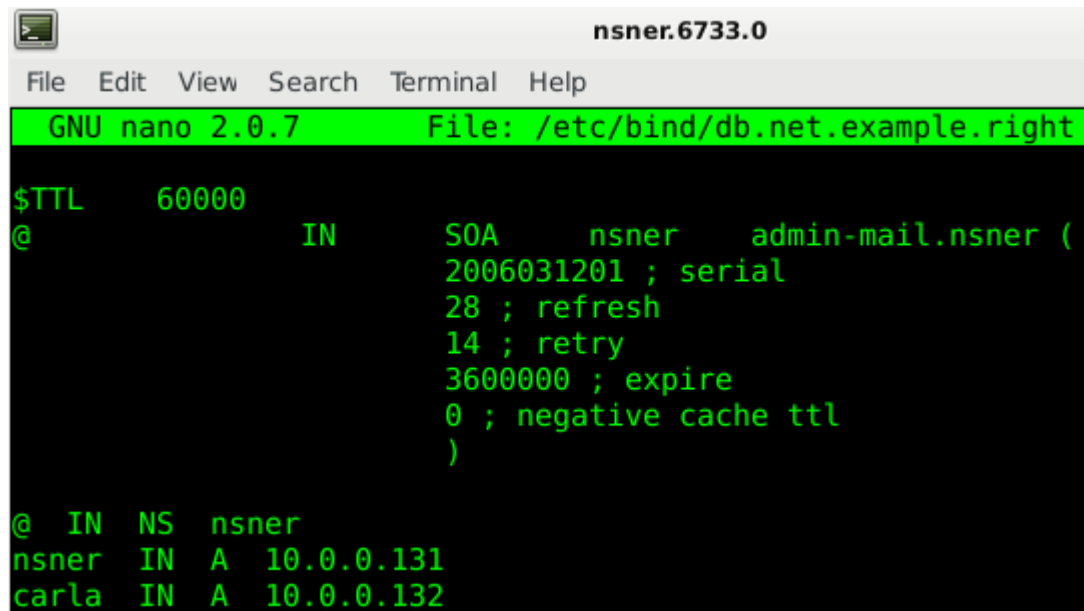
```
nsn.5010.0
File Edit View Search Terminal Help
GNU nano 2.0.7 File: /etc/bind/db.net

$TTL      60000
@          IN      SOA      nsn      admin-mail.nsn(
                2006031201 ; serial
                28800 ; refresh
                14400 ; retry
                3600000 ; expire
                0 ; negative cache ttl
                )
@          IN      NS       nsn
nsn        IN      A        10.0.0.111

example.net.      IN      NS       nsne.example
nsne.example.net. IN      A        10.0.0.121
```

**//from nsner**

**nano /etc/bind/db.net.example.right**



```
nsner.6733.0
File Edit View Search Terminal Help
GNU nano 2.0.7 File: /etc/bind/db.net.example.right

$TTL      60000
@          IN      SOA      nsner     admin-mail.nsner (
                2006031201 ; serial
                28 ; refresh
                14 ; retry
                3600000 ; expire
                0 ; negative cache ttl
                )
@ IN NS nsner
nsner IN A 10.0.0.131
carla IN A 10.0.0.132
```

**//from nsne**

**nano /etc/bind/db.net.example**

```

nsne.5867.0
File Edit View Search Terminal Help
GNU nano 2.0.7 File: /etc/bind/db.net.example

$TTL 60000
@           IN      SOA      nsne      admin-mail.nsne (
2006031201 ; serial
28 ; refresh
14400 ; retry
3600000 ; expire
15 ; negative cache ttl
)

@ IN NS nsne
nsne IN A 10.0.0.121
david IN A 10.0.0.122
bob.left 30 IN A 10.0.0.133

right IN NS nsner
nsner IN A 10.0.0.131

```

2. Notice that the server nsn has a mistake in its initial configuration file, describe this mistake.

initially there was a “.” missing at the end and the domains were interpreted as no qualified names.

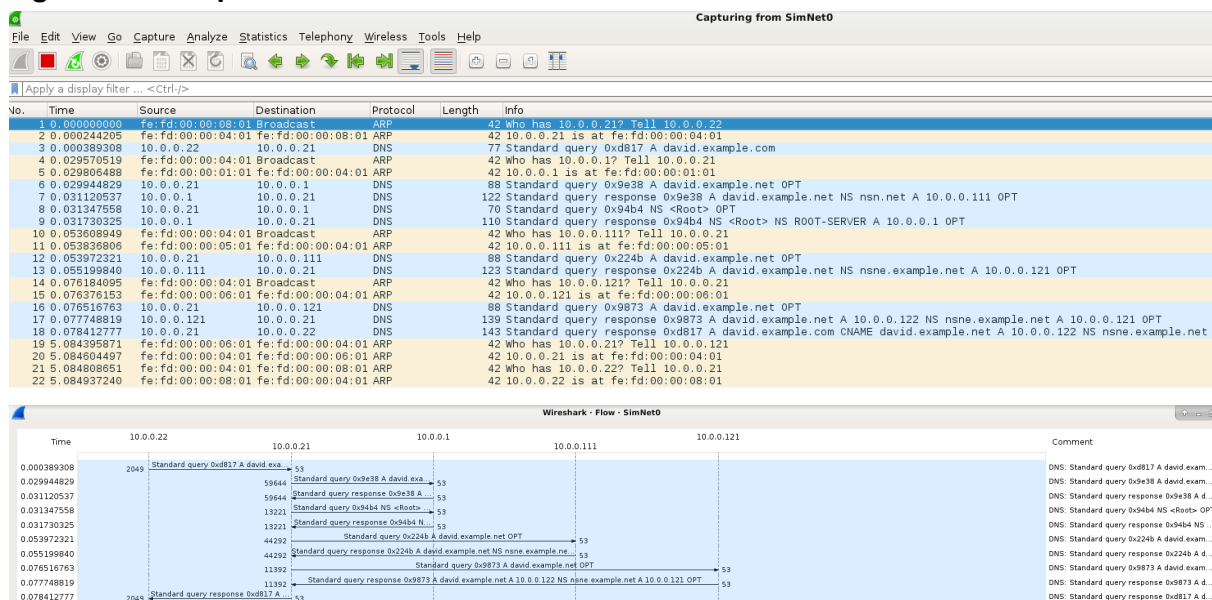
3. After you have implemented the configuration, reset bind in all the name servers of the scenario, capture with wireshark tap0 and comment the traffic and the results observed when executing:

//terminal

exec resetbind

//alice

dig david.example.com



The flow is the same, but the particularity here is at the end we received a query response, one of the domain example.net. and other of the domain example.com.

**Exercise 1.4– (\*)** In this exercise we study some more features of the DNS service using the same scenario.

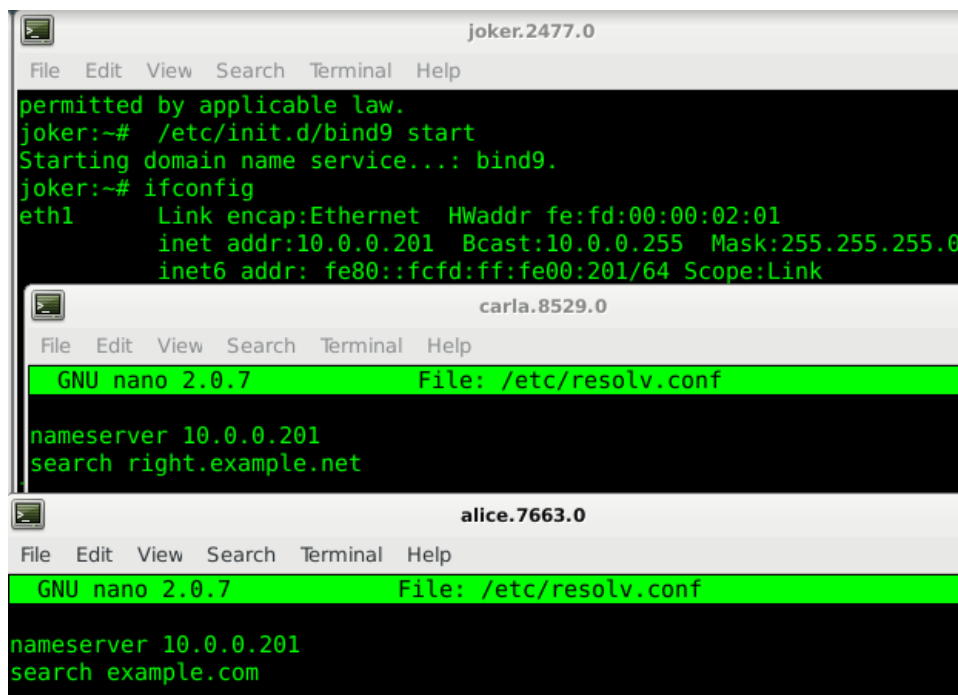
1. Use the machine joker as a DNS server just for caching and for making queries to the DNS tree on behalf of its clients. Make the clients alice and carla point to this server and test your configuration, for example asking for one register from alice and then, for the same register from carla.

```
//from joker
```

```
/etc/init.d/bind9 start
```

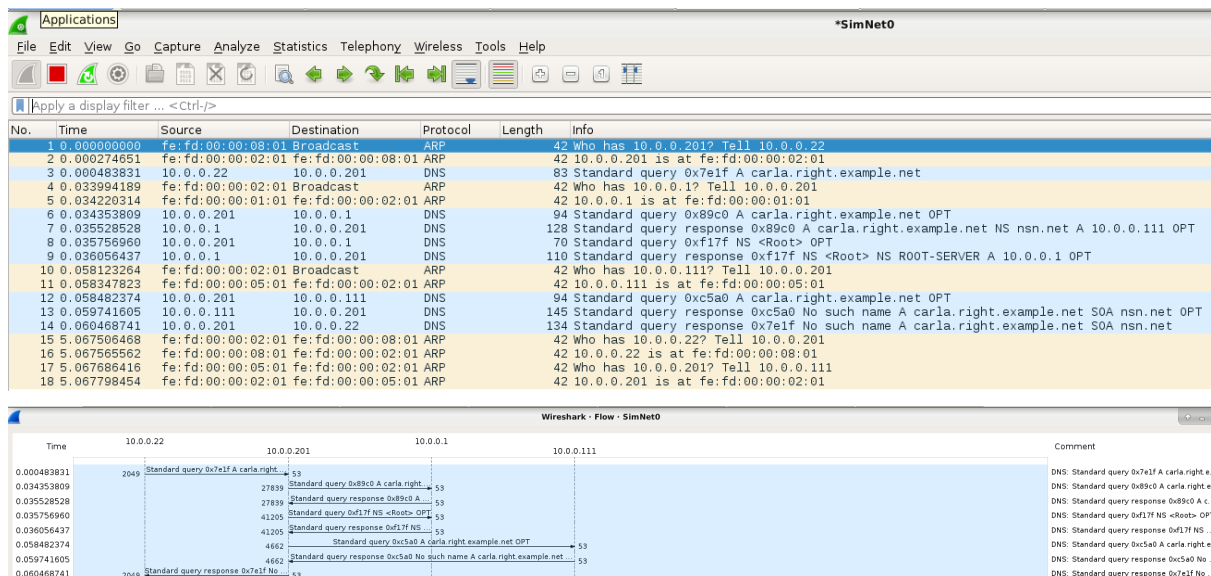
```
//from alice //from carla
```

```
ano /etc/resolv.conf
```



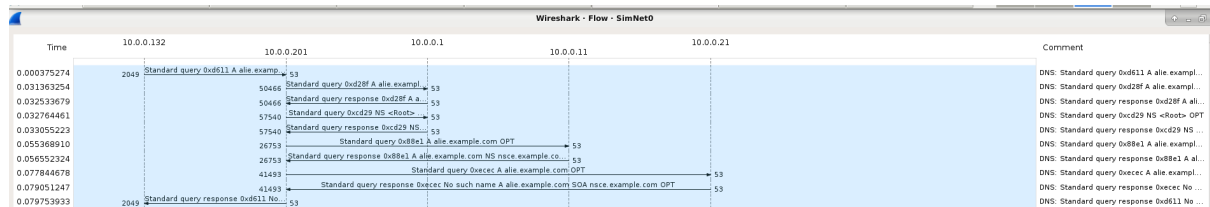
```
//from alice
```

```
dig carla.right.example.net
```



```
//from carla
dig alie.example.com
```

Capturing from SimNet0						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	fe:fd:00:00:09:01	Broadcast	ARP	42	Who has 10.0.0.201? Tell 10.0.0.132
2	0.000228768	fe:fd:00:00:02:01	fe:fd:00:00:09:01	ARP	42	10.0.0.201 is at fe:fd:00:00:02:01
3	0.000375274	10.0.0.132	10.0.0.201	DNS	76	Standard query 0xd611 A alie.example.com
4	0.030992219	fe:fd:00:00:02:01	Broadcast	ARP	42	Who has 10.0.0.1? Tell 10.0.0.201
5	0.031226646	fe:fd:00:00:01:01	fe:fd:00:00:02:01	ARP	42	10.0.0.1 is at fe:fd:00:00:01:01
6	0.031363254	10.0.0.201	10.0.0.1	DNS	87	Standard query 0xd28f A alie.example.com OPT
7	0.032533679	10.0.0.1	10.0.0.201	DNS	121	Standard query response 0xd28f A alie.example.com NS nsc.com A 10.0.0.11 OPT
8	0.032764461	10.0.0.201	10.0.0.1	DNS	70	Standard query 0xcd29 NS <Root> OPT
9	0.033055223	10.0.0.1	10.0.0.201	DNS	110	Standard query response 0xcd29 NS <Root> NS R00T-SERVER A 10.0.0.1 OPT
10	0.054954403	fe:fd:00:00:02:01	Broadcast	ARP	42	Who has 10.0.0.11? Tell 10.0.0.201
11	0.055151315	fe:fd:00:00:03:01	fe:fd:00:00:02:01	ARP	42	10.0.0.11 is at fe:fd:00:00:03:01
12	0.055368910	10.0.0.201	10.0.0.11	DNS	87	Standard query 0x88e1 A alie.example.com OPT
13	0.05552324	10.0.0.11	10.0.0.201	DNS	122	Standard query response 0x88e1 A alie.example.com NS nsce.example.com A 10.0.0.21 OPT
14	0.077546638	fe:fd:00:00:02:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.201
15	0.077708575	fe:fd:00:00:04:01	fe:fd:00:00:02:01	ARP	42	10.0.0.21 is at fe:fd:00:00:04:01
16	0.077844678	10.0.0.201	10.0.0.21	DNS	87	Standard query 0xecec A alie.example.com OPT
17	0.079051247	10.0.0.21	10.0.0.201	DNS	139	Standard query response 0xecec No such name A alie.example.com SOA nsce.example.com OPT
18	0.079753933	10.0.0.201	10.0.0.132	DNS	128	Standard query response 0xd611 No such name A alie.example.com SOA nsce.example.com
19	0.080720394	fe:fd:00:00:04:01	fe:fd:00:00:02:01	ARP	42	Who has 10.0.0.201? Tell 10.0.0.21
20	0.080908979	fe:fd:00:00:02:01	fe:fd:00:00:09:01	ARP	42	Who has 10.0.0.132? Tell 10.0.0.201
21	0.080969321	fe:fd:00:00:09:01	fe:fd:00:00:02:01	ARP	42	10.0.0.132 is at fe:fd:00:00:09:01
22	0.081122342	fe:fd:00:00:02:01	fe:fd:00:00:04:01	ARP	42	10.0.0.201 is at fe:fd:00:00:02:01



As we can see in both case, the joker does the function of the server who communicates with root to establish the path of alice and carla or vice versa.

2. Change the configuration to delegate the reverse lookup of all the IP addresses of the scenario to the machine joker. Describe how you test that your configuration is correct.