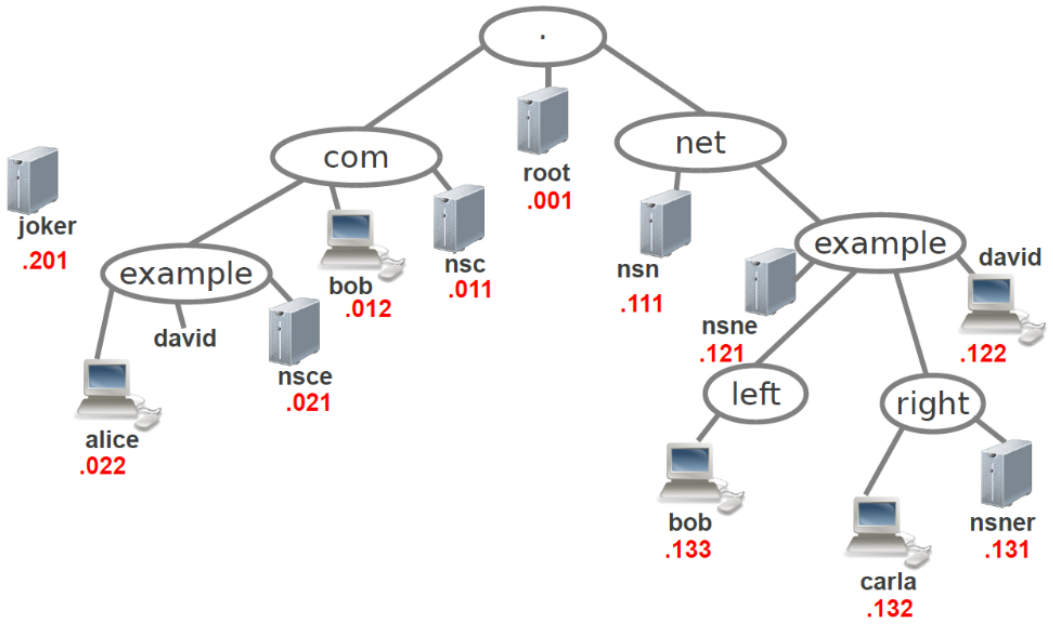
EXERCISE 1



. (root) delegates:

- .com --> nsc.com
- .net --> nsn.net

nsc.com delegates:

- example.com -> nsce.example.com

nsce.example.com delegates:

- (To any server) alice.example.com, david.example.com

nsn.net delegates:

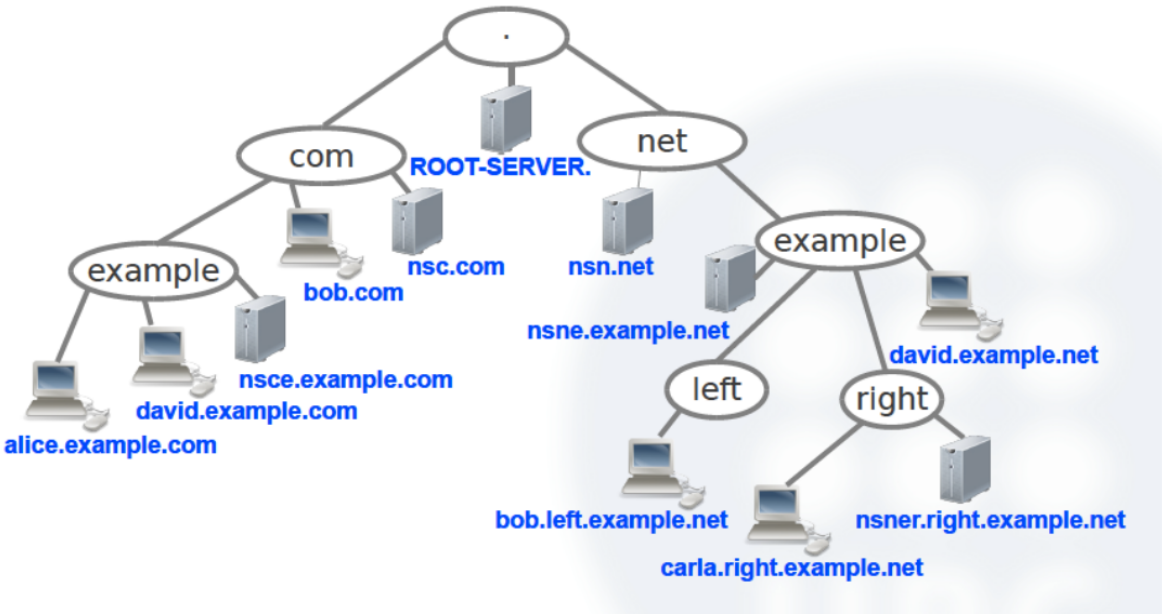
- example.net -> nsne.example.net

nsne.example.net delegates:

- (To any server) directament a david.example.net
- (To any server) directament a bob.left.example.net
- right.example.net -> nsner.right.example.net

nsner.right.example.net delegates:

- (To any server) directament a carla.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.example.com
- An A record for peter.left.example.net -> nsne.example.net
- An A record for peter.right.example.net -> nsner.right.example.net
- The SOA record of right.example.net -> nsner.right.example.net
- A PTR record for peter.example.com -> nsce.example.com
- A MX record for right.example.net -> nsner.right.example.net
- A MX record for left.example.net -> nsne.example.net
- A NS record for right.example.net -> nsner.right.example.net

Starting scenario:

```
phyhost$ simctl dns-basic start
```

Load the initial conf:

```
phyhost$ simctl dns-basic exec initial
```

To reset all the name servers processes to clear its caches and reload conf:

```
phyhost$ simctl dns-basic exec resetbind
```

1) Get a console at alice and looking at the configuration explain which is the name server used by this host.
   alice: ~# cat /etc/resolv.conf -> nameserver 10.0.0.21 (nsce)

```
alice:~# cat /etc/resolv.conf
nameserver 10.0.0.21
search example.com
```

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

As we see in the AUTHORITY SECTION (SOA RR) example.com server is nsce.example.com. Integrated for Alice i nsce, the zone nameserver. Joker is not included in the configuration and David is a reference to the example.net host.
To see the conf of the server-> nsce: ~# cat /etc/bind/named.conf
Zone conf-> nsce: ~# 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
```

3) In nsce using the command netstat, identify the name of the DNS server process.
nsce: ~# netstat –unlp

```
nsce:~# netstat -unlp
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:*                           1200/named
udp        0      0 10.0.0.21:53            0.0.0.0:*                           1200/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
```

As we see, our server is .21 through port 53. The PID is 1200 and its process name is named.
With nsce: ~# netstat –anpl, also appears the tcp connection.

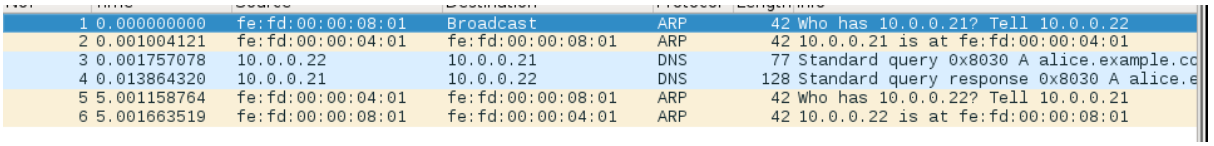
4) In this exercise, we analyze a simple query from alice. In first place, reset the name servers processes and then, capture with wireshark tap0 and explain the output of the following command:

```
alice:~# dig alice.example.com
```

“dig” sends a “query” of the alice.example.com and additionals RR of the nameserver.

```
alice:~# dig alice.example.com

; <<>> DiG 9.6-ESV-R4 <<>> alice.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32816
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL:

;; 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: 56 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Sat Mar 20 10:54:43 2021
;; MSG SIZE  rcvd: 86
```

SimNet0:



1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42 Who has 10.0.0.21? Tell 10.0.0.22
2	0.001004121	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	0.001757078	10.0.0.22	10.0.0.21	DNS	77 Standard query 0x8030 A alice.example.cc
4	0.013864320	10.0.0.21	10.0.0.22	DNS	128 Standard query response 0x8030 A alice.e
5	5.001158764	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	5.001663519	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

In wireshark its captured the broadcast ARP frame sended by Alice to connect with the nameserver and the DNS frame query with the information requested.

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

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: 26861
;; 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. 200603
1 28 14 3600000 20

;; Query time: 29 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Sat Mar 20 11:04:47 2021
;; MSG SIZE  rcvd: 87
```


https://drive.google.com/drive/folders/1ugbAayGfGTtjjVX0rLdREecVnr-qNXcN?usp=sharing

Cause of the unreachable host, we receive a SOA (error cases). The nsce server is who has more information about joker so it response with this error to finish the process.

SImNet0:

```
     8 604.272760811 fe:fd:00:00:08:01    Broadcast           ARP         42 Who has 10.0.0.21? Tell 10.0.0.22
     9 604.272969640 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
    10 604.273129079 10.0.0.22            10.0.0.21           DNS         77 Standard query 0x68ed A joker.example.cc
    11 604.273830969 10.0.0.21            10.0.0.22           DNS        129 Standard query response 0x68ed No such r
    12 609.285075288 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
    13 609.285289531 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
```

In wireshark its captured a DNS query but also a no such name error response from server.

6) Add the adequate RR in the appropriate server to map the name joker.example.com to the IP address 10.0.0.201.
In order to reach joker, we need to configure the type A joker RR in the nameserver:

nsce:~# nano /etc/bind/db.com.example -> joker IN A 10.0.0.201
dns-basic -> exec resetbind
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: 18578
;; 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       10.0.0.201

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

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

;; Query time: 30 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Sat Mar 20 15:13:21 2021
;; MSG SIZE  rcvd: 86
```

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 0.000348732 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 0.000547157 10.0.0.22           10.0.0.21           DNS         77 Standard query 0x4892 A joker.example.cc
     4 0.003246542 10.0.0.21           10.0.0.22           DNS        128 Standard query response 0x4892 A joker.e
     5 5.005202173 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 5.005423037 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 the server finds the joket.example.com host.

https://drive.google.com/drive/folders/1ugbAayGfGTtjjVX0rLdREecVnr-qNXcN?usp=sharing

7) Which IP address will be contacted by a mail server if it has to send an e-mail to john@example.com.
   As we can see in the configuration file of nsce server (/etc/bind/db.com.example) the @IP needed to send a mail from the server is 10.0.0.25 or .26.
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: 23342
;; 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      20 mailserver2.example.com.
example.com.            60000   IN      MX      10 mailserver1.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: 50 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Sat Mar 20 15:27:52 2021
;; MSG SIZE  rcvd: 152
```

It specifies some information about the mailservers.

EXERCISE 2 - DNS-basic (caching strategy)

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:

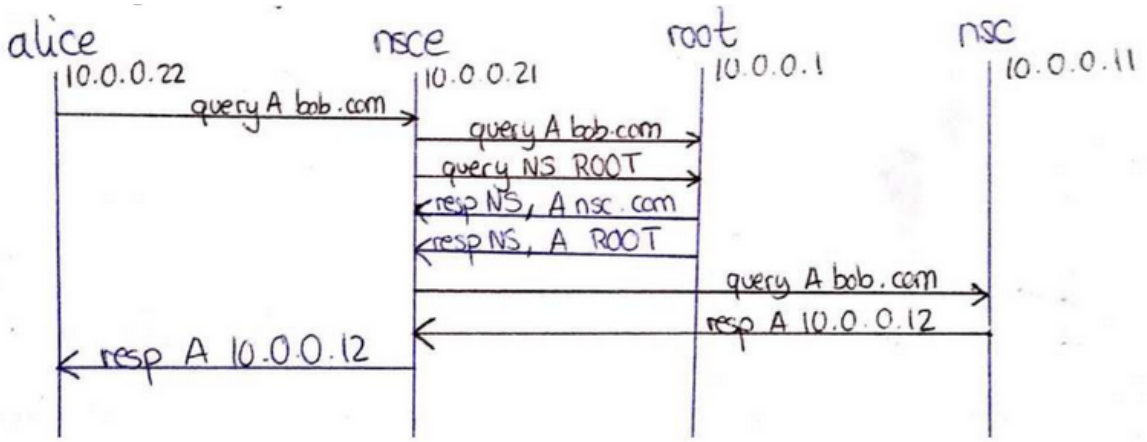
```
alice:~# dig bob.com
```

```
alice:~# dig bob.com

; <<>> DiG 9.6-ESV-R4 <<>> bob.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19952
;; 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

;; AUTHORITY SECTION:
com.                    60000   IN      NS      nsc.com.

;; Query time: 131 msec
;; SERVER: 10.0.0.21#53(10.0.0.21)
;; WHEN: Sat Mar 20 15:34:35 2021
;; MSG SIZE  rcvd: 59
```

SimNet0:

1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.22
2	0.000319128	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	0.000540113	10.0.0.22	10.0.0.21	DNS	67	Standard query 0x4df0 A bob.com
4	0.046720210	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 10.0.0.1? Tell 10.0.0.21
5	0.046937385	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	0.047132809	10.0.0.21	10.0.0.1	DNS	78	Standard query 0xbdca A bob.com OPT
7	0.047150670	10.0.0.21	10.0.0.1	DNS	70	Standard query 0xf9c5 NS <Root> OPT
8	0.050348345	10.0.0.1	10.0.0.21	DNS	112	Standard query response 0xbdca A bob.com
9	0.050745258	10.0.0.1	10.0.0.21	DNS	110	Standard query response 0xf9c5 NS <Root>
10	0.074074208	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 10.0.0.11? Tell 10.0.0.21
11	0.074275965	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	0.074466878	10.0.0.21	10.0.0.11	DNS	78	Standard query 0xb5e3 A bob.com OPT
13	0.077342106	10.0.0.11	10.0.0.21	DNS	128	Standard query response 0xb5e3 A bob.com
14	0.078374554	10.0.0.21	10.0.0.22	DNS	101	Standard query response 0x4df0 A bob.com
15	5.083645378	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
16	5.083828198	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
17	5.084239684	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
18	5.084381663	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

QUERY: Alice contacts with nsce-> nsce broadcast ARP to root cause it doesnt know wheres bob. RESPONSE: root says bob.com is in nsc. QUERY to nsc to know bob.com. RESPONSE from nsc to nsce with bob location-> nsce responds to alice with bobs.com @IP.



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:

slides 08-09

https://drive.google.com/drive/folders/1ugbAayGfGTtjjVX0rLdREecVnr-qNXcN?usp=sharing

```
alice:~# dig bob.com ; sleep 5 ; dig bob.com
```

SimNet0:

1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42 Who has 10.0.0.21? Tell 10.0.0.22
2	0.000238446	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	0.000422320	10.0.0.22	10.0.0.21	DNS	67 Standard query 0x913a A bob.com
4	0.046416360	fe:fd:00:00:04:01	Broadcast	ARP	42 Who has 10.0.0.1? Tell 10.0.0.21
5	0.046632427	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	0.046829094	10.0.0.21	10.0.0.1	DNS	78 Standard query 0xe4c9 A bob.com OPT
7	0.046842835	10.0.0.21	10.0.0.1	DNS	70 Standard query 0xb450 NS <Root> OPT
8	0.049979618	10.0.0.1	10.0.0.21	DNS	112 Standard query response 0xe4c9 A bob.com
9	0.050325413	10.0.0.1	10.0.0.21	DNS	110 Standard query response 0xb450 NS <Root>
10	0.074136563	fe:fd:00:00:04:01	Broadcast	ARP	42 Who has 10.0.0.11? Tell 10.0.0.21
11	0.074383527	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	0.074596976	10.0.0.21	10.0.0.11	DNS	78 Standard query 0x1ddc A bob.com OPT
13	0.077595021	10.0.0.11	10.0.0.21	DNS	128 Standard query response 0x1ddc A bob.com
14	0.078551914	10.0.0.21	10.0.0.22	DNS	101 Standard query response 0x913a A bob.com
15	5.079229831	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	5.079428707	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	5.089616616	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	5.089718780	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
19	5.268563729	10.0.0.22	10.0.0.21	DNS	67 Standard query 0xfe09 A bob.com
20	5.268992516	10.0.0.21	10.0.0.22	DNS	101 Standard query response 0xfe09 A bob.com

As in the previous exercise, we receive the same response, but now, after 5 seconds, alice sends the same query and there are not needed all the broadcast frame cause alice has bobs route in its cache list. So, alice only needs to send the DNS frame.

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:

```
alice:~# dig bob.com ; sleep 5 ; dig bob.com ; sleep 30 ; dig bob.com
```

Its the same as before. Only that after 30 seconds we dont see the ususal path to bob with the last dig.

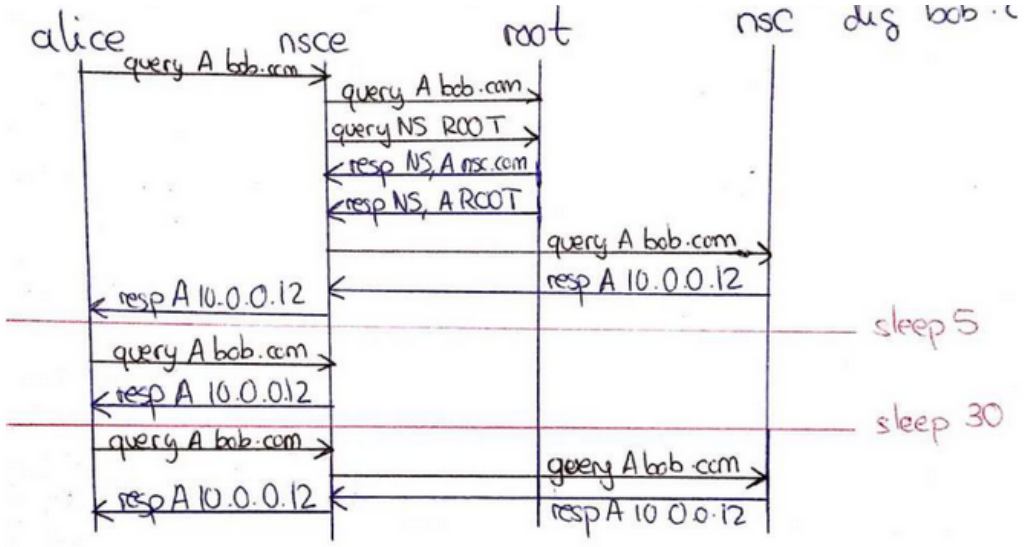
SimNet0:

1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42 Who has 10.0.0.21? Tell 10.0.0.22
2	0.000397766	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	0.000583629	10.0.0.22	10.0.0.21	DNS	67 Standard query 0x130b A bob.com
4	0.026222942	10.0.0.21	10.0.0.1	DNS	78 Standard query 0xf751 A bob.com OPT
5	0.029517786	10.0.0.21	10.0.0.1	DNS	70 Standard query 0x81f7 NS <Root> OPT
6	0.051181621	fe:fd:00:00:01:01	Broadcast	ARP	42 Who has 10.0.0.21? Tell 10.0.0.1
7	0.051703056	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
8	0.052842025	10.0.0.1	10.0.0.21	DNS	112 Standard query response 0xf751 A bob.com
9	0.052885318	10.0.0.1	10.0.0.21	DNS	110 Standard query response 0x81f7 NS <Root>
10	0.058689406	10.0.0.21	10.0.0.11	DNS	78 Standard query 0x685d A bob.com OPT
11	0.085649861	fe:fd:00:00:03:01	Broadcast	ARP	42 Who has 10.0.0.21? Tell 10.0.0.11
12	0.085876579	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	0.086027171	10.0.0.11	10.0.0.21	DNS	128 Standard query response 0x685d A bob.com
14	0.086985917	10.0.0.21	10.0.0.22	DNS	101 Standard query response 0x130b A bob.com
15	5.095271316	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
16	5.095447785	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
17	5.257813395	10.0.0.22	10.0.0.21	DNS	67 Standard query 0xb204 A bob.com
18	5.258253155	10.0.0.21	10.0.0.22	DNS	101 Standard query response 0xb204 A bob.com
19	35.410830094	10.0.0.22	10.0.0.21	DNS	67 Standard query 0xccc8 A bob.com
20	35.411720649	10.0.0.21	10.0.0.11	DNS	78 Standard query 0x8d9b A bob.com OPT
21	35.412116160	10.0.0.11	10.0.0.21	DNS	128 Standard query response 0x8d9b A bob.com
22	35.412783407	10.0.0.21	10.0.0.22	DNS	101 Standard query response 0xccc8 A bob.com
23	40.322912521	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
24	40.323038965	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
25	40.408512555	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42 Who has 10.0.0.11? Tell 10.0.0.21
26	40.409172591	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

In fact, if we look at the file /etc/bind/db.com in the nsc server, it says that bobs TTL is 30 seconds and thats why bobs route was deleted from nsc cache.
Anyway, the DNS frame doesnt pass through root cause its TTL is 60000 for everyone. So, its not needed to contact with root. The nsce server would have the bob route in its cache and the frame is sended directly to nsc.com.

https://drive.google.com/drive/folders/1ugbAayGfGTtjjVX0rLdREecVnr-qNXcN?usp=sharing

```
 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
                        0 ; negative cache ttl
                        )
com.                IN      NS      nsc.com.
nsc.com.            IN      A       10.0.0.11
bob.com.    30      IN      A       10.0.0.12

example.com.        IN      NS      nsce.example.com.
nsce.example.com.   IN      A       10.0.0.21
```



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:

```
alice:~# dig alice.com ; sleep 5 ; dig alice.com
```

1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42 Who has 10.0.0.21? Tell 10.0.0.22
2	0.000291041	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	0.002634356	10.0.0.22	10.0.0.21	DNS	69 Standard query 0x8d70 A alice.com
4	0.057577206	fe:fd:00:00:04:01	Broadcast	ARP	42 Who has 10.0.0.1? Tell 10.0.0.21
5	0.057847422	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	0.057999572	10.0.0.21	10.0.0.1	DNS	80 Standard query 0x77db A alice.com OPT
7	0.058011718	10.0.0.21	10.0.0.1	DNS	70 Standard query 0x0e2a NS <Root> OPT
8	0.061065912	10.0.0.1	10.0.0.21	DNS	114 Standard query response 0x77db A alice.c
9	0.061377330	10.0.0.1	10.0.0.21	DNS	110 Standard query response 0x0e2a NS <Root>
10	0.085154826	fe:fd:00:00:04:01	Broadcast	ARP	42 Who has 10.0.0.11? Tell 10.0.0.21
11	0.086109829	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	0.086357607	10.0.0.21	10.0.0.11	DNS	80 Standard query 0xc3b8 A alice.com OPT
13	0.091739477	10.0.0.11	10.0.0.21	DNS	131 Standard query response 0xc3b8 No such r
14	0.093330614	10.0.0.21	10.0.0.22	DNS	120 Standard query response 0x8d70 No such r
15	5.086818329	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
16	5.087004576	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
17	5.097108825	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
18	5.097317681	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
19	5.254616182	10.0.0.22	10.0.0.21	DNS	69 Standard query 0xadb5 A alice.com
20	5.255378752	10.0.0.21	10.0.0.11	DNS	80 Standard query 0x4b12 A alice.com OPT
21	5.255823850	10.0.0.11	10.0.0.21	DNS	131 Standard query response 0x4b12 No such r
22	5.256560942	10.0.0.21	10.0.0.22	DNS	120 Standard query response 0xadb5 No such r

https://drive.google.com/drive/folders/1ugbAayGfGTtjjVX0rLdREecVnr-qNXcN?usp=sharing

Alice is not a FQDN (should be alice.example.com) so nsce responds that alice doesnt exists.
In the second dig its the same but its not needed the hole route so its in alice and nsce cache.

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:

```
alice:~# dig alice.com ; sleep 5 ; dig alice.com ; sleep 10 ; dig alice.com
```

nsc: ~# nano /etc/bind/db.com -> 10; negative cache TTL

```
  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
                        )
com.                IN      NS      nsc.com.
nsc.com.            IN      A       10.0.0.11
bob.com.     30     IN      A       10.0.0.12

example.com.        IN      NS      nsce.example.com.
nsce.example.com.   IN      A       10.0.0.21

                        [ Read 14 lines ]
```

SimNet0:

1	0.000000000	10.0.0.22	10.0.0.21	DNS	69	Standard query 0xbfb5 A alice.com
2	0.051313485	fe:fd:00:00:04:01	Broadcast	ARP	42	Who has 10.0.0.1? Tell 10.0.0.21
3	0.051915931	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
4	0.052433324	10.0.0.21	10.0.0.1	DNS	80	Standard query 0xaff6 A alice.com OPT
5	0.052468093	10.0.0.21	10.0.0.1	DNS	70	Standard query 0x7478 NS <Root> OPT
6	0.061028493	10.0.0.1	10.0.0.21	DNS	114	Standard query response 0xaff6 A alice.c
7	0.064329714	10.0.0.21	10.0.0.11	DNS	80	Standard query 0x4fee A alice.com OPT
8	0.069390914	10.0.0.1	10.0.0.21	DNS	110	Standard query response 0x7478 NS <Root>
9	0.071222301	10.0.0.11	10.0.0.21	DNS	131	Standard query response 0x4fee No such r
10	0.072678694	10.0.0.21	10.0.0.22	DNS	120	Standard query response 0xbfb5 No such r
11	5.061758338	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	5.062127677	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
13	5.062147813	fe:fd:00:00:04:01	fe:fd:00:00:03:01	ARP	42	Who has 10.0.0.11? Tell 10.0.0.21
14	5.062159892	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
15	5.062172459	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
16	5.062865325	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
17	5.062882687	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
18	5.063111068	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
19	5.399572332	10.0.0.22	10.0.0.21	DNS	69	Standard query 0xfb7b A alice.com
20	5.399990910	10.0.0.21	10.0.0.22	DNS	120	Standard query response 0xfb7b No such r
21	10.161855480	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
22	10.162343994	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
23	15.643230610	10.0.0.22	10.0.0.21	DNS	69	Standard query 0x2d5a A alice.com
24	15.644067037	10.0.0.21	10.0.0.11	DNS	80	Standard query 0x7479 A alice.com OPT
25	15.644410907	10.0.0.11	10.0.0.21	DNS	131	Standard query response 0x7479 No such r
26	15.645068051	10.0.0.21	10.0.0.22	DNS	120	Standard query response 0x2d5a No such r

As first the route to find alice.com is the same as always. After 5 seconds, the 2nd dig doesnt go till root or asks for nsc cause its ttl is 10 and not 0s. In the last dig it goes to nsc again.

slides 08-09
https://drive.google.com/drive/folders/1ugbAayGfGTtjjVX0rLdREecVnr-qNXcN?usp=sharing

EXERCISE 3 - 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.

nsn:~# nano /etc/bind/db.net

```
  GNU nano 2.0.7            File: /etc/bind/db.net

$ORIGIN 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         IN      NS      nsne.example
```

nsne:~# nano /etc/bind/db.net.example

```
  GNU nano 2.0.7          File: /etc/bind/db.net.example

$ORIGIN example.net.
$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        IN      A       10.0.0.133
nsner.right     IN      NS      NSNER
nsner           IN      A       10.0.0.131
```

nsner:~# nano /etc/bind/db.net.example.right

```
  GNU nano 2.0.7        File: /etc/bind/db.net.example.right

$ORIGIN right.example.net.
$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
```

/etc/init.d/bind9 start -> Restart all servers

https://drive.google.com/drive/folders/1ugbAayGfGTtjjVX0rLdREecVnr-qNXcN?usp=sharing

2) Notice that the server nsn has a mistake in its initial configuration file, describe this mistake.
It has example.net added and righ.example.net which is wrong. It should be example.net. and right.example.net.

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:

```
alice:~# dig david.example.com
```

```
fd:00:00:08:01    Broadcast           ARP        42 Who has 10.0.0.21? Tell 10.0.0.22
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
0.0.22            10.0.0.21           DNS        77 Standard query 0x99d1 A david.example.com
fd:00:00:04:01    Broadcast           ARP        42 Who has 10.0.0.111? Tell 10.0.0.21
fd:00:00:05:01    fe:fd:00:00:04:01   ARP        42 10.0.0.111 is at fe:fd:00:00:05:01
0.0.21            10.0.0.111          DNS        88 Standard query 0x6289 A david.example.net OPT
0.0.111           10.0.0.21           DNS       123 Standard query response 0x6289 A david.example.net NS nsne.example
fd:00:00:04:01    Broadcast           ARP        42 Who has 10.0.0.121? Tell 10.0.0.21
fd:00:00:06:01    fe:fd:00:00:04:01   ARP        42 10.0.0.121 is at fe:fd:00:00:06:01
0.0.21            10.0.0.121          DNS        88 Standard query 0xe4a0 A david.example.net OPT
0.0.121           10.0.0.21           DNS       139 Standard query response 0xe4a0 A david.example.net A 10.0.0.122 NS
0.0.21            10.0.0.22           DNS       143 Standard query response 0x99d1 A david.example.com CNAME david.exa
fd:00:00:04:01    fe:fd:00:00:08:01   ARP        42 Who has 10.0.0.22? Tell 10.0.0.21
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
fd:00:00:06:01    fe:fd:00:00:04:01   ARP        42 Who has 10.0.0.21? Tell 10.0.0.121
fd:00:00:04:01    fe:fd:00:00:06:01   ARP        42 10.0.0.21 is at fe:fd:00:00:04:01
```

EXERCISE 4

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.

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.
joker:~# /etc/init.d/bind9 start-> start DNS server on joker
joker:~# cat /etc/bind/named.conf -> file “/etc/bind/db.root”

```
        // prime the server with knowledge of the root servers
zone "." {
        type hint;
        file "/etc/bind/db.root";
        };
```

joker:~# cat /etc/bind/db.root

```
joker:~# cat /etc/bind/db.root
.                       IN  NS    ROOT-SERVER.
ROOT-SERVER.            IN  A     10.0.0.1
```

joker:~# /etc/init.d/bind9 restart
alice:~# vi /etc/resolv.conf-> change nameserver for 10.0.0.201
carla:~# vi /etc/resolv.conf-> change nameserver for 10.0.0.201
carla:~# dig alice.example.com

1	0.000000000	fe:fd:00:00:09:01	Broadcast	ARP	42 Who has 10.0.0.201? Tell 10.0.0.132
2	0.001073114	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	0.001815549	10.0.0.132	10.0.0.201	DNS	77 Standard query 0x9077 A alice.example.c
4	0.064337985	fe:fd:00:00:02:01	Broadcast	ARP	42 Who has 10.0.0.1? Tell 10.0.0.201
5	0.064490454	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	0.064663341	10.0.0.201	10.0.0.1	DNS	88 Standard query 0x67c7 A alice.example.c
7	0.064689756	10.0.0.201	10.0.0.1	DNS	70 Standard query 0xce1c NS <Root> OPT
8	0.065083642	10.0.0.1	10.0.0.201	DNS	122 Standard query response 0x67c7 A alice.
9	0.065257978	10.0.0.1	10.0.0.201	DNS	110 Standard query response 0xce1c NS <Root
10	0.089757545	fe:fd:00:00:02:01	Broadcast	ARP	42 Who has 10.0.0.11? Tell 10.0.0.201
11	0.090123841	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	0.090627920	10.0.0.201	10.0.0.11	DNS	88 Standard query 0xc71c A alice.example.c
13	0.099207873	10.0.0.11	10.0.0.201	DNS	123 Standard query response 0xc71c A alice.
14	0.122095083	fe:fd:00:00:02:01	Broadcast	ARP	42 Who has 10.0.0.21? Tell 10.0.0.201
15	0.122365360	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	0.122713598	10.0.0.201	10.0.0.21	DNS	88 Standard query 0x7e0c A alice.example.c
17	0.123482997	10.0.0.21	10.0.0.201	DNS	139 Standard query response 0x7e0c A alice.
18	0.125364072	10.0.0.201	10.0.0.132	DNS	112 Standard query response 0x9077 A alice.
19	5.131644429	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	5.131963065	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
21	5.142114548	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
22	5.142291014	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

alice:~# dig carla.right.example.net

1	0.000000000	fe:fd:00:00:08:01	Broadcast	ARP	42 Who has 10.0.0.201? Tell 10.0.0.22
2	0.000247587	fe:fd:00:00:02:01	fe:fd:00:00:08:01	ARP	42 10.0.0.201 is at fe:fd:00:00:02:01
3	0.000392574	10.0.0.22	10.0.0.201	DNS	83 Standard query 0x9de9 A carla.right.exam
4	0.007003611	10.0.0.201	10.0.0.1	DNS	94 Standard query 0x27e1 A carla.right.exam
5	0.007310840	10.0.0.1	10.0.0.201	DNS	128 Standard query response 0x27e1 A carla.r
6	0.008002441	10.0.0.201	10.0.0.1	DNS	70 Standard query 0x722f NS <Root> OPT
7	0.008254901	10.0.0.1	10.0.0.201	DNS	110 Standard query response 0x722f NS <Root>
8	0.030090338	fe:fd:00:00:02:01	Broadcast	ARP	42 Who has 10.0.0.111? Tell 10.0.0.201
9	0.030262712	fe:fd:00:00:05:01	fe:fd:00:00:02:01	ARP	42 10.0.0.111 is at fe:fd:00:00:05:01
10	0.030470632	10.0.0.201	10.0.0.111	DNS	94 Standard query 0x56d7 A carla.right.exam
11	0.030843647	10.0.0.111	10.0.0.201	DNS	129 Standard query response 0x56d7 A carla.r
12	0.052316692	fe:fd:00:00:02:01	Broadcast	ARP	42 Who has 10.0.0.121? Tell 10.0.0.201
13	0.052584654	fe:fd:00:00:06:01	fe:fd:00:00:02:01	ARP	42 10.0.0.121 is at fe:fd:00:00:06:01
14	0.052738046	10.0.0.201	10.0.0.121	DNS	94 Standard query 0xb6a2 A carla.right.exam
15	0.053736841	10.0.0.121	10.0.0.201	DNS	146 Standard query response 0xb6a2 No such r
16	0.054638303	10.0.0.201	10.0.0.22	DNS	135 Standard query response 0x9de9 No such r
17	5.007395040	fe:fd:00:00:02:01	fe:fd:00:00:01:01	ARP	42 Who has 10.0.0.1? Tell 10.0.0.201
18	5.007666294	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
19	5.049921480	fe:fd:00:00:06:01	fe:fd:00:00:02:01	ARP	42 Who has 10.0.0.201? Tell 10.0.0.121
20	5.050626691	fe:fd:00:00:02:01	fe:fd:00:00:06:01	ARP	42 10.0.0.201 is at fe:fd:00:00:02:01
21	5.070469004	fe:fd:00:00:02:01	fe:fd:00:00:08:01	ARP	42 Who has 10.0.0.22? Tell 10.0.0.201
22	5.070747748	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

https://drive.google.com/drive/folders/1ugbAayGfGTtjjVX0rLdREecVnr-qNXcN?usp=sharing