

Εργαστήριο Δικτύων Υπολογιστών

Εργαστηριακή Άσκηση 11

Δημήτριος Κόγιος

03119220

Όνομα PC: lekog-HP-Laptop-15s-fq1xxx

Άσκηση 1:

1.1) echo 'ifconfig_emo_ipv6="inet6 accept_rtadv" ' >>
/etc/rc.conf

1.2) service netif stop
service netif start

1.3) ifconfig emo
Έχει αποδοθεί η link-local : fe80::a00:27ff:fe29:1df8%emo / 64

1.4) ifconfig emo
Έχει αποδοθεί η link-local : fe80::a00:27ff:fe46:b04f%emo / 64

1.5) Είναι link-local διευθύνσεις οι οποίες παράγονται από τη διεύθυνση MAC βάζοντας το fe80:0000 στην αρχή , παρεμβάλλοντας το ff:fe στο μέσο της και αντιστρέφοντας το 7 bit του πρώτου byte δηλαδή στο PC1 που έχει MAC 08:00:27:99:1d:f8 προκύπτει η διεύθυνση fe80::0a00:27ff:fe99:1df8.

1.6) netstat -r -6
Υπάρχουν 9 εγγραφές.

1.7) Μόνο μία , η link-local fe80::%emo/64 .

1.8) Περιέχει τις:

fe80::%emo/64 emo

fe80::%loo/64 loo

fe80::1%loo loo

1.9) ping6 ::1

Απαντάει το ίδιο το PC1 αφού η ::1 είναι η loopback διεύθυνση.

1.10) ping6 fe80::a00:27ff:fe99:1df8%emo

Πρέπει στο τέλος να προσθέσουμε τον δείκτη ζώνης.

1.11) ping6 fe80::a00:27ff:fe46:b04f%emo

Πρέπει στο τέλος να προσθέσουμε τον δείκτη ζώνης.

1.12) Γνωρίζουμε ότι η διεύθυνση ffo1::1%emo είναι όλοι οι κόμβοι στη διεπαφή emo.

Άρα με ping6 ffo1::1%emo απαντά το ίδιο το PC1.

1.13) Απαντά και το PC2, αυτό γίνεται γιατί η διεύθυνση ffo2::1%emo παριστάνει όλους τους κόμβους στη τοπική ζεύξη που είναι συνδεδεμένη η διεπαφή emo.

1.14) ifconfig emo inet6 fd00:1::2/64

1.15) ifconfig emo inet6 fd00:1::3/64

1.16) Γνωρίζουμε ότι οι διευθύνσεις με πρόθεμα της μορφής fd.../48 δεν δρομολογούνται στο δημόσιο διαδίκτυο και έχουν χρήση ανάλογη των 10.0.0.0/8 , 172.16.0.0/12 και 192.168.0.0/16 στο IPv4.

1.17) Έχουν 2.

1.18) Πλέον έχουμε 11 εγγραφές αφού προστέθηκαν οι εγγραφές για τα fd00:1::/64 και η fd00:1::{2,3}.

1.19) Στο αρχείο /etc/hosts πρέπει να προσθέσουμε:

Στο PC1 : fd00:1::3 PC2

Στο PC2 : fd00:1::2 PC1

1.20) Ναι με ping6 PC2 , ping6 PC1 από το PC1 και PC2 αντίστοιχα.

1.21) arp -a

Δεν υπάρχει καμία εγγραφή

1.22) man ndp => coltrol/diagnose IPv6 neighbor discovery protocol.

Στο IPv6 η neighbor-discovery αντικαθιστά το πρωτόκολλο ARP.

1.23) ndp -a

1.24) Βλέπουμε 4 εγγραφές , οι πρώτες δύο είναι για τις διευθύνσεις του PC1, οι επόμενες δύο για τις διευθύνσεις του PC2.

```
root@R0:~ # ndp -a
Neighbor                               Linklayer Address  Netif  Expire   S  Flags
fd00:1::2                             08:00:27:99:1d:f8  em0    permanent R
fe80::a00:27ff:fe99:1df8%em0          08:00:27:99:1d:f8  em0    permanent R
PC2                                    08:00:27:46:b0:4f  em0    23h45m45s S
fe80::a00:27ff:fe46:b04f%em0          08:00:27:46:b0:4f  em0    23h48m23s S
```

Το state των δύο πρώτων είναι R δηλαδή Reachable ενώ των επόμενων δύο είναι S δηλαδή Stale.

1.25) ndp -p

Η λίστα προθεμάτων ζεύξης (on-link) χρησιμεύει κατά την αποστολή πακέτων για να προσδιορισθεί το κατά πόσο ο προορισμός είναι επί της ζεύξης, δηλαδή, τοπικά προσβάσιμος σε αντίθεση με προθέματα εκτός ζεύξης (off-link) όπου απαιτείται η διαμεσολάβηση δρομολογητή.

Υπάρχουν εγγραφές για τα προθέματα fd00:1::/64 , fe80::%em0/64 και fe80::%lo0/64 που έχουν expire = Never.

```
root@R0:~ # ndp -p
fd00:1::/64 if=em0
flags=L0 vlttime=infinity, pltime=infinity, expire=Never, ref=1
No advertising router
fe80::%em0/64 if=em0
flags=LA0 vlttime=infinity, pltime=infinity, expire=Never, ref=0
No advertising router
fe80::%lo0/64 if=lo0
flags=LA0 vlttime=infinity, pltime=infinity, expire=Never, ref=0
No advertising router
root@R0:~ #
```

1.26) Τα flags των τριών εγγραφών είναι LO, LAO, LAO συνεπώς μπορεί να χρησιμοποιηθούν τα δύο τελευταία προθέματα από τον μηχανισμό αυτόματης απόδοσης διευθύνσεων (SLAAC) καθώς έχουν το flag “A” (autonomous)

1.27) ndp -c

1.28) tcpdump -vv -n

1.29) 6 πακέτα.

1.30) Μεταφέρουν μηνύματα του πρωτοκόλλου ICMPv6 με τιμή next-header 58.

1.31) PC1 -> ff02::1:ff00:3 : neighbor solicitation

PC2 -> PC1 : neighbor advertisement

PC1 -> PC2 : echo request

PC2 -> PC1 : echo reply

PC2 -> PC1 : neighbor solicitation

PC1 -> PC2 : neighbor advertisement

1.32) Η διεύθυνση ff02::111:ff00:3 είναι η multicast διεύθυνση solicited node η οποία χρησιμοποιείται όταν δεν είναι γνωστή εκ των προτέρων η διεύθυνση αποστολής στη φάση του Neighbor Discovery. Προκύπτει από τα τελευταία 24 bit της διεύθυνσης unicast ή anycast προσθέτοντας σε αυτά το πρόθεμα ff02:0:0:0:0:1:ff00:0/104 . Άρα η solicited node του PC2 που έχει fd00::3/64 είναι η ff02:0:0:0:0:1:ff02:3 aka ff02::1:ff02:3 .

1.33) Είναι fd00:1::2 που είναι η διεύθυνση που ορίσαμε στο PC1.

1.34) Εμφανίζεται ως R δηλαδή Reachable με expire τα 30 sec (μέγιστη τιμή , άμα αργήσουμε θα μειωθεί). Όταν λήξουν αυτά τα 30 sec γίνεται S δηλαδή Stale.

1.35) Παρατηρούμε τις καταστάσεις R και S.

1.36) Η διάρκεια της R είναι 30sec και όταν λήξουν αυτά γίνεται S. Επειδή το ping6 συνεχίζεται πολύ γρήγορα ξαναγίνεται R.

1.37) Είναι 24h.

1.38) Βλέπω μόνο S.

1.39) Βλέπουμε NS και NA κάθε περίπου 30 sec δηλαδή όταν από R γίνεται S η εγγραφή.

Άσκηση 2:

2.1) echo 'ipv6_gateway_enable="YES"' >> /etc/rc.conf
service routing restart

2.2) ifconfig emo inet6 fd00:1::3/64 delete
ifconfig emo inet6 fd00:2::2/64

2.3) vtysh
configure terminal
interface emo
ip address fd00:1::1/64

2.4) interface em1
ip address fd00:3::1/126

2.5) vtysh
configure terminal
interface em1
ip address fd00:2::1/64

2.6) interface emo
ip address fd00:3::2/126

2.7) route -6 add default fd00:1::1

2.8) route -6 add default fd00:2::1

2.9) tcpdump -i emo

2.10) ndp -c

ping6 -c 1 fd00:2::2/64

Όχι το ping αποτυγχάνει αφού ο R1 δεν έχει εγγραφή στον πίνακα δρομολόγησής του για το PC2.

2.11) Πρώτα το PC1 στέλνει NS για να μάθει την MAC διεύθυνση του R1 , ο R1 του απαντά με NA. Μετά το PC1 προωθεί το ICMP6 echo request που τελικό προορισμό έχει το PC2 στο R1. Ο R1 του στέλνει destination unreachable , NS για να μάθει τη MAC του PC1 και ο PC1 του απαντά με NA.

```
root@R0:~ # tcpdump -i em0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on em0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:55:08.900507 IP6 fd00:1::2 > ff02::1:ff00:1: ICMP6, neighbor solicitation, who has fd00:1::1, length 32
16:55:08.900560 IP6 fd00:1::1 > fd00:1::2: ICMP6, neighbor advertisement, tgt is fd00:1::1, length 32
16:55:08.901000 IP6 fd00:1::2 > fd00:2::2: ICMP6, echo request, seq 0, length 16
16:55:08.901014 IP6 fd00:1::1 > fd00:1::2: ICMP6, destination unreachable, unreachable route fd00:2::2, length 64
16:55:14.688813 IP6 fd00:1::1 > fd00:1::2: ICMP6, neighbor solicitation, who has fd00:1::2, length 32
16:55:14.689260 IP6 fd00:1::2 > fd00:1::1: ICMP6, neighbor advertisement, tgt is fd00:1::2, length 24
```

2.12) ipv6 route fd00:2::0/64 fd00:3::2

2.13) Όχι γιατί το R2 δεν έχει εγγραφή για το PC1.

2.14) ipv6 route fd00:1::0/64 fd00:3::1

2.15) Ναι πλέον λαμβάνουμε απάντηση.

2.16) interface emo

no ipv6 nd suppress-ra

2.17) interface em1

ipv6 nd prefix fd00:1::/64

2.18) interface em1

no ipv6 nd suppress-ra

2.19) interface em0

ipv6 nd prefix fd00:2::/64

2.20) route -6 delete default fd00:1::1

2.21) tcpdump -i em0 -e -n '(icmp6)'

2.22) service netif restart

2.23) RS , RA , NS

```
root@R0:~ # tcpdump -i em0 -e -n icmp6
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on em0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:15:44.950604 08:00:27:99:1d:f8 > 33:33:00:00:00:02, ethertype IPv6 (0x86dd),
length 70: fe80::a00:27ff:fe99:1df8 > ff02::2: ICMP6, router solicitation, length 16
17:15:44.951282 08:00:27:9c:b0:e0 > 33:33:00:00:00:01, ethertype IPv6 (0x86dd),
length 110: fe80::a00:27ff:fe9c:b0e0 > ff02::1: ICMP6, router advertisement, length 56
17:15:45.805956 08:00:27:99:1d:f8 > 33:33:ff:99:1d:f8, ethertype IPv6 (0x86dd),
length 86: :: > ff02::1:ff99:1df8: ICMP6, neighbor solicitation, who has fd00:1:
:a00:27ff:fe99:1df8, length 32
```

2.24) Παράγει NS ώστε να δει αν κάποιος άλλος χρησιμοποιεί τη διεύθυνσή του, τότε θα λάβει NA.

2.25) Χρησιμοποιεί την ακαθόριστη διεύθυνση ::0 ως διεύθυνση πηγής ώστε να ελέγξει αν κάποιος άλλος χρησιμοποιεί τη διεύθυνσή του.

2.26) Χρησιμοποιεί την link-local διεύθυνσή του.

2.27) Η διεύθυνση προορισμού του NS που στέλνει ο PC1 είναι η solicited node διεύθυνση που προκύπτει από τη διεύθυνσή του

αφού αυτήν θέλει να ελέγξει αν χρησιμοποιείται από κάποιον άλλον. Για το RS διεύθυνση προορισμού είναι η ff02::2 δηλαδή όλοι οι δρομολογητές αυτής της ζεύξης και για το RA διεύθυνση προορισμού είναι η ff02::1 δηλαδή όλοι οι κόμβοι στην τοπική ζεύξη.

2.28) Για το NS : 33:33:ff:99:1d:f8 , καθώς για την αποστολή πακέτων multicast IPv6 πάνω από Ethernet δημιουργείται μια MAC με το πρόθεμα 33:33 να ακολουθείται από τα τελευταία 32 bit της IPv6 διεύθυνσης προορισμού.

Για το RS : 33:33:00:00:00:02 αφού IPv6 διεύθυνση προορισμού είναι η ff02::2.

Για το RA : 33:33:00:00:00:01 αφού IPv6 διεύθυνση προορισμού είναι η ff02::1.

2.29) ndp -p

```
root@R0:~ # ndp -p
fd00:1::/64 if=em0
flags=LAO vlttime=2592000, pltime=604800, expire=29d23h58m14s, ref=1
  advertised by
    fe80::a00:27ff:fe9c:b0e0%em0 (reachable)
fe80::%em0/64 if=em0
flags=LAO vlttime=infinity, pltime=infinity, expire=Never, ref=0
  No advertising router
fe80::%lo0/64 if=lo0
flags=LAO vlttime=infinity, pltime=infinity, expire=Never, ref=0
  No advertising router
root@R0:~ #
```

Πλέον ο R1 μας έχει διαφημίσει το πρόθεμα fd00:1::/64 από το προηγούμενο RA.

2.30) ifconfig

emo => fe80::a00:27ff:fe99:1df8%emo / 64

fd00:1::a00:27ff:fe99:1df8 /64

lo0 => ::1 / 128

fe80::1%lo0

2.31) Ναι υπάρχει default και είναι η διεύθυνση πηγής του RA που είναι η link-local διεύθυνση του R1:

```
root@R0:~ # netstat -r6
Routing tables

Internet6:
Destination      Gateway           Flags      Netif  Expire
:::/96            localhost         UGRS       lo0
default           fe80::a00:27ff:fe9 UG         em0
localhost         link#2            UH         lo0
::ffff:0.0.0.0/96 localhost         UGRS       lo0
fd00:1::/64       link#1            U          em0
fd00:1::a00:27ff:f link#1            UHS        lo0
fe80::/10         localhost         UGRS       lo0
fe80::%em0/64     link#1            U          em0
fe80::a00:27ff:fe9 link#1            UHS        lo0
fe80::%lo0/64     link#2            U          lo0
fe80::1%lo0       link#2            UHS        lo0
ff02::/16         localhost         UGRS       lo0
root@R0:~ #
```

2.32) Από το PC2 μπορούμε να χρησιμοποιήσουμε τη διεύθυνση fd00:1::a00:27ff:fe99:1df8 ενώ από τον R1 μπορούμε να χρησιμοποιήσουμε και την fd00:1::a00:27ff:fe99:1df8 αλλά και την fe80::a00:27ff:fe99:1df8%emo.

Άσκηση 3:

3.1) R1 : no ipv6 route fd00:2::/64 fd00:3::2

R2 : no ipv6 route fd00:1::/64 fd00:3::1

3.2) router ripng

network em0

network em1

3.3) do show ipv6 route ripng

Βλέπουμε μία εγγραφή , για το LAN2 μέσω του R2.

3.4) Ως διεύθυνση του επόμενου κόμβου είναι η link-local του R2 στο WAN.

```
R1>* fd00:2::/64 [120/2] via fe80::a00:27ff:fee6:fb2e, em1, weight 1, 00:00:44  
R1(config)#
```

3.5) Ναι.

3.6) tcpdump -i em1 -vv -n

3.7) Βλέπουμε πακέτα ripng-resp 2 με διεύθυνση προορισμού

ff02::9 που είναι η multicast διεύθυνση που ακούνε οι RIP

routers όπως πληροφορούμαστε από

<https://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xhtml>

3.8) Το hop limit είναι 255 ώστε να βεβαιωθούμε ότι ο

αποστολέας είναι γείτονάς μας γιατί πακέτα του RIP με hop limit μικρότερη του 255 αγνοούνται.

3.9) Χρησιμοποιεί UDP και port 521. Το RIP χρησιμοποιούσε UDP και port 520.

3.10) no router ripng

3.11) do write

3.12) service frr restart

3.13) R1 : router ospf6
router-id 1.1.1.1
ospf6 router-id 1.1.1.1
R2 : router ospf6
router-id 2.2.2.2
ospf6 router-id 2.2.2.2

3.14) interface em0 area 0.0.0.0
interface em1 area 0.0.0.0

3.15) interface em0 area 0.0.0.0
interface em1 area 0.0.0.0

3.16) do show ip route ospf6

Υπάρχουν 3 εγγραφές μόνο μια είναι επιλεγμένη. Το κόστος προκύπτει από τη διαίρεση $Cost = reference\ bandwidth / interface\ bandwidth$.

```
R1>* fd00:1::/64 [110/200] via fe80::a00:27ff:fe59:221a, em0, weight 1, 00:00:24
0   fd00:2::/64 [110/100] is directly connected, em1, weight 1, 00:00:36
0   fd00:3::/126 [110/100] is directly connected, em0, weight 1, 00:00:29
R2(config-ospf6)#
```

3.17) Είναι η link-local διεύθυνση του R1 στο WAN.

3.18) tcpdump -i em0 -vv -n

3.19) Παρατηρώ μηνύματα Hello με διεύθυνση προορισμού ff02::5.

3.20) Έχει την τιμή 1.

3.21) next-header OSPF (89) δηλαδή ίδιο με του OSPFv2.

3.22) Ναι μπορούμε.

3.23) no router ospf6

3.24) service frr restart

3.25) router-id 1.1.1.1
router bgp 65010

3.26) no bgp ebgp-requires-policy

3.27) no bgp default ipv4-unicast

3.28) neighbor fd00:3::2 remote-as 65020

3.29) address-family ipv6

3.30) network fd00:1::/64

3.31) neighbor fd00:3::2 activate

3.32) router-id 2.2.2.2
router bgp 65020
no bgp ebgp-requires-policy

```
no bgp default ipv4-unicast
neighbor fd00:3::1 remote-as 65010
address-family ipv6
network fd00:2::/64
neighbor fd00:3::1 activate
```

3.33) do show ip route bgp

Βλέπουμε 1 εγγραφή , αυτή για το LAN2.

3.34) Η διεύθυνση του επόμενου κόμβου για το LAN2 είναι η link-local διεύθυνση του R2 στο WAN.

```
R1>* fd00:2::/64 [20/0] via fe80::a00:27ff:fee6:fb2e, em1, weight 1, 00:04:53
R1(config)#
```

3.35) tcpdump -i em1 -vv -n '(not icmp6)'

3.36) Βλέπουμε μηνύματα keepalive που χρησιμοποιούν το TCP στη θύρα 179 δηλαδή ίδια με το IPv4.

3.37) Έχει τιμή 1.

3.38) Ναι μπορούμε.

3.39) reboot

```
vttysh
configure terminal
interface em0
ipv6 address fd00:1::2/64
```

3.40) router-id 1.1.0.0

```
router bgp 65010
```

3.41) no bgp default ipv4-unicast

3.42) neighbor fd00:1::1 remote-as 65010

3.43) address-family ipv6
neighbor fd00:1::1 activate

3.44) neighbor fd00:1::2 remote-as 65010

3.45) address-family ipv6
neighbor fd00:1::2 next-hop-self

3.46) do show ip bgp neighbors

3.47) do show ipv6 route bgp

Βλέπουμε δύο εγγραφές και είναι επιλεγμένη μόνο μία , αυτή για το LAN1.

3.48) Γιατί υπάρχει η directly connected που έχει μικρότερη διαχειριστική απόσταση.

3.49) Είναι η link-local του R1 στο LAN1.

3.50) Ναι μπορούμε.

Άσκηση 4:

4.1) interface em0

ip address 192.168.1.1/24

4.2) interface em1

ip address 192.168.2.1/24

4.3) interface em0

ip address 192.168.1.2/24

ip route 0.0.0.0/0 192.168.1.1

4.4) interface em0

ip address 192.168.2.2/24

ip route 0.0.0.0/0 192.168.2.1

4.5) vi /etc/rc.conf

4.6) service ipfw start

4.7) ipfw list

Έχει 12 κανόνες.

4.8) Ναι μπορούμε.

4.9) ipfw nat64clat nat64 create clat_prefix fd00:3:1::/96

plat_prefix 64:ff9b::/96 allow_private log

4.10) ipfw add 2000 nat64clat nat64 ip4 from any to not me recv
em0

4.11) ipfw add 3000 nat64clat nat64 ip6 from 64:ff9b::/96 to
fd00:3:1::/96 recv em1

4.12) ipv6 route 64:ff9b::/96 fd00:3::2

4.13) done.

4.14) ipfw nat64lsn nat64 create prefix4 2.2.2.0/24 prefix6
64:ff9b::/96 allow_private log

4.15) ipfw add 2000 nat64lsn nat64 ip6 from fd00:3:1::/96 to
64:ff9b::/96 recv em0

4.16) ipfw add 3000 nat64lsn nat64 ip4 from any to 2.2.2.0/24
recv em1

4.17) ipv6 route fd00:3:1::/96 fd00:3::1

4.18) ip route 0.0.0.0/0 192.168.2.2

4.19) Με ping 192.168.1.1 και ping 192.168.2.2 τα ping είναι
επιτυχή.

4.20) ifconfig ipfwlogo create
tcpdump -i ipfwlogo

4.21) ifconfig ipfwlogo create
tcpdump -i ipfwlogo

4.22) Στην καταγραφή του R1 :

```
03:39:00.371829 IP 192.168.1.2 > 192.168.2.2: ICMP echo request, id 61958, seq 0, length 64
03:39:00.371842 IP6 fd00:3:1::c0a8:102 > 64:ff9b::c0a8:202: ICMP6, echo request, id 0, seq 0, length 64
03:39:00.373191 IP6 64:ff9b::c0a8:202 > fd00:3:1::c0a8:102: ICMP6, echo reply, id 0, seq 0, length 64
03:39:00.373195 IP 192.168.2.2 > 192.168.1.2: ICMP echo reply, id 61958, seq 0, length 64
```

Στην καταγραφή του R2 :

```
03:39:16.990072 IP6 fd00:3:1::c0a8:102 > 64:ff9b::c0a8:202: ICMP6, echo request, id 0, seq 0, length 64
03:39:16.990081 IP 2.2.2.74 > 192.168.2.2: ICMP echo request, id 1024, seq 0, length 64
03:39:16.990542 IP 192.168.2.2 > 2.2.2.74: ICMP echo reply, id 1024, seq 0, length 64
03:39:16.990547 IP6 64:ff9b::c0a8:202 > fd00:3:1::c0a8:102: ICMP6, echo reply, id 0, seq 0, length 64
```

4.23) interface em0

ip address 172.17.17.2/24

ip address 10.0.0.2/24

4.24) Με ping 172.17.17.2 και ping 10.0.0.2 τα ping είναι επιτυχή.

4.25) ipfw nat64lsn nat64 show states

Κενό.

4.26) Όταν κάνουμε ping από το PC1 προς το PC2 βλέπουμε ότι δημιουργείται state στον R2 που διαρκεί περίπου ένα λεπτό.

Άσκηση 5:

5.1) dhclient emo

5.2) pkg install miredo

5.3) sysrc miredo_enable="YES"

5.4) vi /usr/local/etc/miredo/miredo.conf
service miredo start

5.5) Βλέπουμε τη διεπαφή teredo με διεύθυνση η οποία έχει δύο IPv6 διευθύνσεις : fe80::ffff:ffff:ffff%teredo / 64 και
2001:0:d911:cod9:10f1:4068:6c99:1118/128

5.6) tcpdump -i emo -n

5.7) Είναι 217.17.192.217

5.8) Χρησιμοποιείται UDP και θύρα 3544.

5.9) Παρατηρώ μηνύματα ICMPv6.

5.10) Ναι και τα τρία ping επιτυγχάνουν.

5.11) ping6 www.ntua.gr

5.12) Παρατηρώ μηνύματα Direct IPv6 Connectivity Test.

5.13) Όχι.

5.14) UDP και θύρα 3545.

5.15) `tcpdump -i teredo -n`

5.16) Παρατηρούμε πακέτα ICMP6 και πρωτόκολλο ανωτέρου στρώματος IP6.

5.17) Όχι δεν παίρνουμε απάντηση στο `ping6` μας.

5.18) Ναι ICMP6 echo requests.

5.19) Δεν παράγονται.

5.20) `ping6 www.quad9.net` : επιλέγεται ο 216.218.142.110
`ping6 www.f5.com` : επιλέγεται ο 216.66.86.178