

**Εργαστηριο δικτυων υπολογιστων**

**Εργαστηριακη ασκηση 11: Το πρωτΟκολλο IPv6**



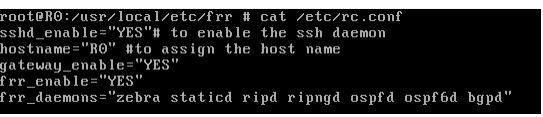


31 Μαιου, 2023

ΘΟΔΩΡΗΣ ΑΡΑΠΗΣ – EL18028

|  |  |
| --- | --- |
| **Ονοματεπώνυμο:** Θοδωρής Αράπης | **Ομάδα:** 3 |
| **Όνομα PC/ΛΣ:** DESKTOP-JGHL94V/ WINDOWS 10 | **Ημερομηνία:** 31/5/2023 |

**Προετοιμασία στο σπίτι**

Δημιουργούμε το μηχάνημα:

**Άσκηση 1: Εισαγωγή στο IPv6**

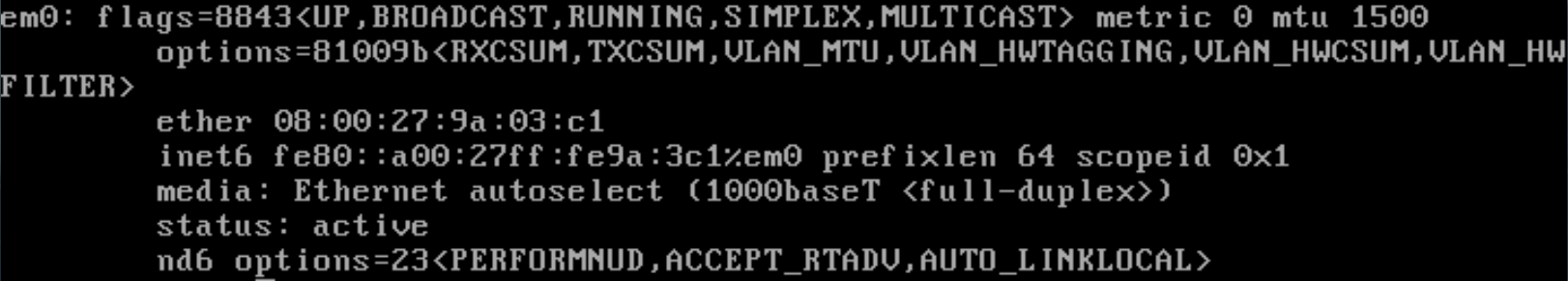
***1.1***

Εκτελούμε την παρακάτω εντολή στα PC.

***1.2***

Εκτελούμε “**service netif restart**”.

***1.3***

Έχει αποδοθεί η παρακάτω:

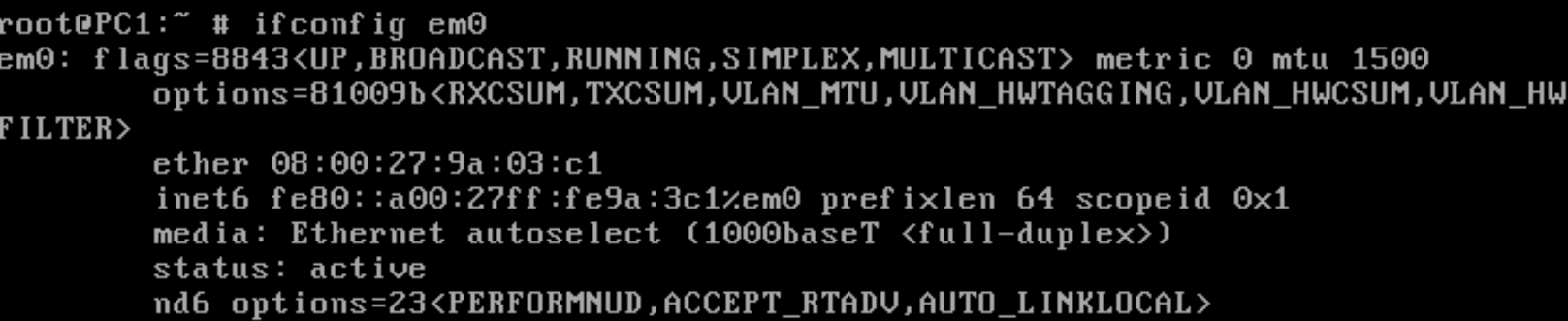
***1.4***

Έχει αποδοθεί η παρακάτω:

***1.5***

Οι παραπάνω διευθύνσεις είναι link-local και έχουν παραχθεί έχοντας ως πρώτα 64 bit το fe80::/64 και ως τελευταία 64 την EUI-64 MAC address, ο καθένας τη δική του. Για παράδειγμα, όσον αφορά το PC1 παράχθηκε ως εξής:

* **Πρώτα 64 bits**: fe80::/64 (**fe80:0000:0000:0000: …**)
* **Τελευταία 64** **bits**: Από την MAC address 08:00:27:9a:03:c1, αντιστρέφουμε το 7ο bit του πρώτου byte (0 816 = 0000 10002), οπότε και γίνεται 0a:00:27:9a:03:c1 και στη συνέχεια παρεμβάλουμε τα ff:fe στη μέση της MAC, οπότε γίνεται 0a:00:27:ff:fe:9a:03:c1 . **(… :0a00:27ff:fe9a:03c1**)

Συνενώνοντας τα παραπάνω παίρνουμε fe80:0000:0000:0000:0a00:27ff:fe9a:03c1, τα οποία απλοποιούνται στο fe80::a00:27ff:fe9a:3c1. Βλέπουμε παρακάτω πληροφορίες για την em0 του PC1.

***1.6***

Εκτελούμε στο PC1 “**netstat –rn6**” και βλέπουμε τις παρακάτω 9 εγγραφές.

***1.7***

Μία μόνο.

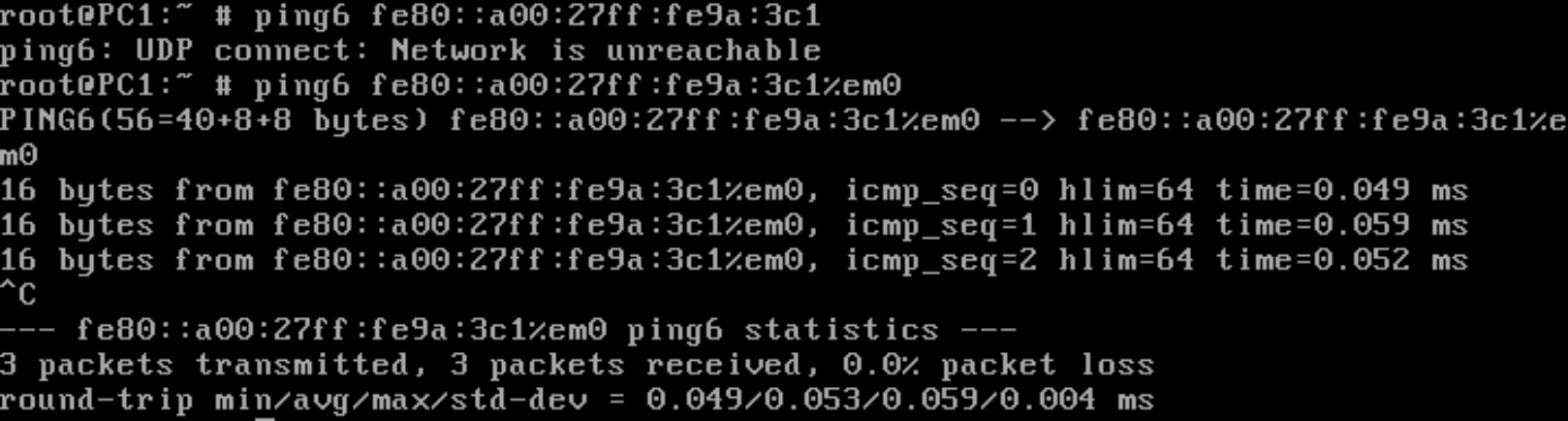
***1.8***

Τις “fe80::%em0/64” και “fe80::%lo0/64”.

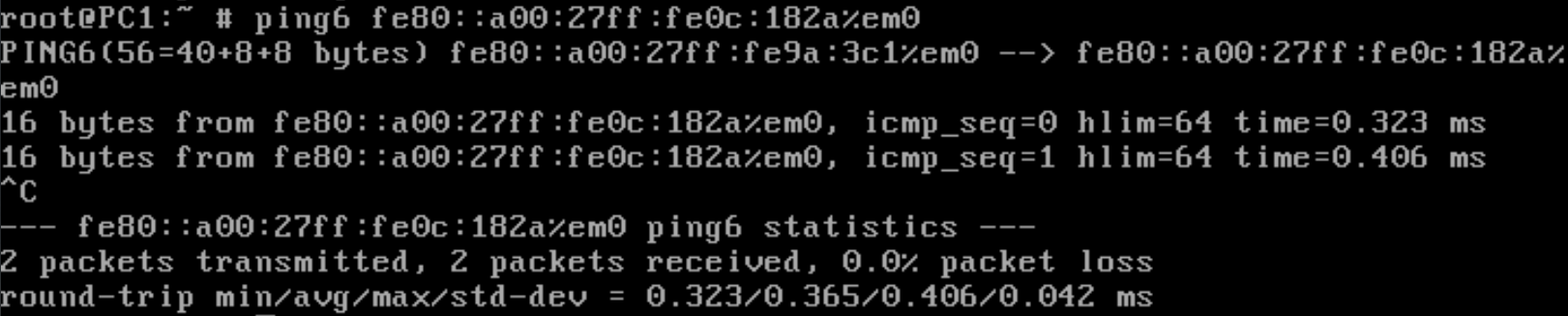
***1.9***

Το ίδιο το PC1.

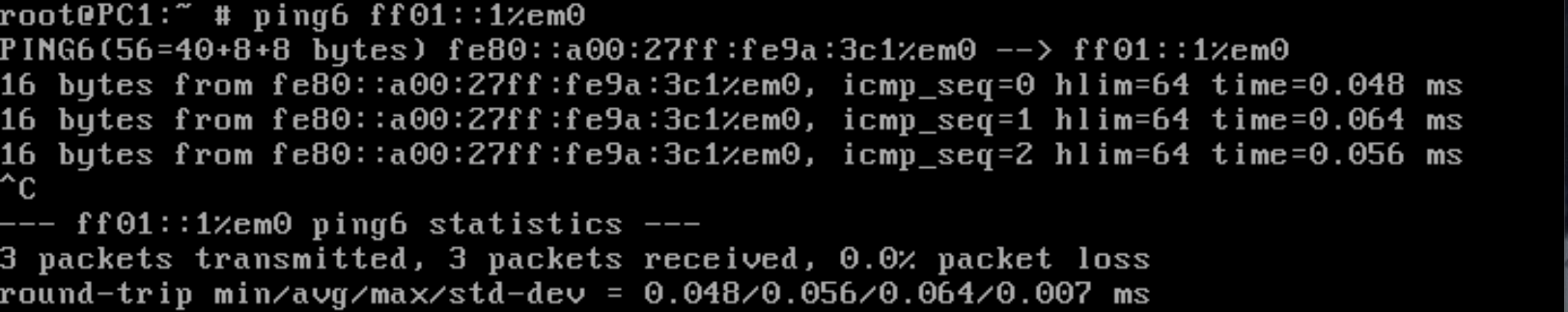
***1.10***

Πρέπει να προστεθεί το %em0.

***1.11***

Όμοια με πριν απαιτείται το %em0 στο τέλος.

***1.12***

Κάνοντας το ping6 από το PC1 στην διεύθυνση που αφορά όλους τους κόμβους στη τοπική διεπαφή, απαντάει προφανώς μόνο το PC1.

***1.13***

Παρατηρούμε ότι απαντάνε και τα 2 PC, αφού η διεύθυνση ff02::1 αφορά όλους τους κόμβους στην τοπική ζεύξη.



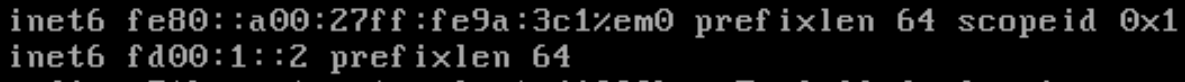
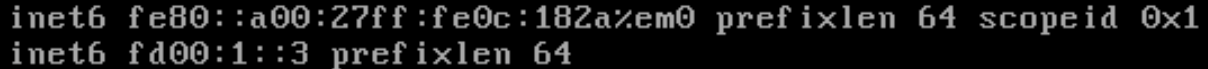
***1.14***

***1.15***

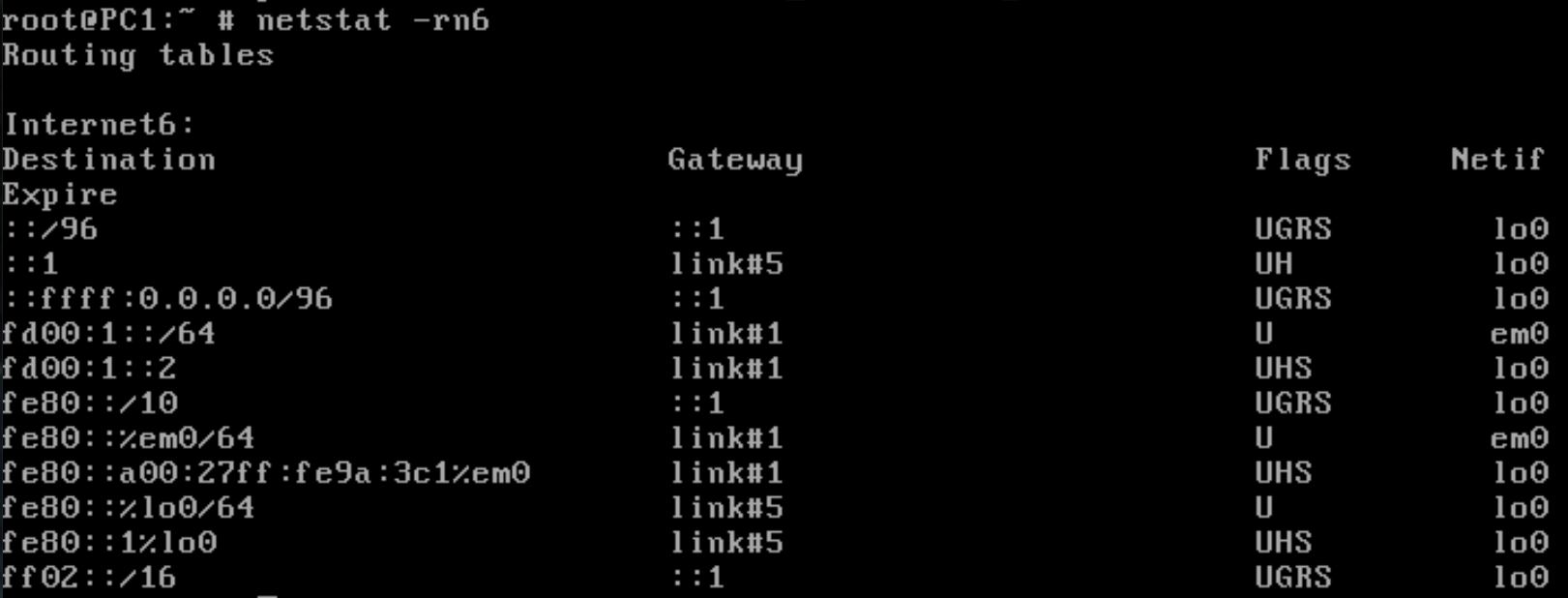
***1.16***

Είναι διευθύνσεις μη δρομολογούμενες στο δημόσιο διαδίκτυο, αντίστοιχες με τις ιδιωτικές διευθύνσεις στο IPv4, 10.0.0.0/8, 172.16.0.0/12 και 192.168.0.0/16.

***1.17***

Υπάρχουν πλέον από 2 σε κάθε PC.

***1.18***

Βλέπουμε πως υπάρχουν πλέον 11 εγγραφές. Προστέθηκαν 2 νέες.

***1.19***

Θα τροποποιήσουμε το αρχείο /etc/hosts σε κάθε PC προσθέτοντας τις παρακάτω γραμμές στα PC1 και PC2 αντίστοιχα.

***1.20***

Ναι.

***1.21***

Καμία.

***1.22***

Εκτελούμε “**man ndp**”.

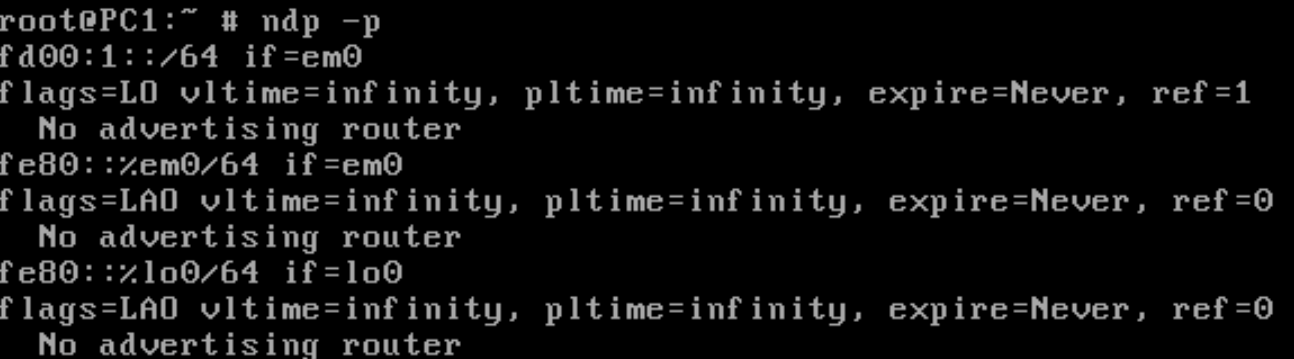
***1.23***

Εκτελούμε “**ndp -a**”.

***1.24***

3, σε όσες κάναμε ping6 προηγουμένως. Η πρώτη (PC2) είναι stale ενώ οι άλλες δύο Reachable.

***1.25***

Εκτελούμε “**ndp -p**”. Εμφανίζονται 2 εγγραφές για το πρόθεμα fe80::/64 (για τις διεπαφές em0 και lo0) και μία για το πρόθεμα fd00:1::/64. Ακόμη, βλέπουμε άπειρη διάρκεια ζωής (expire=Never) σε όλες τις εγγραφές.

***1.26***

Από τα προθέματα αυτά, μπορούν να χρησιμοποιηθούν από τον μηχανισμό αυτόματης απόδοσης διευθύνσεων αυτές που έχουν το Flag A, επομένως τα προθέματα fe80::%em0/64 και fe80::%lo0/64.

***1.27***

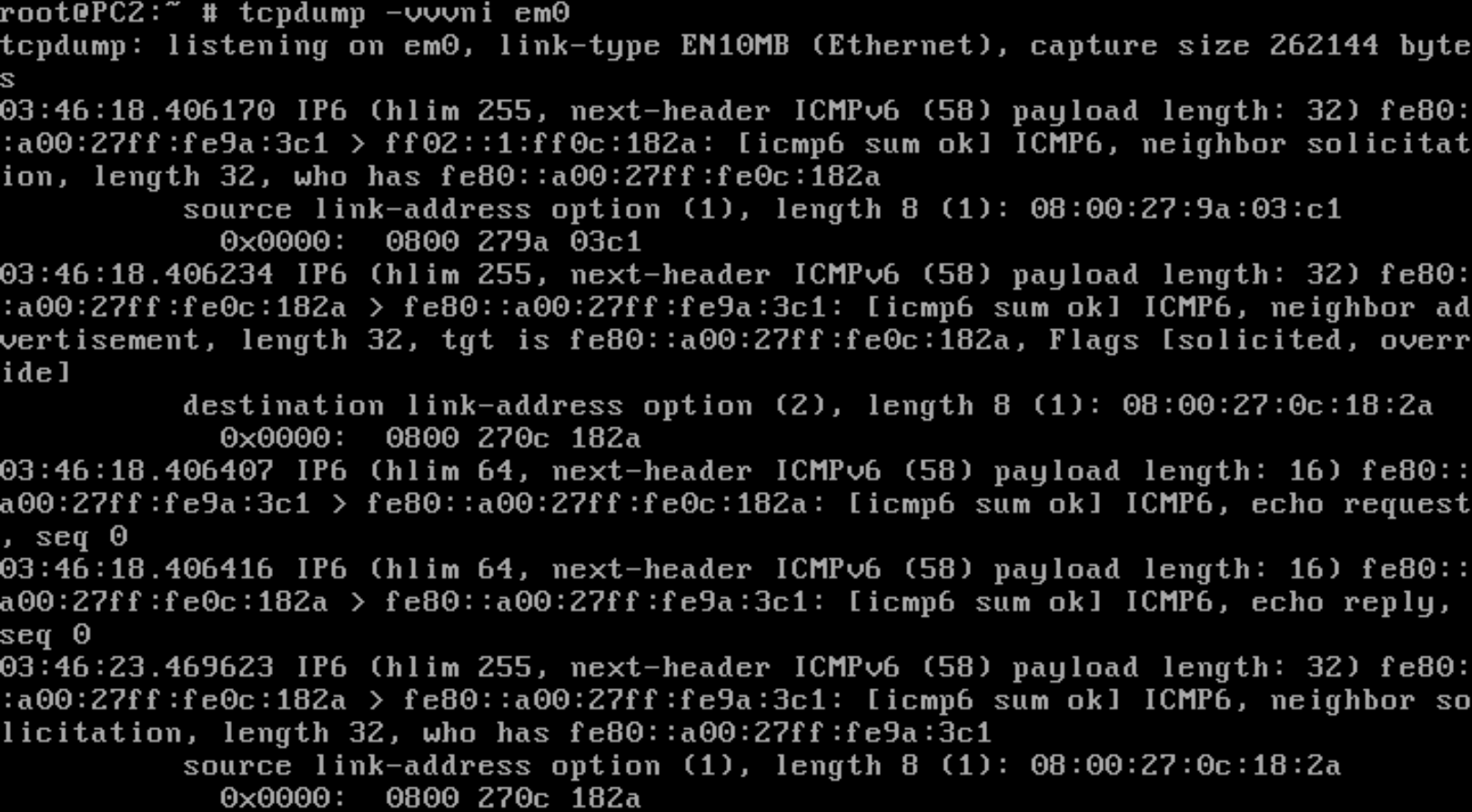
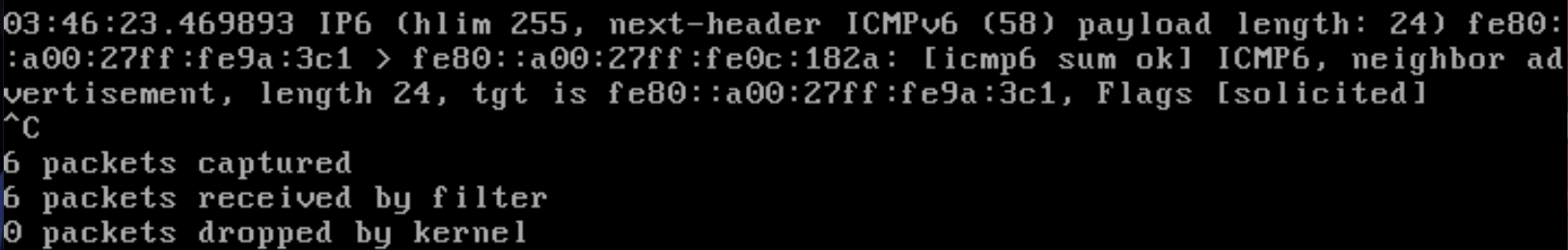
Εκτελούμε “**ndp -c**”.

***1.28***

Εκτελούμε σε νέο παράθυρο στο PC2 “**tcpdump -vvvni em0**”.

***1.29***

Βλέπουμε 6 IPv6 πακέτα, ένα Neighbor Solicitation με το οποίο ο PC1 ρωτάει για τη MAC του PC2, ένα Neighbor Advertisement που περιέχει την απάντηση στο παραπάνω και στη συνέχεια το ICMPv6 Request και Reply. Τέλος βλέπουμε άλλο ένα ζέυγος NS NA, με την αντίστροφη όμως φορά επικοινωνίας.



***1.30***

Μεταφέρονται μηνύματα του πρωτοκόλλου ICMPv6, τα οποία προσδιορίζονται από την τιμή 58 του πεδίου next-header.

***1.31***

1. **PC1 → PC2** (Neighbor Solicitation)
2. **PC2 → PC1** (Neighbor Advertisement)
3. **PC1 → PC2** (ICMPv6 Echo Request)
4. **PC2 → PC1** (ICMPv6 Echo Reply)
5. **PC2 → PC1** (Neighbor Solicitation)
6. **PC1 → PC2** (Neighbor Advertisement)

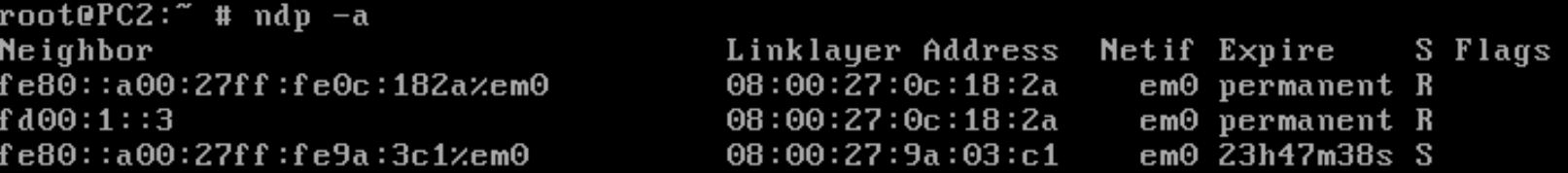
***1.32***

Το πρώτο πακέτο που καταγράψαμε έχει διεύθυνση προορισμού την ff02::1:ff0c:182a. Η διεύθυνση αυτή προκύπτει προσθέτοντας στο πρόθεμα ff02:0:0:0:0:1:ff00:0/104 τα τελευταία 24 bit της unicast διεύθυνσης του προορισμού. Εν προκειμένω, προορισμός ήταν η διεύθυνση του PC2 (fe80::a00:27ff:fe0c:182a), εκ της οποία τελευταία 24 bits είναι τα 0c:182a, οπότε και προκύπτει η παραπάνω διεύθυνση ως multicast Solicited Node.

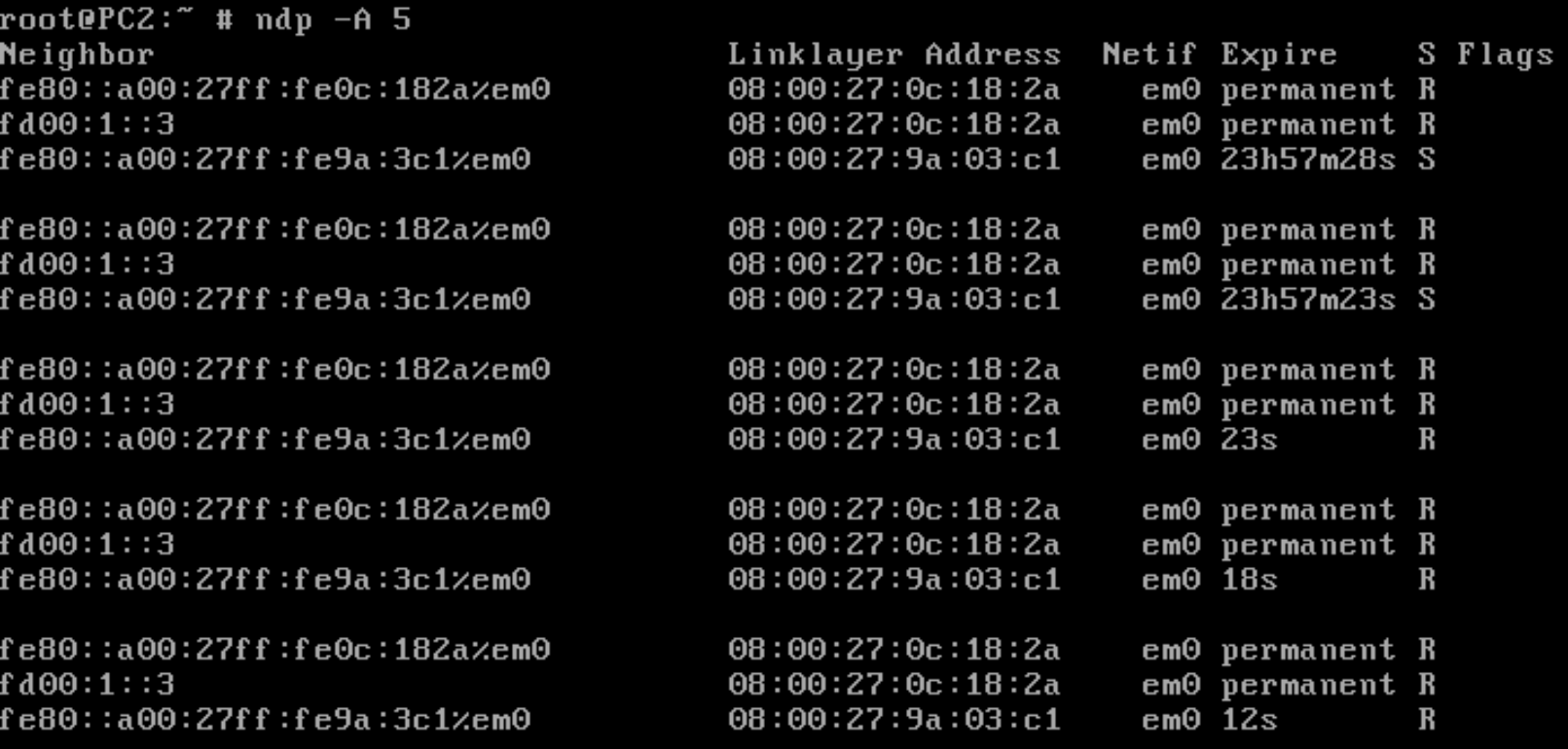
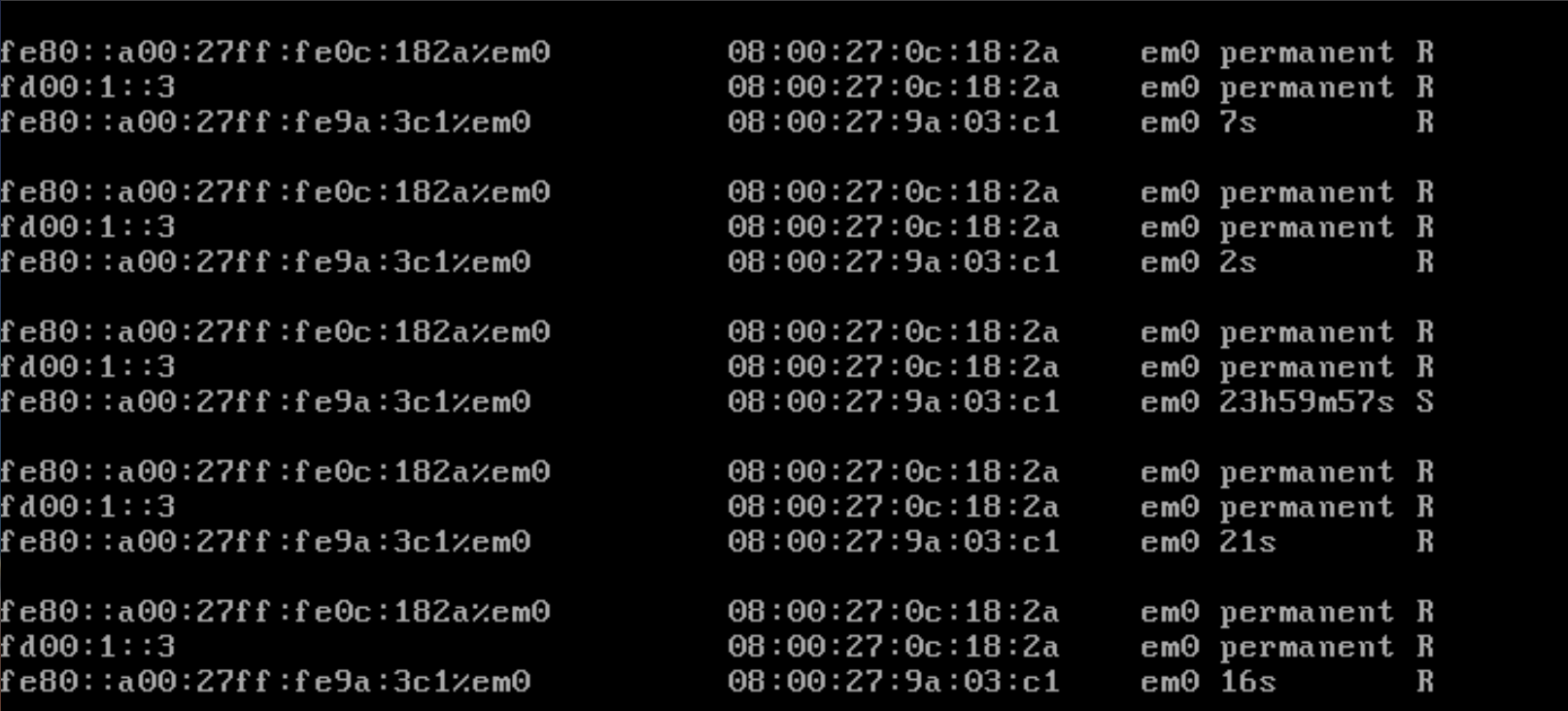
***1.33***

Το δεύτερο πακέτο που καταγράψαμε έχει διεύθυνση προορισμού την fe80::a00:27ff:fe9a:3c1, δηλαδή την link-local address του PC1.

***1.34***

Η εγγραφή που αφορά το PC1 είναι σε κατάσταση Stale ενώ η διάρκεια ζωής της είναι 24 ώρες (στην αρχή).

***1.35***

Εκτελούμε στο PC2 “**ndp –A 5**“ ώστε να εμφανίζει τον πίνακα γειτόνων ανά 5 δευτερόλεπτα. Εκτελούμε στο PC1 “**ping6 PC2**”. Παρατηρήσαμε τις καταστάσεις R (Reachable) και S (Stale).

***1.36***

Η κατάσταση R διαρκεί 23sec, μετά την παρέλευση των οποίων μεταπίπτει στην κατάσταση S για λίγο, προτού ξαναμπεί στην κατάσταση R.

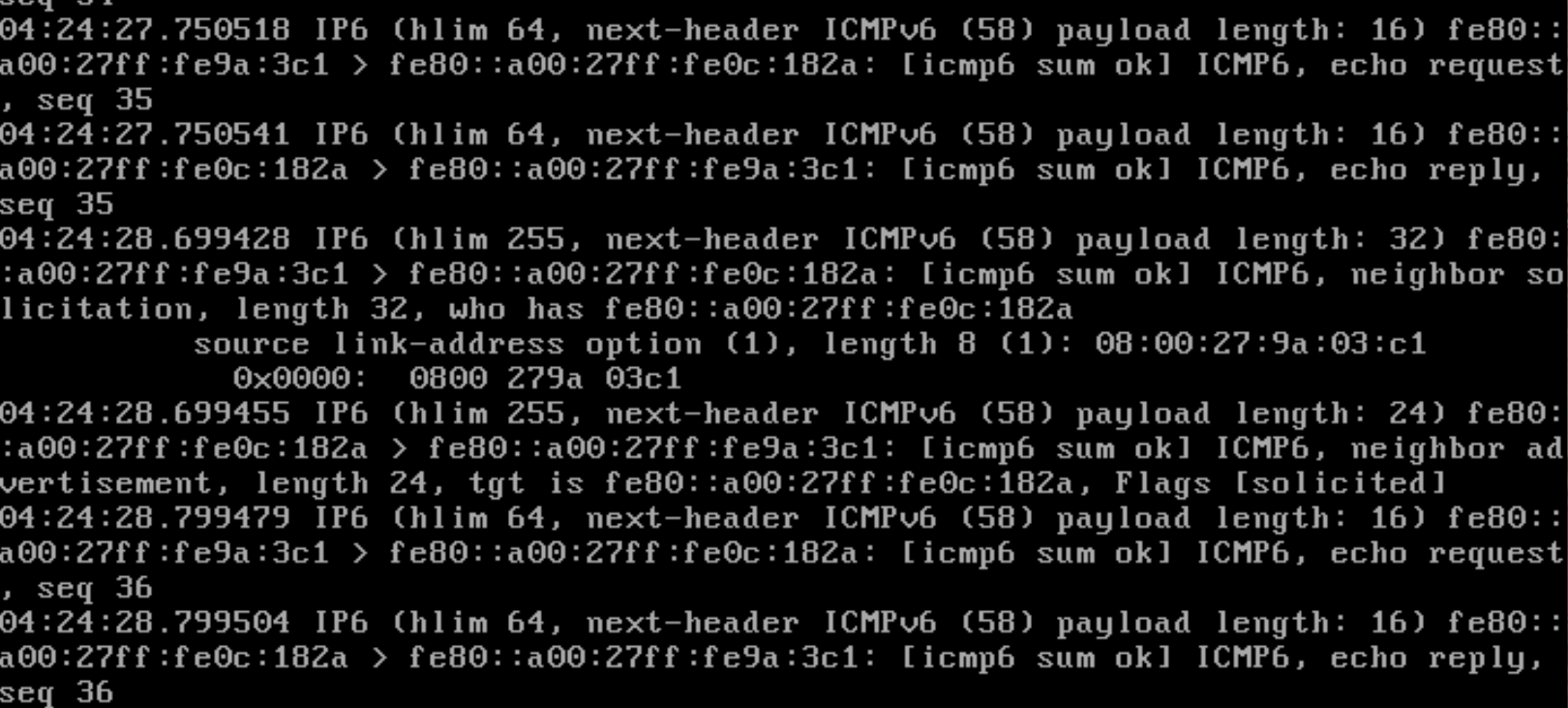
***1.37***

Η διάρκεια ζωής της κατάστασης S είναι ένα 24ωρο.

***1.38***

Παρατηρούμε τις ίδιες καταστάσεις, με τη διαφορά πως τώρα όταν λήξουν τα 23s της κατάστασης R, μεταβαίνουμε στην κατάσταση S όπου και ο χρόνος βαίνει μειούμενος από τις 24 ώρες χωρίς να ξαναπάμε στην κατάσταση R.

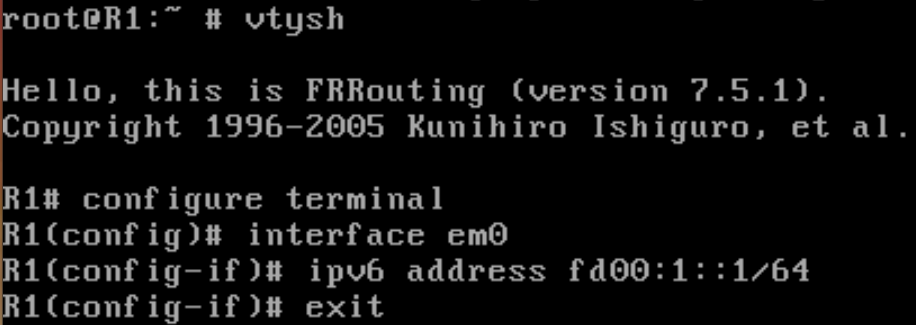
***1.39***

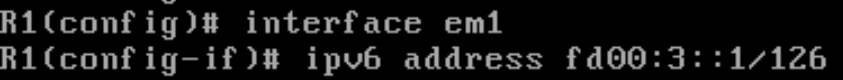
Παρατηρούμε επιπλέον πακέτα NS και NA, τα οποία επαναλαμβάνονται ανά 23s, όποτε δηλαδή λήγει η κατάσταση R.

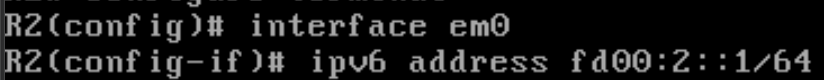
**Άσκηση 2: SLAAC και Στατική δρομολόγηση IPv6**

***2.1***

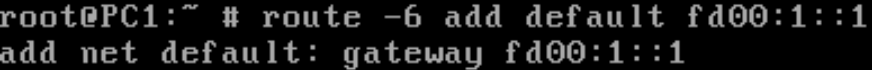
***2.2***

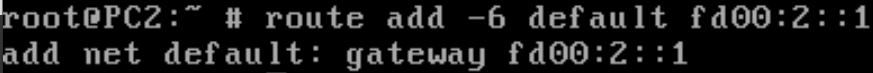
***2.3***

***2.4***

***2.5***

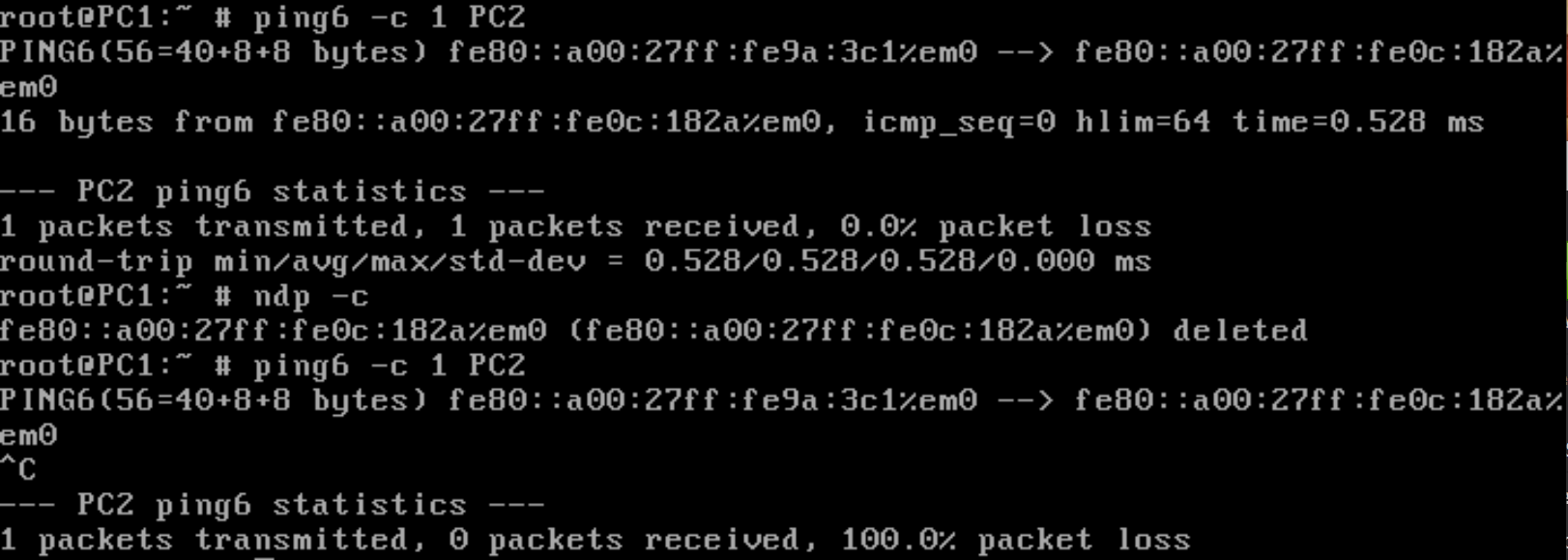
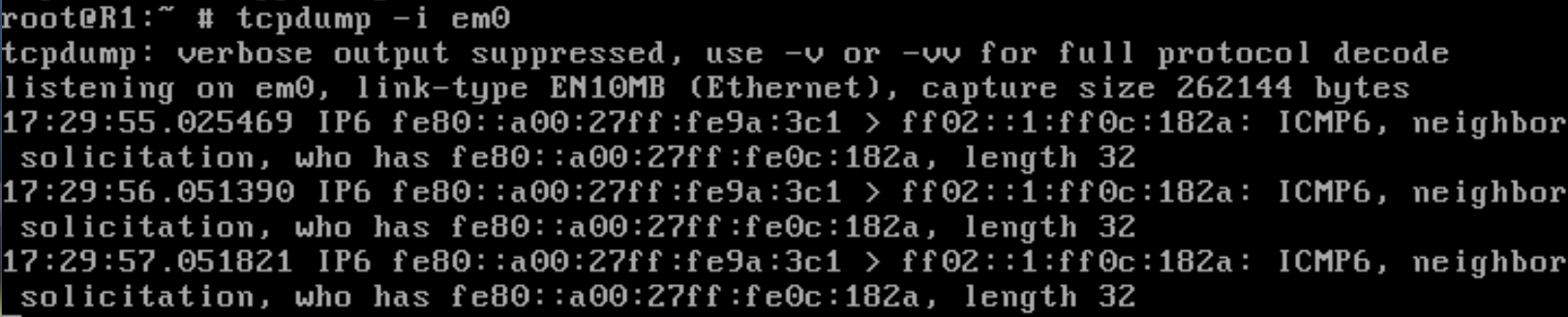
***2.6***

***2.7***

***2.8***

***2.9***

***2.10***

Το ping αποτυγχάνει, καθώς το PC1 κάνει στην αρχή NS, ώστε να μάθει που είναι το PC2, χωρίς ωστόσο να παίρνει απάντηση αφού ούτε ο R1 γνωρίζει.

***2.11***

Όπως βλέπουμε, παράχθηκαν 3 NS με προορισμό την ff02::1:ff0c:182a, δηλαδή την Solicited Node του PC2.

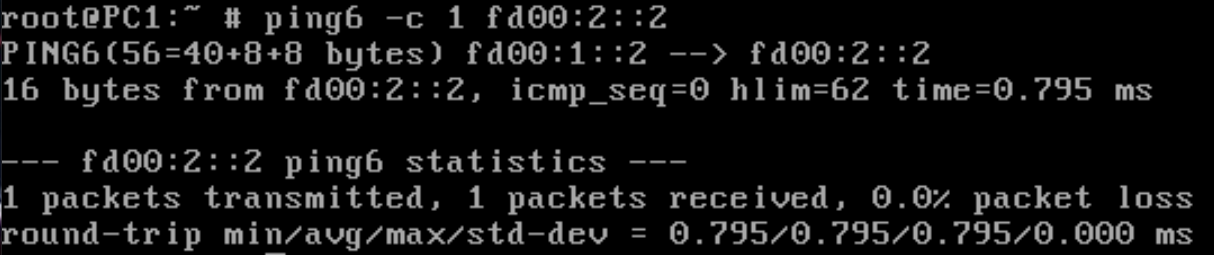
***2.12***

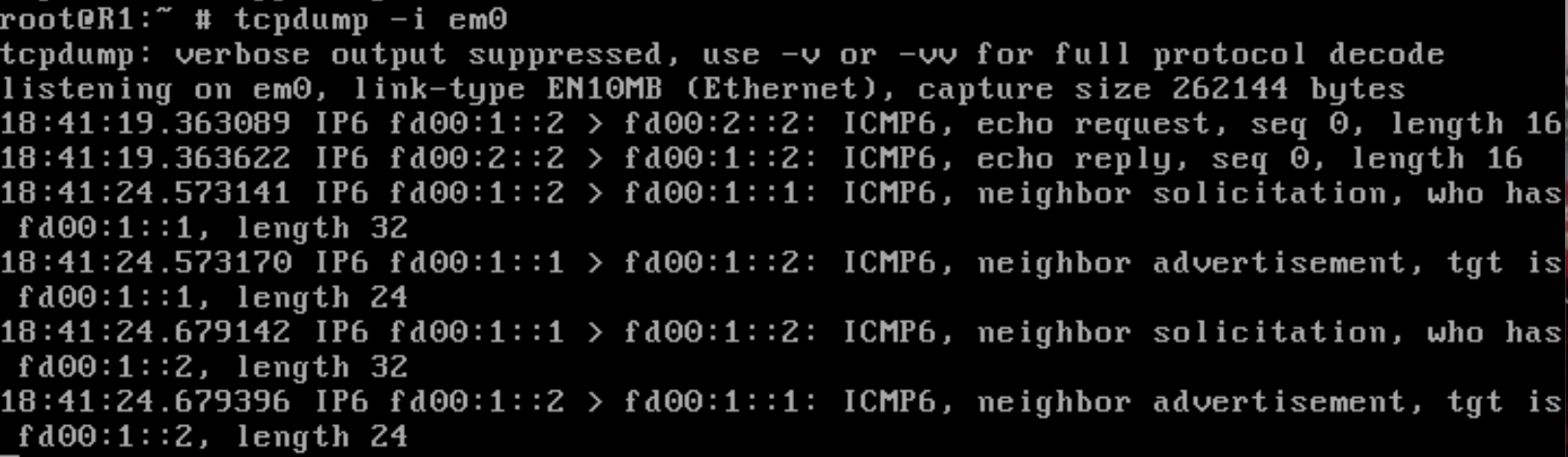
***2.13***

Ακόμα δε μπορούμε να κάνουμε ping, καθώς δε μπορεί να δρομολογήσει ο R2 την απάντηση προς το PC1.

***2.14***

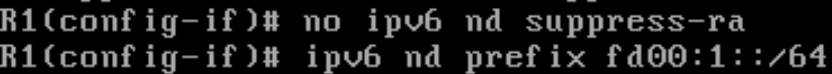
***2.15***

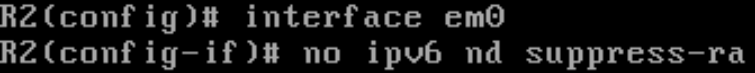
Πλέον το ping επιτυγχάνει, όσον αφορά τη στατική διεύθυνση fd00:2::2 που ορίσαμε στον PC2.



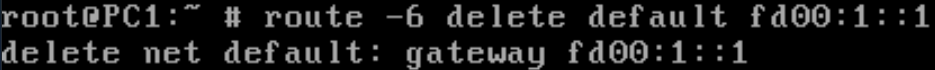
***2.16***

Εκτελούμε στο R1 σε GCM “**interface em0**” → “**no ipv6 nd suppress-ra**”.

***2.17***

***2.18***

***2.19***

***2.20***

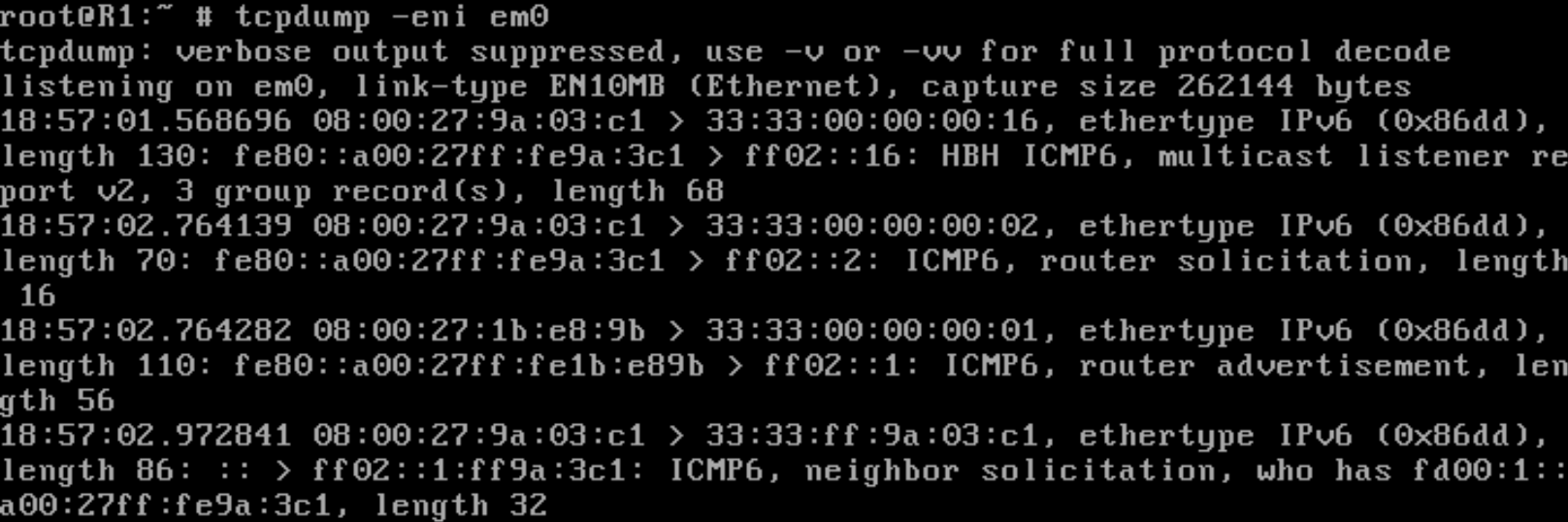
***2.21***

Εκτελούμε “**tcpdump -eni em0**”.

***2.22***

Εκτελούμε “**service netif restart**”.

***2.23***

Καταγράφουμε τα παρακάτω πακέτα:

Έχουμε τα παρακάτω πακέτα:

1. Tο PC1 στέλνει multicast IPv6 over Ethernet στη MAC address 33:33:00:00:00:16, η οποία παράχθηκε βάζοντας στην αρχή το πρόθεμα 33:33 και στη συνέχεια τα τελευταία 32 bits της IPv6 προορισμού, η οποία εν προκειμένω είναι η ff02::16 (IPv6 source: 90: :: ).
2. Το PC1 στέλνει Router Solicitation στο MAC/IPv6 destination 33:33:00:00:00:02 / ff02::2, ώστε να ζητήσει να λαμβάνει μηνύματα RA.
3. Ο R1 απαντάει στο destination MAC/IPv6 33:33:00:00:00:01 / ff02::1 με Router Advertisement.
4. Το PC1 στέλνει Neighbour Solicitation (destination MAC 33:33:ff:9a:03:c1, IPv6 source: :: (ακαθόριστη), IPv6 destination: ff02::1:ff9a:3c1 το οποίο είναι το solicited node του προορισμού), με το οποίο ρωτάει αν έχει κανείς την IPv6 του. (Δεν λαμβάνει απάντηση)

***2.24***

Το συγκεκριμένο NS χρησιμοποιείται για την ανίχνευση ταυτόσημων διευθύνσεων (DAD), ώστε να ορίσει σωστά τη διεύθυνσή του.

***2.25***

Βλέπουμε πως χρησιμοποιείται η undefined “::”, καθώς έτσι ορίζει το DAD.

***2.26***

Την fe80::a00:27ff:fe9a:3c1.

***2.27***

Έχουμε:

* NS → ff02::1:ff9a:3c1 (Solicited node)
* RS → ff02::2 (Multicast, όλοι οι δρομολογητές σε αυτή τη ζεύξη)
* RA → ff02::1 (Multicast, όλοι οι κόμβοι σε αυτή τη ζεύξη)

***2.28***

Έχουμε:

• NS → 33:33:ff:9a:03:c1

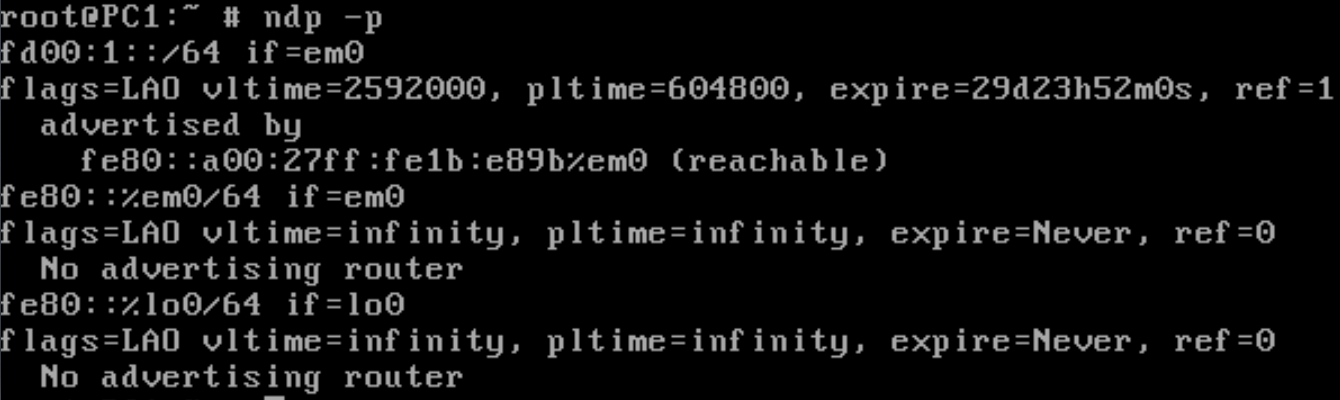
• RS → 33:33:00:00:00:02

• RA → 33:33:00:00:00:01

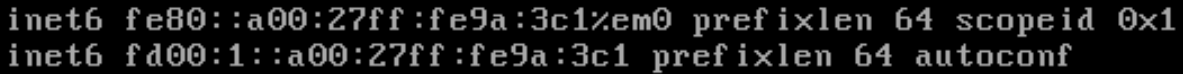
Όπως είδαμε και στο 2.23 το πρόθεμα 33:33 χρησιμοποιείται για IPv6 Multicast Over Ethernet, ενώ ως τελευταία 32 bit μπαίνουν τα τελευταία 32 της IPv6 προορισμού.

***2.29***

Παρατηρούμε πως πλέον η πρώτη εγγραφή έχει επιπρόσθετα το flag A και πεπερασμένους χρόνους στα διάφορα πεδία.



***2.30***

Τις παρακάτω.

***2.31***

Βλέπουμε πως υπάρχει η προκαθορισμένη διαδρομή και προέκυψε ως η link-state διεύθυνση του δρομολογητή με τον οποίο επικοινώνησε πριν το PC1 (δηλαδή η em0 του R1).

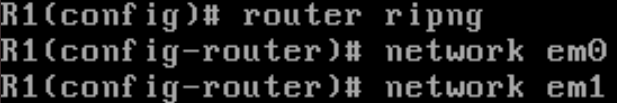
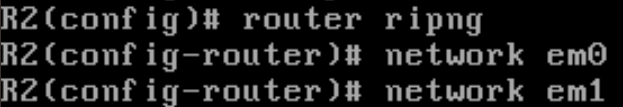
***2.32***

Από το PC2 μπορούμε να κάνουμε ping μόνο στην 2η διεύθυνση του PC1, ενώ από το R1 και στις 2, καθώς είναι στο ίδιο LAN, οπότε μπορεί να επικοινωνήσει και με τη link-local address.

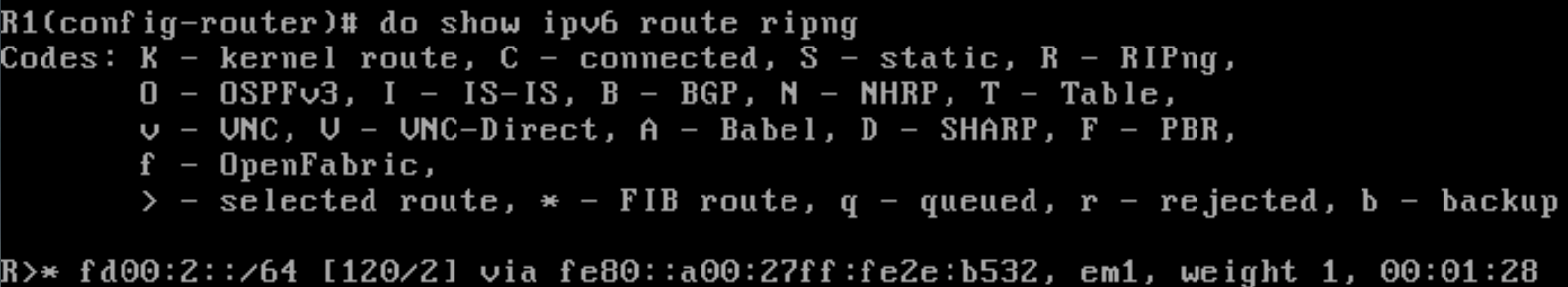
**Άσκηση 3: Δυναμική δρομολόγηση IPv6**

***3.1***



***3.2***

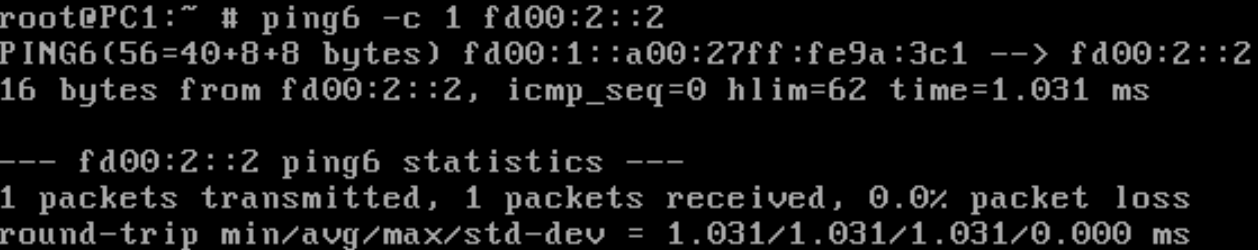
***3.3***

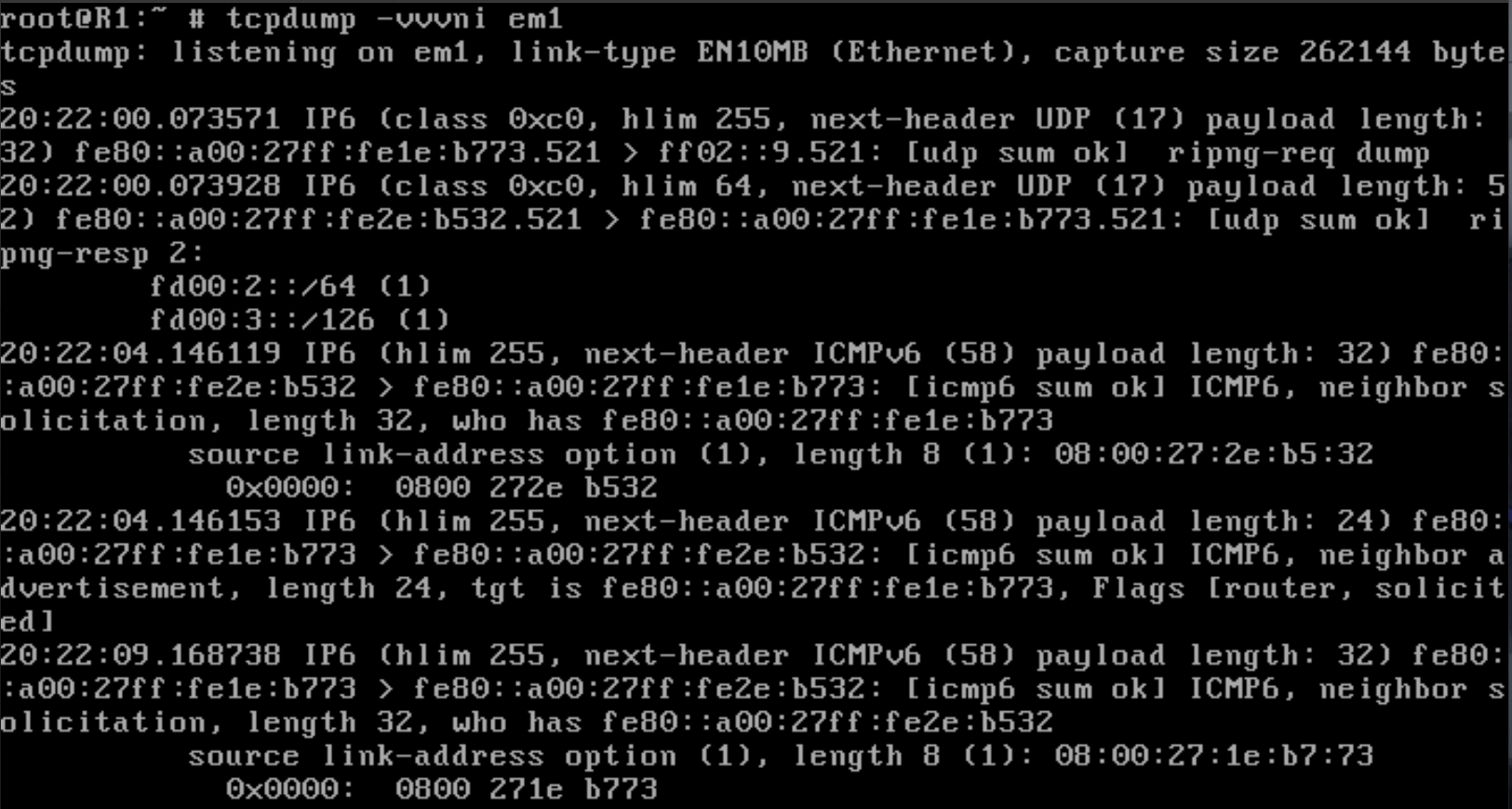
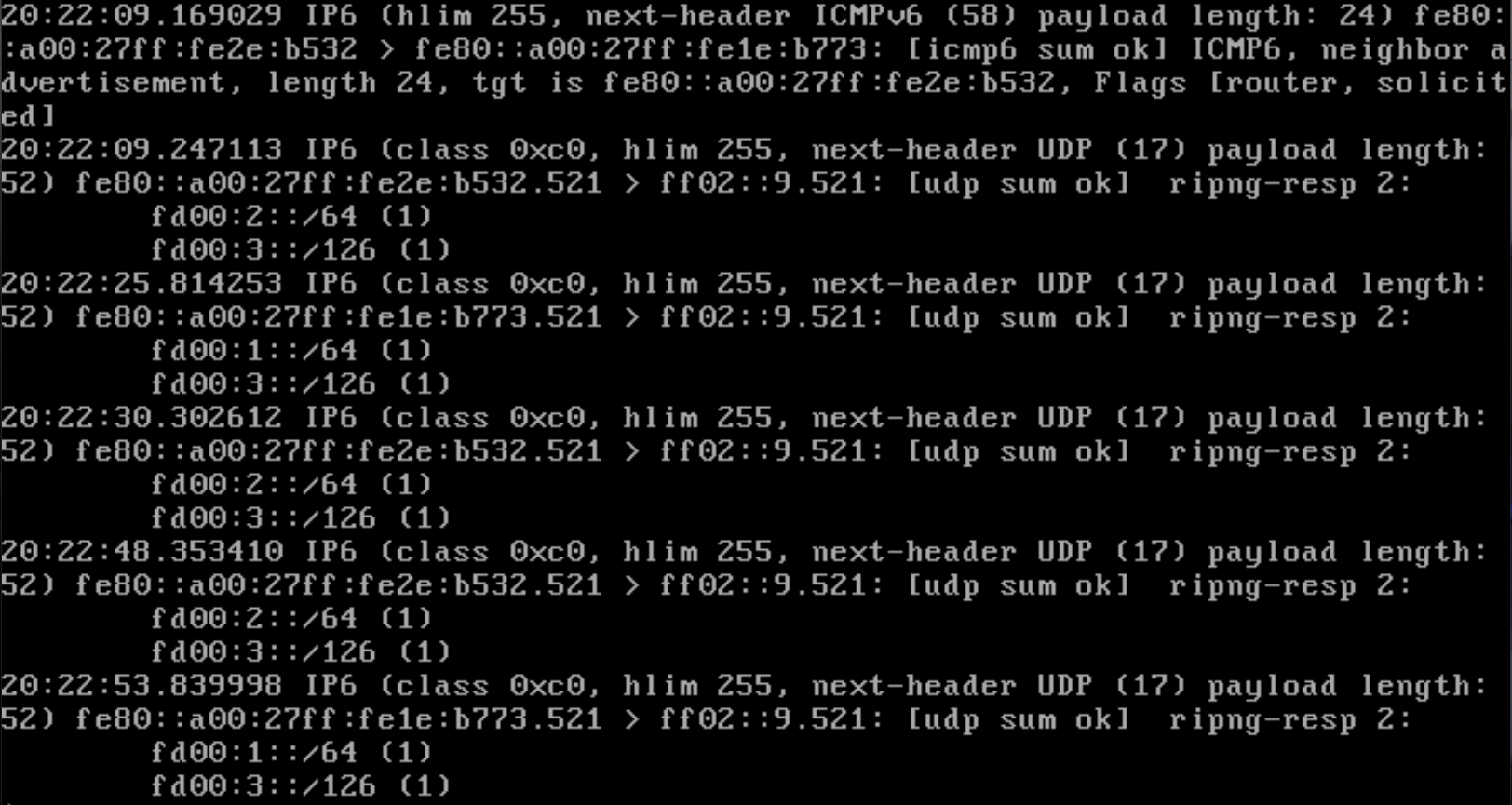
Βλέπουμε μία μόνο εγγραφή

***3.4***

Επόμενος κόμβος για το fd00:2::/64 είναι η fe80::a00:27ff:fe382e:b532, δηλαδή η link-local address του em1 του R2.

***3.5***

Μπορούμε χρησιμοποιώντας την private address “**ping6 –c 1 fd00:2::2**”.

***3.6***

***3.7***

Παρατηρούμε, αρχικά, ένα πακέτο ripng-request (R1em1 ff02::9) ακολουθούμενο από ένα πακέτο ripng-response (R2em1 R1em1), όπου R1em1, R2em1 οι link-local διευθύνσεις τους. Μετά βλέπουμε την διαδικασία Neigbour discovery και ύστερα ακολουθούν πακέτα ripng-response εκπεμπόμενα από τις WAN διεπαφές των R1, R2. Το μεν R1 στέλνει διαφημίσεις για το LAN1 και το WAN1, ενώ το R2 για το LAN2 και το WAN1. Διεύθυνση προορισμού σε αυτά είναι η ff02::9, η οποία αποτελεί multicast address για RIP routers.

***3.8***

Το hop\_limit έχει τιμή 255, ώστε να διασφαλιστεί ότι δε θα περάσει από ενδιάμεσο δρομολογητή.

***3.9***

Χρησιμοποιείται το UDP και η θύρα 521, ενώ στο RIP είχαμε μεν UDP, αλλά χρησιμοποιούνταν η θύρα 520.

***3.10***

Εκτελούμε στα R1, R2 “**no router ripng**”.

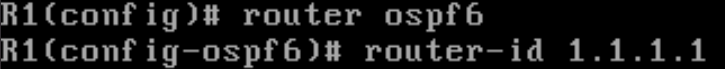
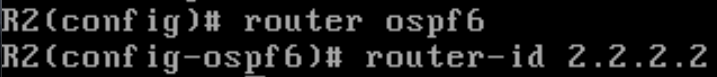
***3.11***

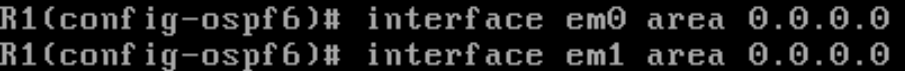
Εκτελούμε στα R1, R2 σε GCM “**do write memory**”.

***3.12***

Εκτελούμε σε απλό terminal στα R1, R2 “**service frr restart**”.

***3.13***

Εκτελούμε τα παρακάτω στον R1 και αντίστοιχα στον R2.

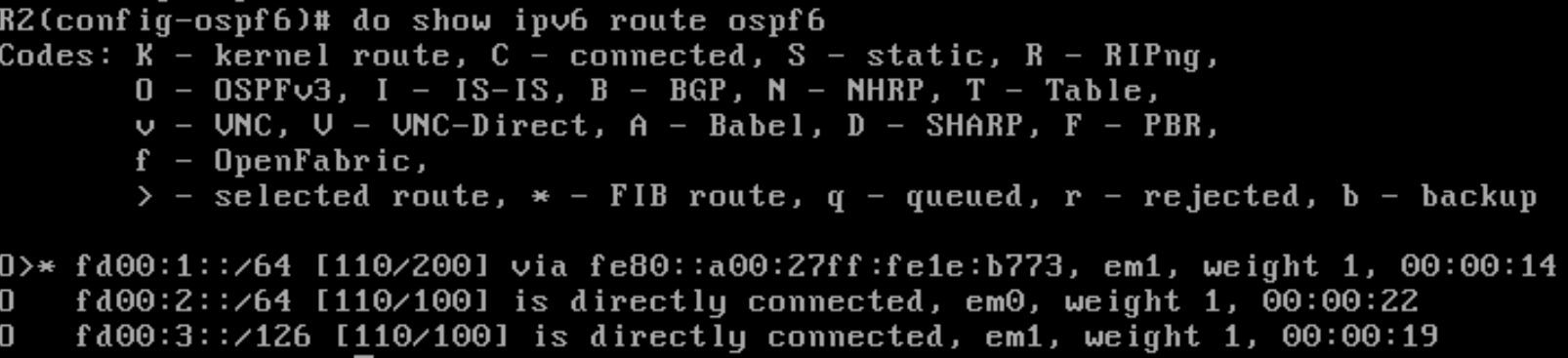
***3.14***

***3.15***

Εκτελούμε αντίστοιχα στον R2 “**interface em0 area 0.0.0.0**” και “**interface em1 area 0.0.0.0**”.

***3.16***

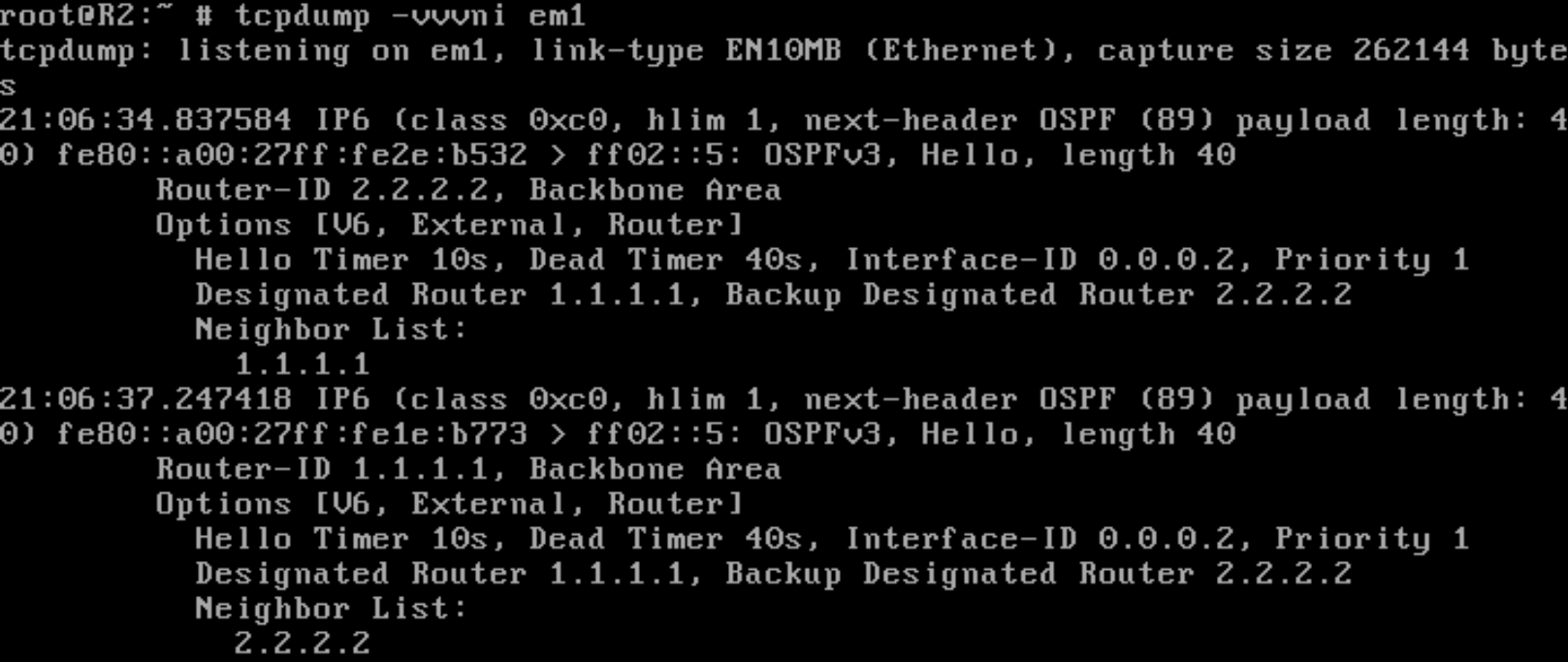
Βλέπουμε τις παρακάτω 3 εγγραφές. Το κόστος για το LAN1 (πρώτη εγγραφή) είναι 200 (100+100), ενώ για τα LAN2 και WAN1 (δεύτερη και τρίτη εγγραφή αντίστοιχα) είναι 100, καθώς το R2 είναι άμεσα συνδεδεμένο με αυτά.



***3.17***

Είναι η διεύθυνση fe80::a00:27ff:fe1e:b773, η οποία αποτελεί την link-local address της em1 του R1.

***3.18***

Εκτελούμε στο R2 “**tcpdump -vvvni em1**”. Παρουσιάζουμε ενδεικτικά κάποιες καταγραφές.

***3.19***

Παρατηρούμε να εκπέμπονται OSPFv3 Hello πακέτα από τις διεπαφές em1 των R1 και R2, με διεύθυνση προορισμού την ff02::5, η οποία αφορά multicast διευθύνσεις OSPFIGP.

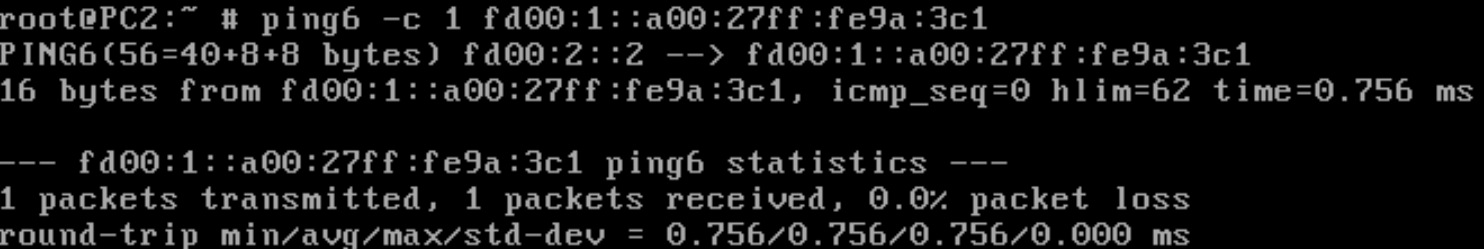
***3.20***

Hop Limit = 1.

***3.21***

Βλέπουμε πως χρησιμοποιεί τον αριθμό πρωτοκόλλου 89, ίδιος δηλαδή με αυτόν του OSPFv2.

***3.22***

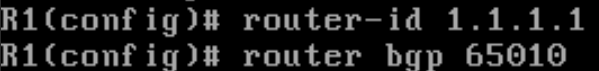
Μπορούμε να κάνουμε στην private address του PC1.

***3.23***

Εκτελούμε στα R1 και R2 “**no router ospf6**”.

***3.24***

Επανεκκινούμε την υπηρεσία frr στα R1/R2.

***3.25***

***3.26***

***3.27***



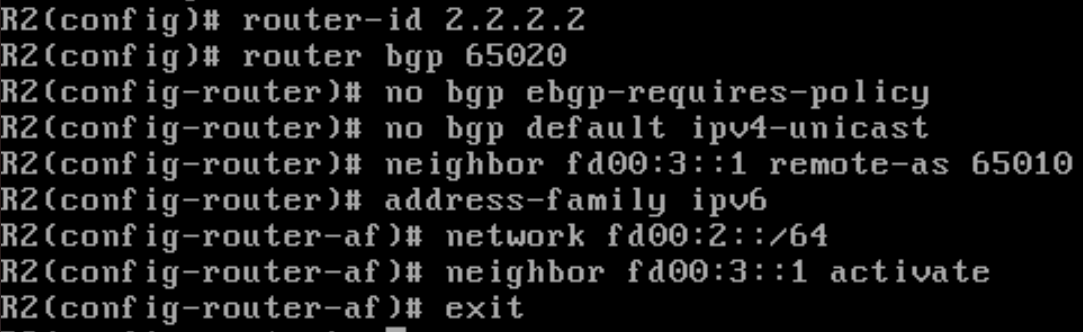
***3.28***

***3.29***

***3.30***

***3.31***

***3.32***

Εκτελούμε τις εντολές για τον R2.

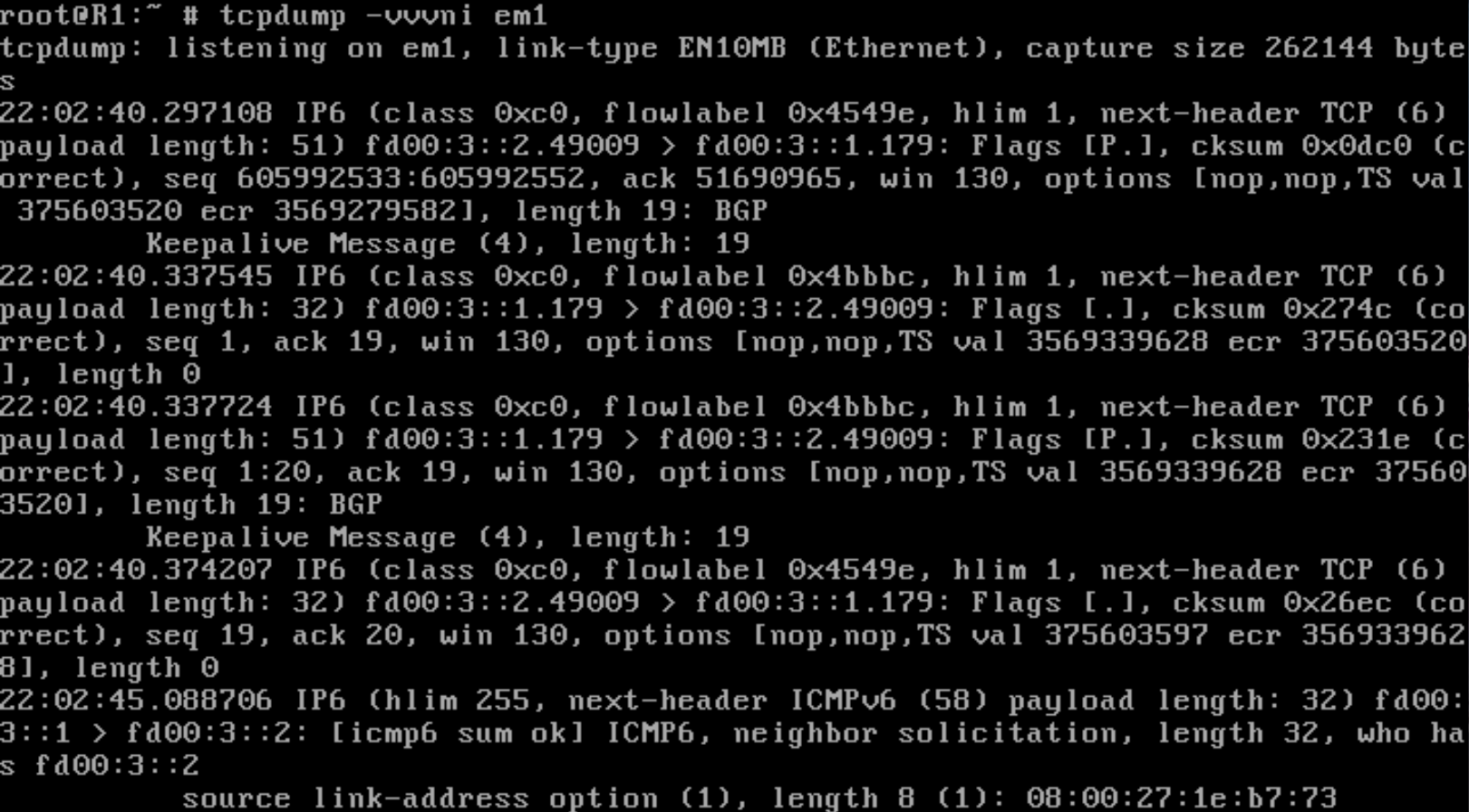
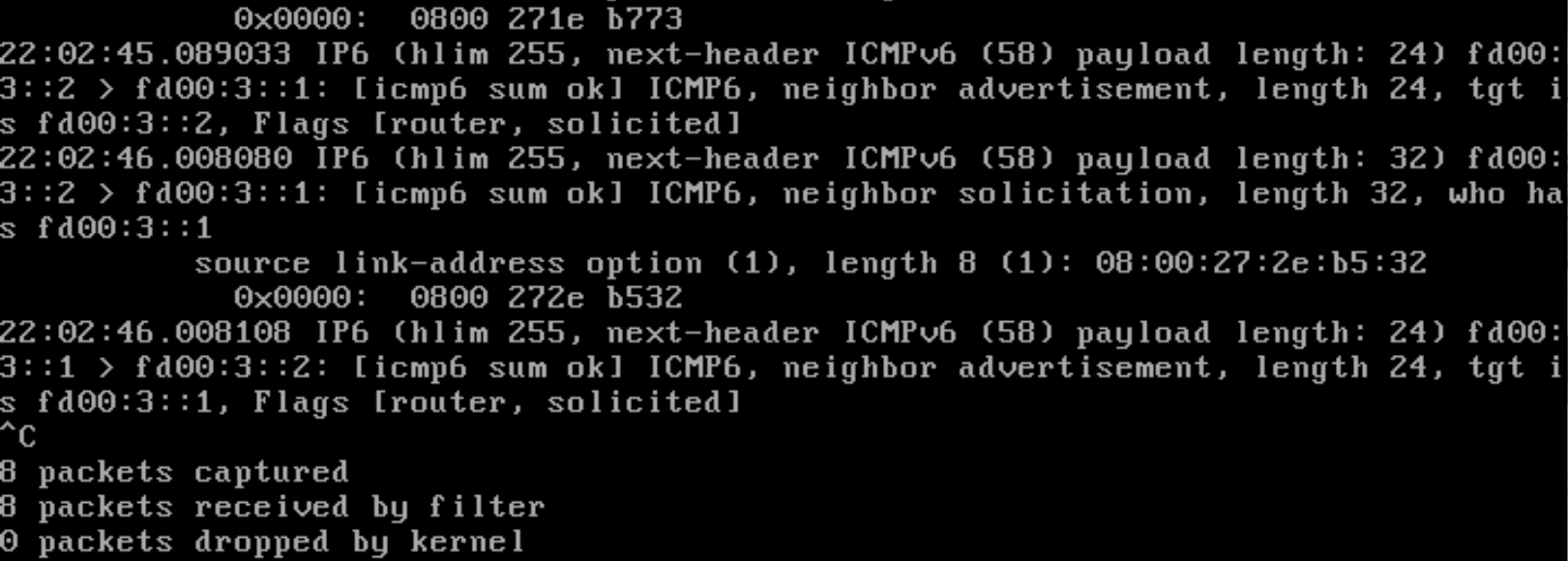
***3.33***

Με “**do show ipv6 route bgp**” βλέπουμε μία μόνο εγγραφή, για το LAN2:

***3.34***

Διεύθυνση του επόμενου κόμβου για το LAN2 είναι η fe80::a00:27ff:fe2e:b532, η οποία αποτελεί την link-local address της em1 του R2.

***3.35***

Εκτελούμε “**tcpdump -vvvni em1**”.

***3.36***

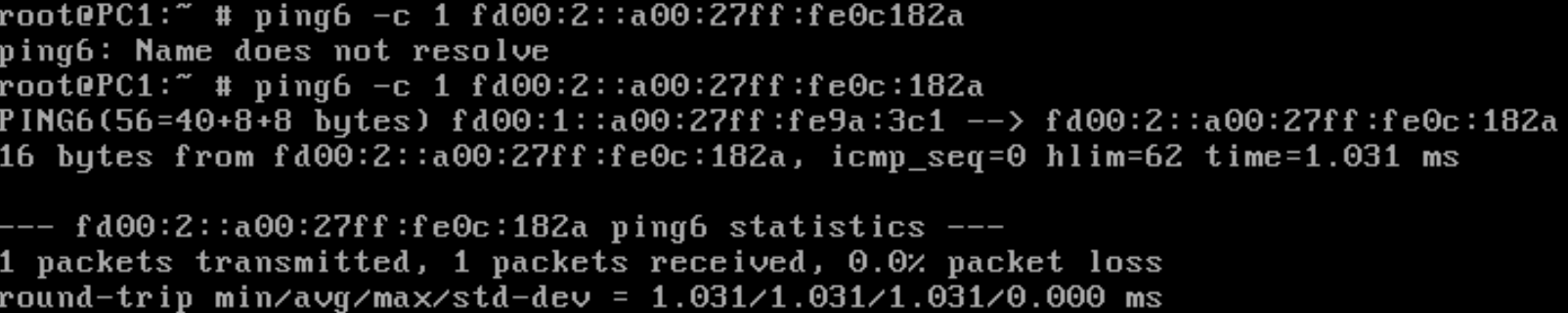
Βλέπουμε μηνύματα BGP Keepalive μεταξύ των R1-R2. Χρησιμοποιείται το πρωτόκολλο TCP και η θύρα (προορισμού) 179, όπως στο IPv4. (θύρα πηγής δυναμική)

***3.37***

hlim=1.

***3.38***

Μπορούμε στις παρακάτω 2 διευθύνσεις:



***3.39***

Εκτελούμε στο PC1 αρχικά “**reboot**”. Συνδεόμαστε πάλι και εκτελούμε τα ζητούμενα:

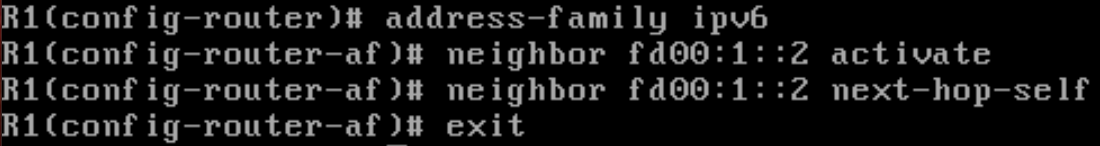
***3.40***

***3.41***

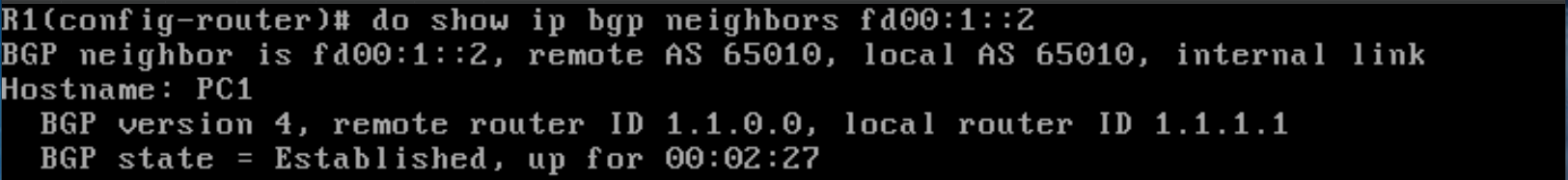
***3.42***

***3.43***

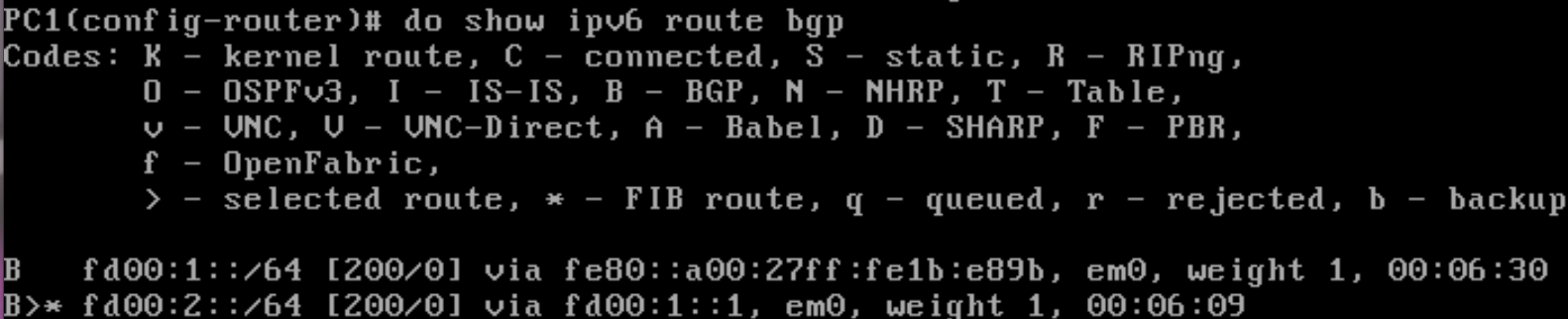
***3.44***

***3.45***

***3.46***

Με την εντολή “**do show ip bgp neighbors fd00:1::2**” στον R1 ή “**do show ip bgp neighbors fd00:1::1**” στο PC1 βλέπουμε πως αναφέρεται στην πρώτη σειρά η πληροφορία “internal link”

***3.47***

Βλέπουμε τις παρακάτω 2 εγγραφές, για τα LAN1 και LAN2.

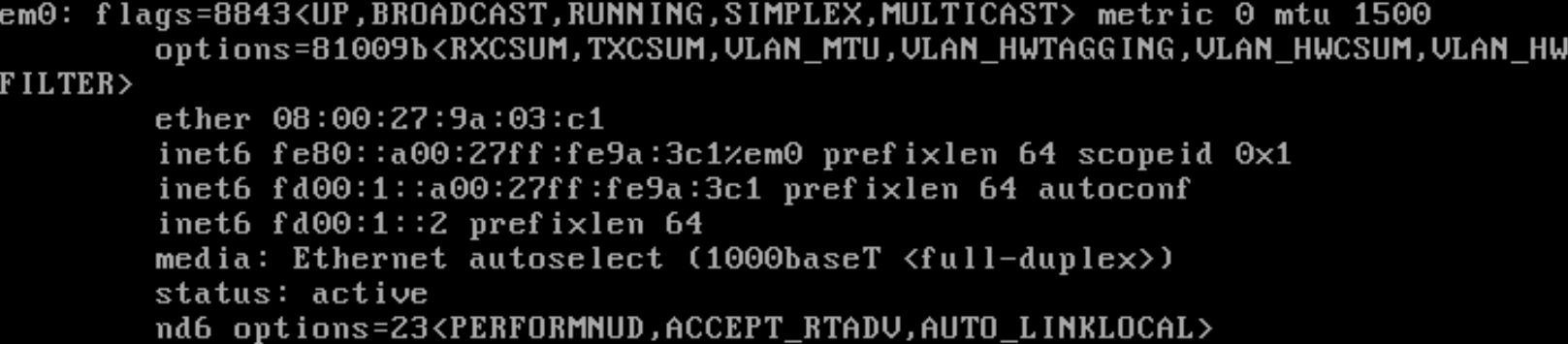
***3.48***

Διότι όντας άμεσα συνδεδεμένο, επιλέγει αυτή τη διαδρομή για το fd00:1::/64 (LAN1)

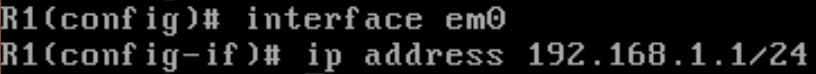
***3.49***

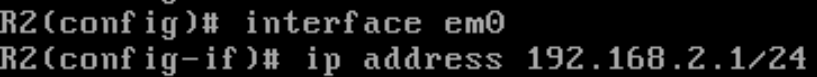
Είναι η fd00:1::1, η οποία είναι η private IPv6 address της em0 του R1

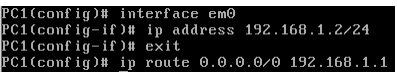
***3.50***

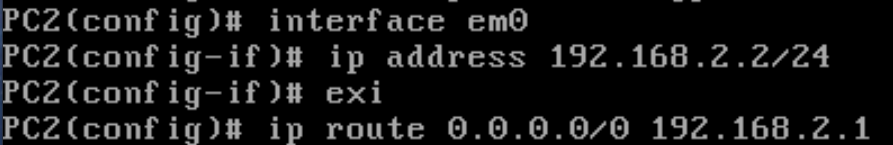
Μπορούμε να κάνουμε στις παρακάτω 2 διευθύνσεις:

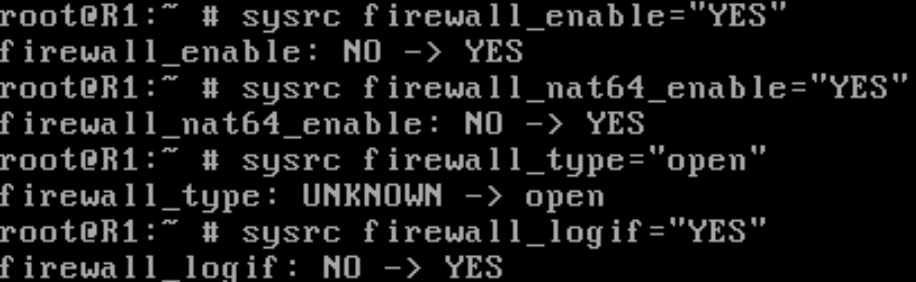
**Άσκηση 4: Μηχανισμός μετάβασης 464 XLAT**

***4.1***

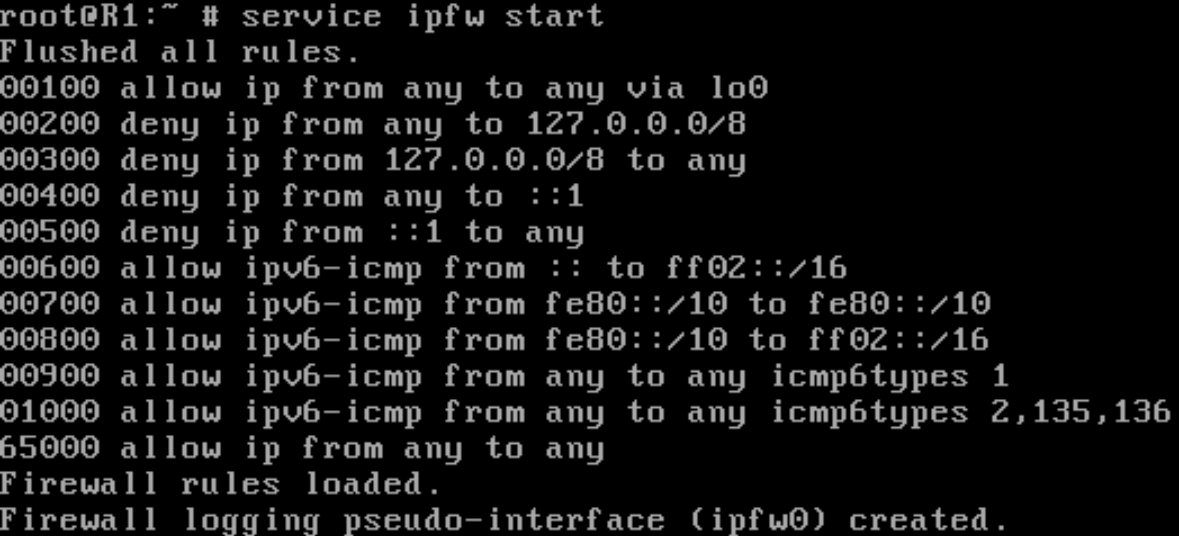
***4.2***

***4.3***

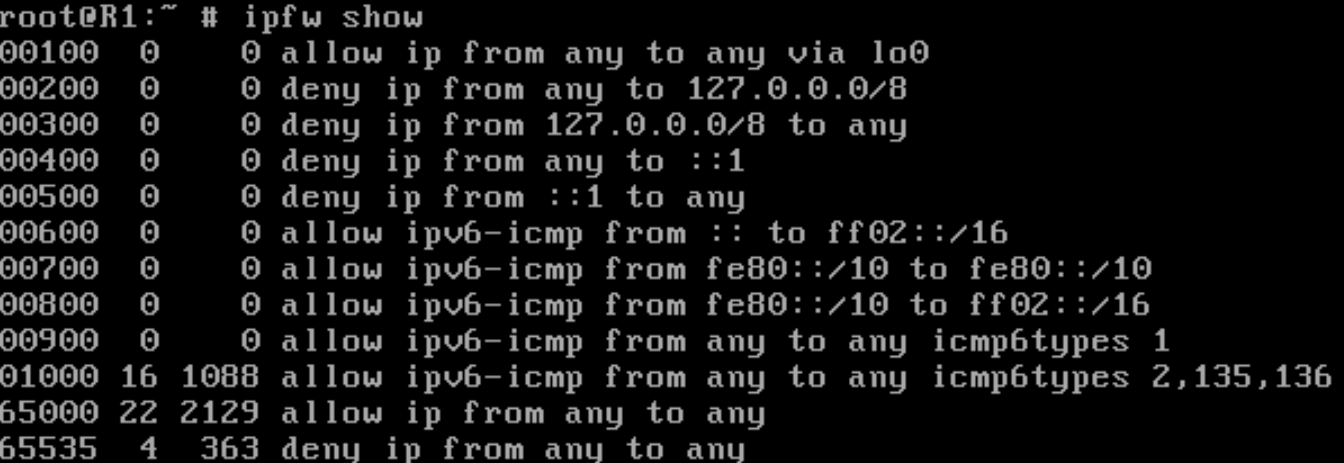
***4.4***

***4.5***

***4.6***

Εκκινούμε το ipfw στο R1.

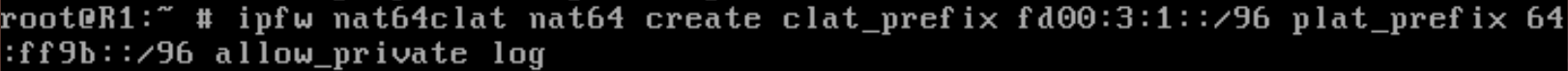
***4.7***

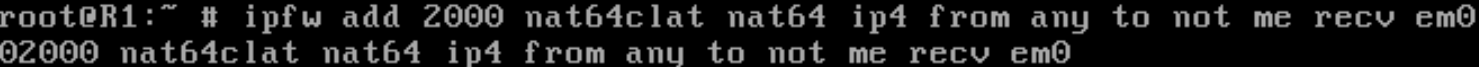
Περιέχει τους παρακάτω 12 κανόνες:

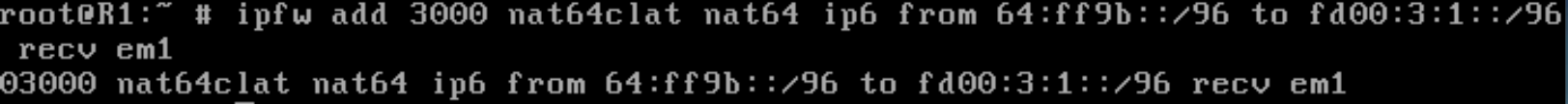
***4.8***

Μπορούμε αλλά μόνο στις 2 τελευταίες από τις παρακάτω διευθύνσεις:



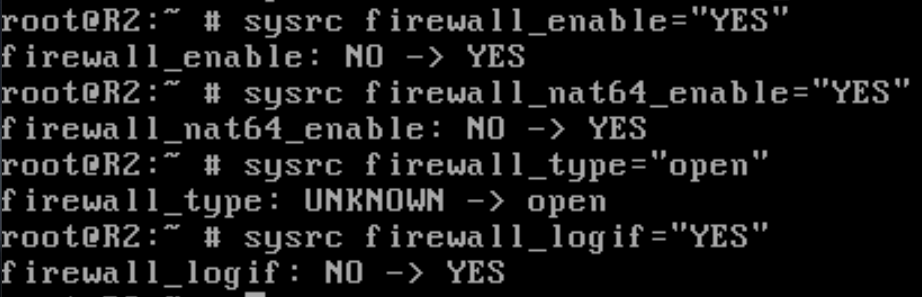
***4.9***

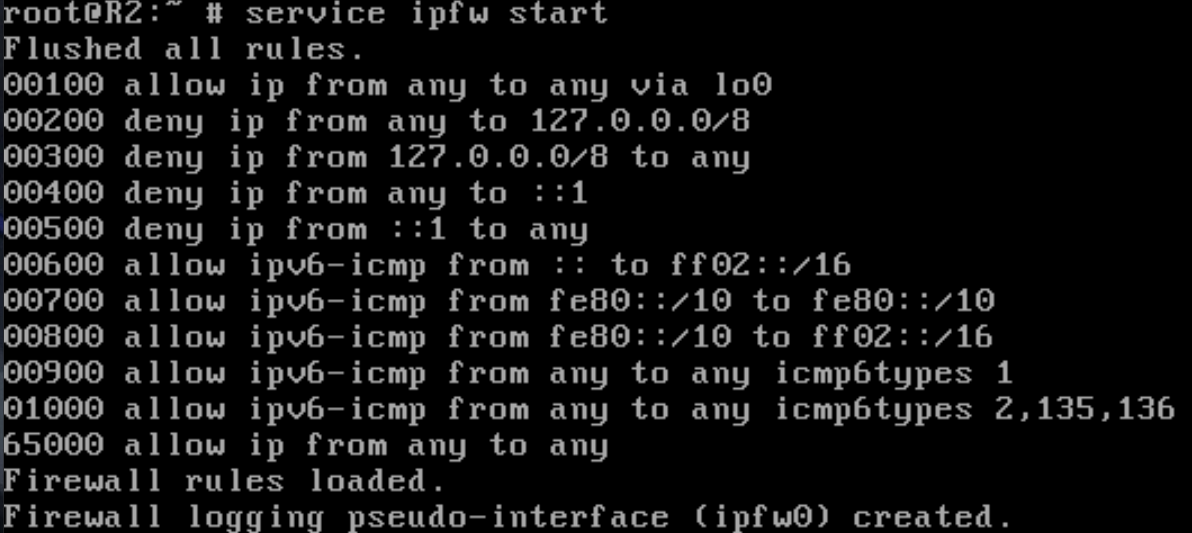
***4.10***

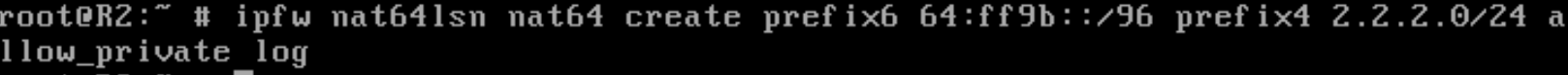
***4.11***

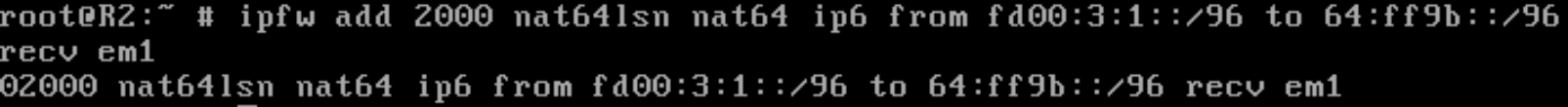
***4.12***

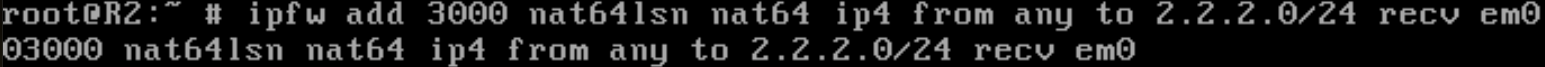
***4.13***





***4.14***

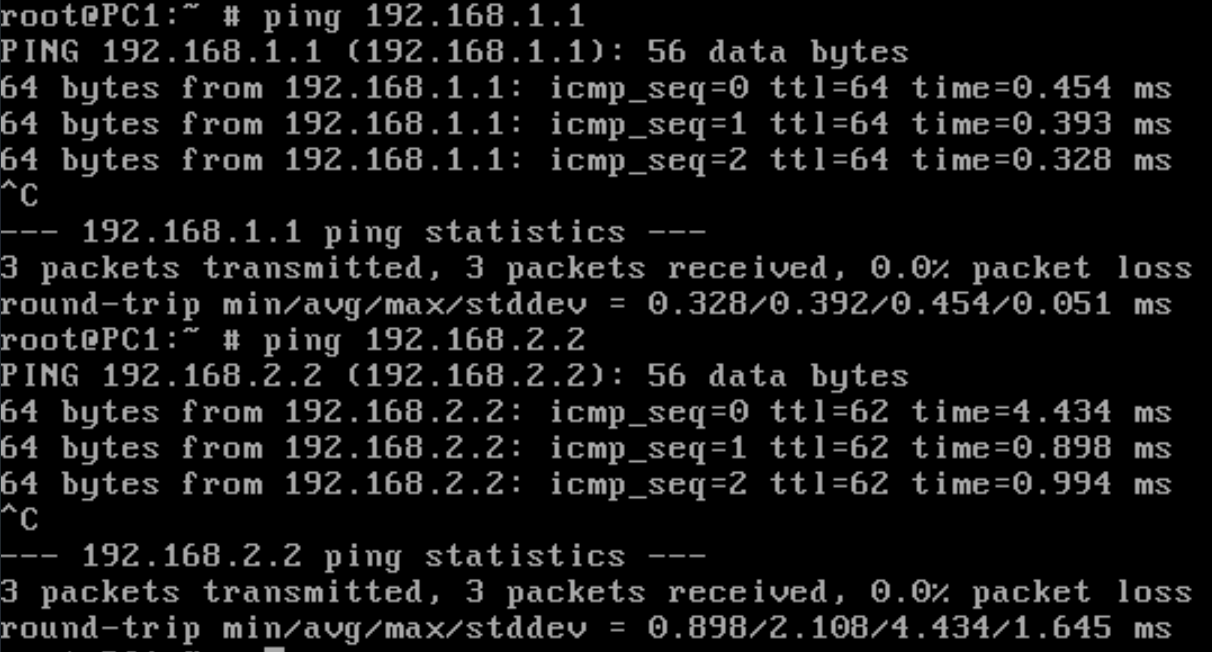
***4.15***

***4.16***

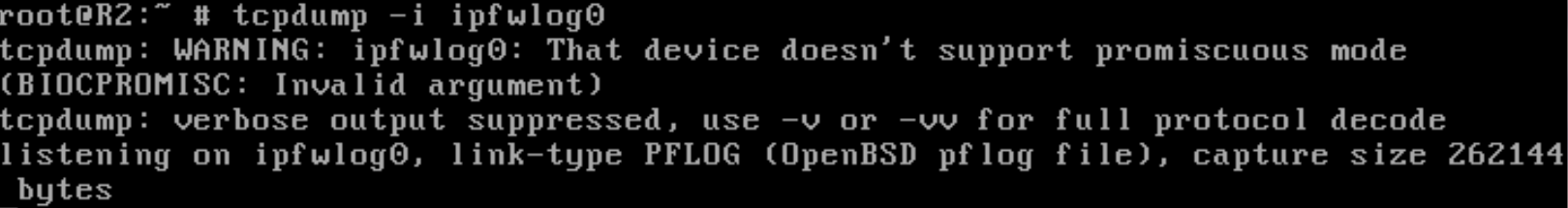
***4.17***

***4.18***

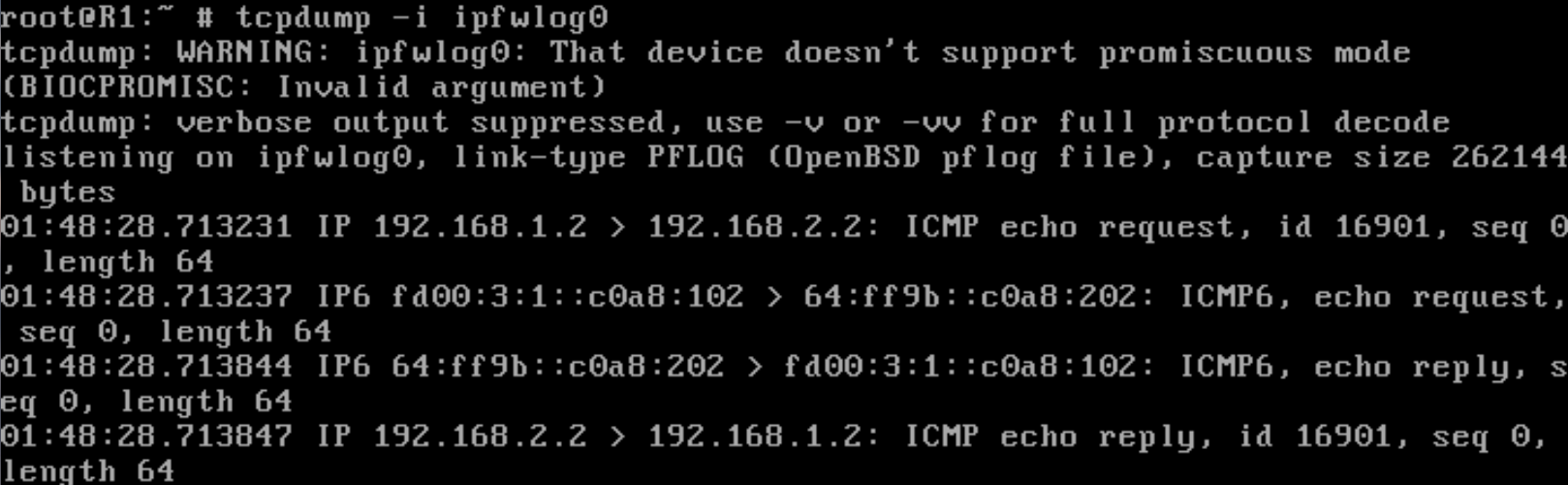
***4.19***

Πετυχαίνουν αμφότερα.

***4.20***

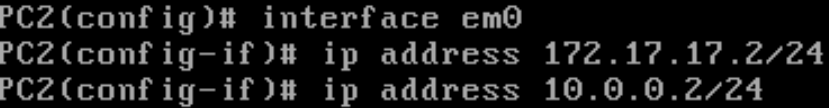
***4.21***

***4.22***

Παρατηρούμε τα παρακάτω πακέτα στα R1 και R2 αντίστοιχα.

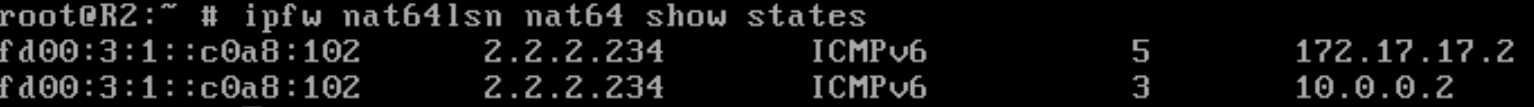


Βλέπουμε πως αρχικά το PC1 στέλνει ένα ICMP echo request με προορισμό το PC2, των οποίων οι IPv4 διευθύνσεις πηγής και προορισμού μετατρέπονται σε IPv6 καθώς διέρχονται από το WAN1, μέχρι να μετατραπούν ξανά σε IPv4 στο LAN2. Κατά την αντίστροφη πορεία έχουμε ξανά μετατροπή από IPv4 στο LAN2 σε IPv6 στο WAN1 και ξανά σε IPv4 στο LAN1.

***4.23***

***4.24***

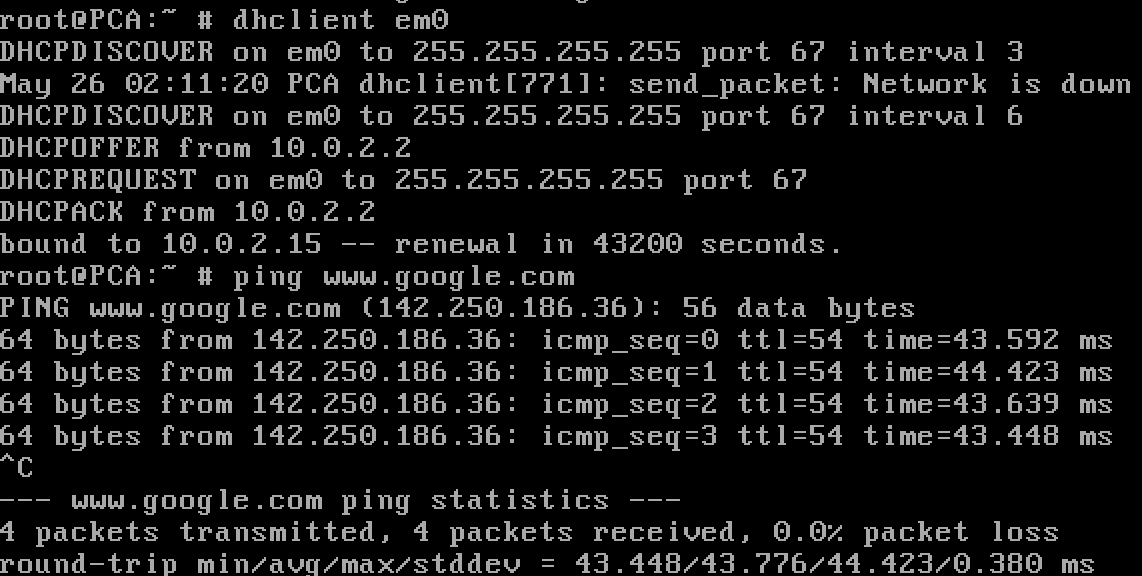
Ναι.

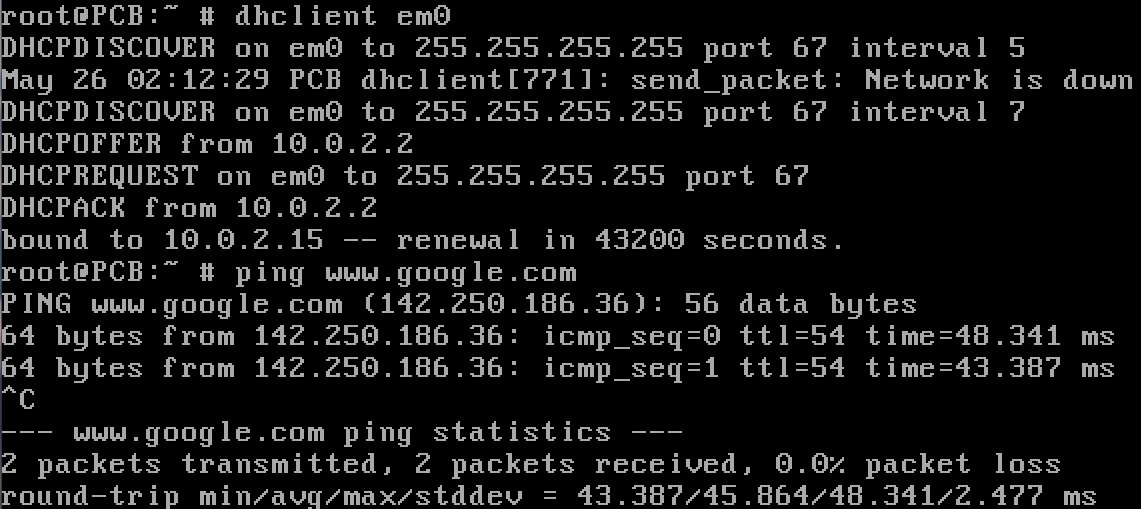
***4.25***

***4.26***

Παρατηρούμε πως ο χρόνος αναπαριστά τον χρόνο που απομένει μέχρι να διαγραφεί η εγγραφή από τον πίνακα και ανέρχεται σε περίπου 1 λεπτό.

**Άσκηση 5: Μηχανισμός μετάβασης Teredo**

***5.1***



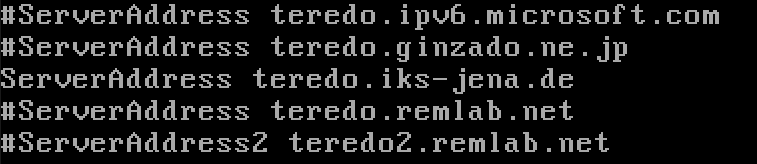
***5.2***

Εκτελούμε στα PC1, PC2 “**pkg install miredo**”.

***5.3***

Εκτελούμε στα PC1, PC2 “**sysrc miredo\_enable=”YES**””.

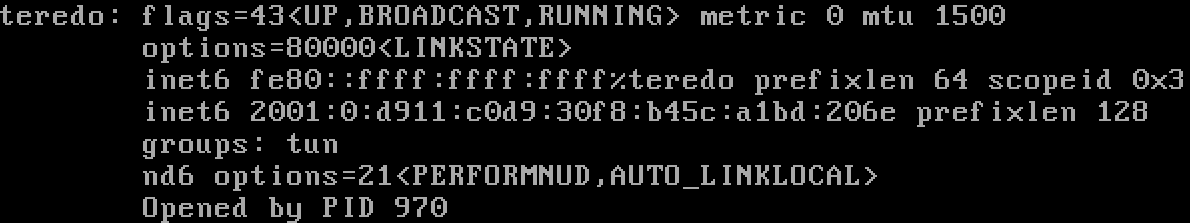
***5.4***

Εκτελούμε στα PC1, PC2 “**vi /usr/local/etc/miredo/miredo.conf**” και κάνουμε τις αλλαγές:

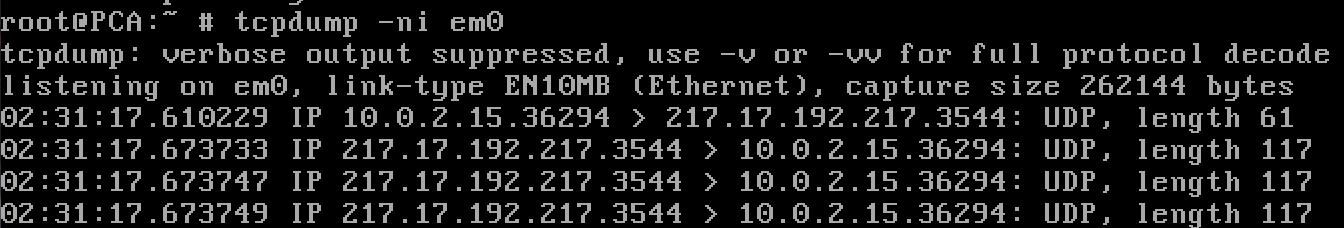
Ύστερα εκτελούμε και στα δύο PC “**service miredo start**”.

***5.5***

Βλέπουμε την επιπλέον διεπαφή teredo με IPv6 address:

2001:0:d911:c0d9:30f8:b45c:a1bd:206e/128.

***5.6***

Εκτελούμε στο PC1 “**tcpdump -ni em0**”.

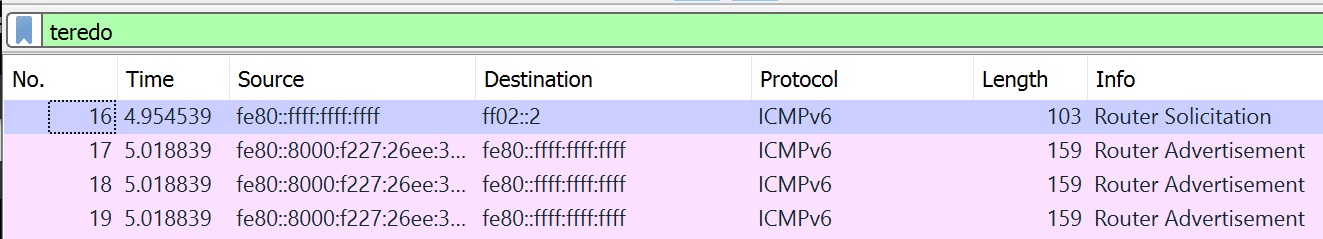
***5.7***

Είναι 217.17.192.217.

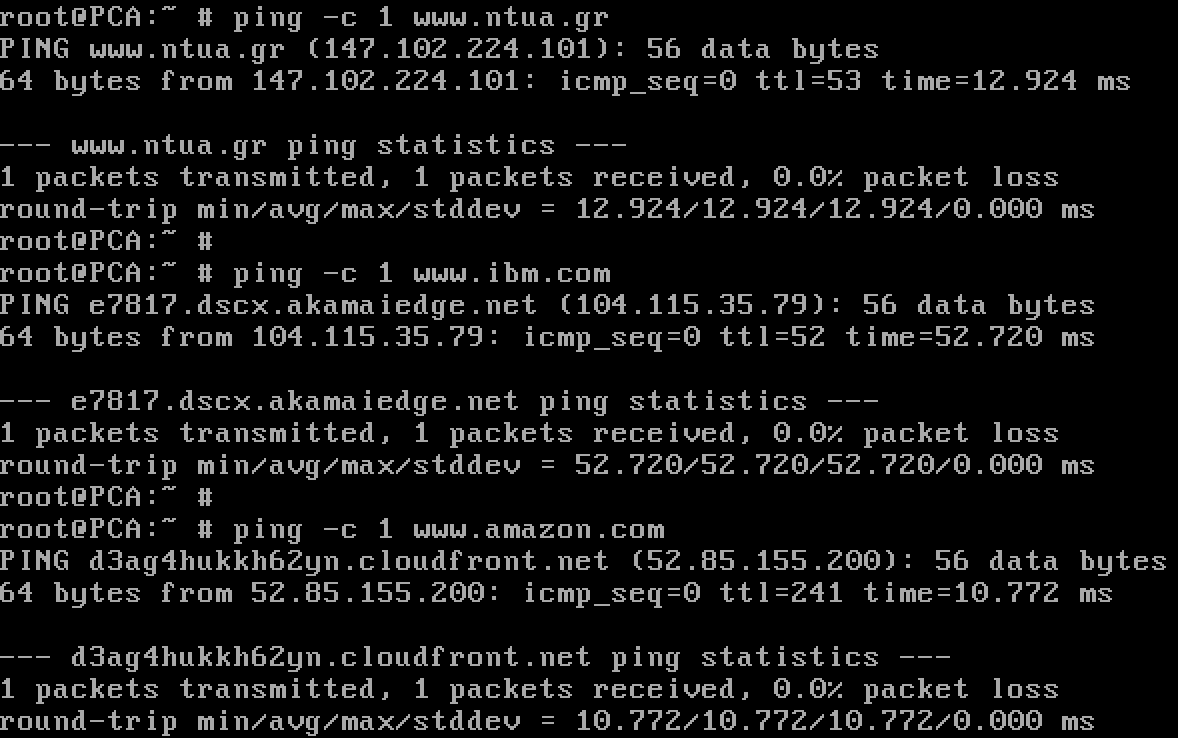
***5.8***

Χρησιμοποιείται το UDP στο Transport Layer και η θύρα 3544.

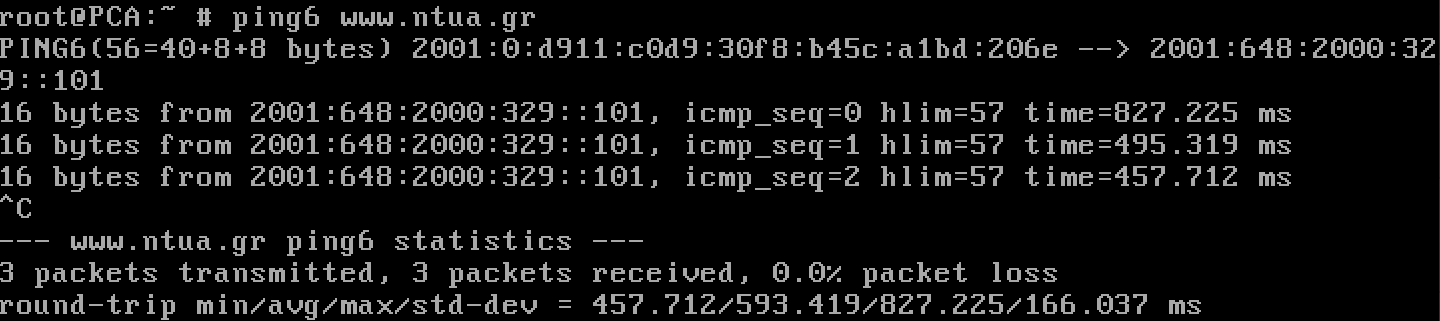
***5.9***

Παρατηρούμε πακέτα ICMPv6.

***5.10***

Ναι σε όλα.

***5.11***

Κάνουμε “**ping6 www.ntua.gr**”.

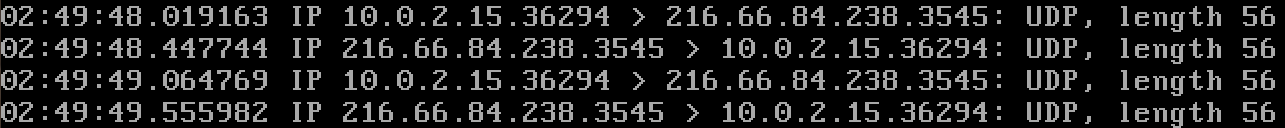
***5.12***

Παρατηρούμε το εξής:

***5.13***

Όχι.

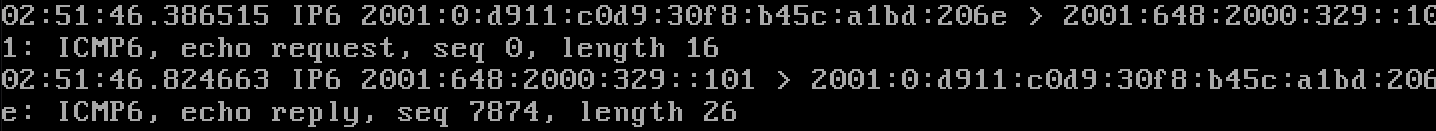
***5.14***

Βλέπουμε το πρωτόκολλο UDP και την θύρα 3545.

***5.15***

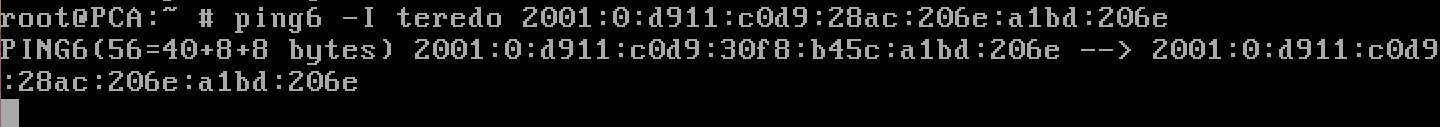
Εκτελούμε στο PC1 “**tcpdump -ni teredo**”.

***5.16***

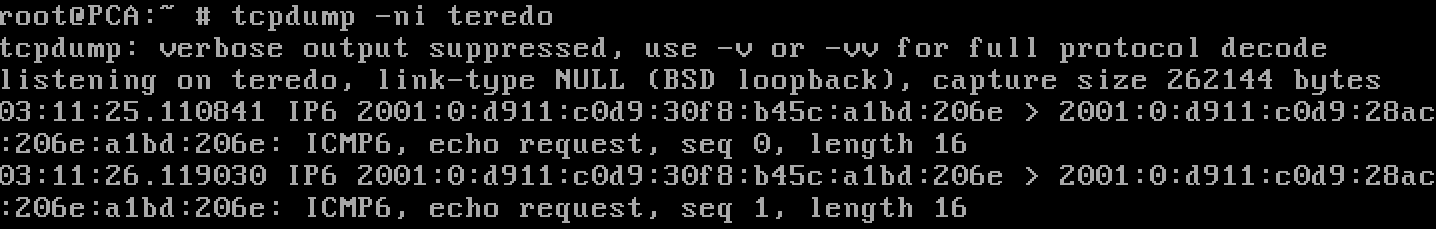
Βλέπουμε ICMPv6 Echo requests/replies.

***5.17***

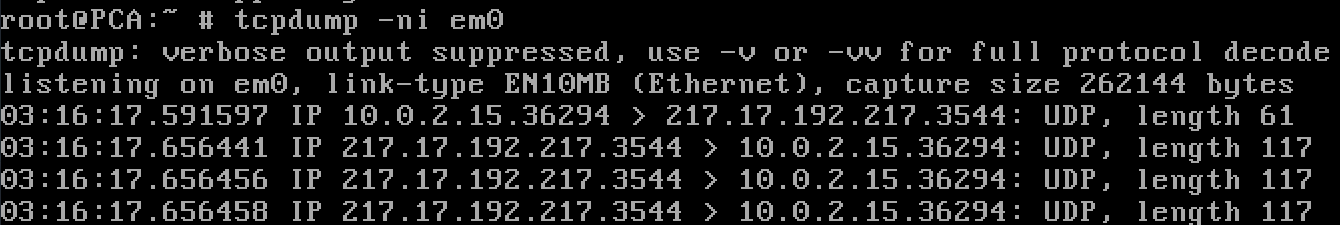
Όχι, δεν μπορούμε.



***5.18***

Ναι, παράγονται ICMPv6 Echo requests.

***5.19***

Παράγονται και στέλνονται στην 217.17.192.217, όπως είδαμε και πριν.

***5.20***

Κάνοντας ping στο www.quad9.net βλέπουμε πως κάνουμε ping στη θύρα 3544 της 217.17.192.217, ενώ όταν κάνω στο [www.f5.com](http://www.f5.com) βλέπουμε ότι επιλέγεται ο teredo relay στη θύρα 3545 της 216.66.86.186.

