

Όνοματεπώνυμο: Χαράλαμπος Καμπουγέρης

Όνομα PC/ΛΣ: DESKTOP-N90CRE0

Ομάδα: 1, Τρίτη 10:45-13:30, Αιθ.Α4

Ημερομηνία: 27/03/2024

## Εργαστηριακή Άσκηση 5

### Στατική δρομολόγηση

#### Άσκηση 1

**1.1** Εκτελέσαμε τις εξής εντολές:

- PC1: "ifconfig em0 192.168.1.2/24"
- PC2: "ifconfig em0 192.168.2.2/24"

**1.2** sysrc ifconfig\_em0="inet 192.168.1.1 netmask 255.255.255.0" και  
sysrc ifconfig\_em1="inet 192.168.2.1 netmask 255.255.255.0"

Προφανώς για να ισχύσουν οι όποιες τροποποιήσεις κάνετε, θα πρέπει να επανεκκινήσουμε την εμπλεκόμενη υπηρεσία.

**1.3** gateway\_enable="YES"

**1.4** service netif restart && service routing restart

**1.5** Εκτελούμε στο PC1 "route add -net 192.168.2.0/24 192.168.1.1".

**1.6** Για το υποδίκτυο 192.168.2.0/24 βλέπουμε τις σημαίες UGS, οπότε σημαίνει πως η διαδρομή είναι ενεργή (U), ο προορισμός είναι πύλη, η οποία και θα αποφασίσει για την περαιτέρω προώθηση των πακέτων (G) και τέλος η διαδρομή ορίστηκε στατικά (S).

**1.7** Ενώ στάλθηκε το ping, δε λαμβάνουμε απάντηση.

**1.8** Στο LAN1 παρατηρούμε πως το PC1 στέλνει τα ICMP Echo requests του στη διεπαφή em0 του R1, ενώ στο LAN2 το R1 στέλνει μέσω της em1 τα αιτήματα στο PC2. Επομένως, ενώ ο PC2 λαμβάνει κανονικά τα requests του PC1, αδυνατεί να απαντήσει, καθώς δε ξέρει προς τα που πρέπει να προωθήσει τα replies.

**1.9** Εκτελούμε στο PC2 "route add -net 192.168.1.0/24 192.168.2.1".

**1.10** Ναι, πλέον επικοινωνούν κανονικά.

**1.11** Ο πίνακας δρομολόγησης του R1, όπως βλέπουμε στην εικόνα έχει ήδη την απαραίτητη πληροφορία για δρομολόγηση στα LAN1 (192.168.1.0/24) και LAN2 (192.168.2.0/24), οπότε και δεν απαιτείται κάποια επιπλέον ρύθμιση.

```
Internet:
Destination      Gateway          Flags           Netif  Expire
127.0.0.1         link#3           UHS             lo0
192.168.1.0/24    link#1           U               em0
192.168.1.1       link#1           UHS             lo0
192.168.2.0/24    link#2           U               em1
192.168.2.1       link#2           UHS             lo0
```

## Άσκηση 2

**2.1** Εκτελούμε στο PC1 την εντολή “route del 192.168.2.0/24”

**2.2** Στο PC1 “ifconfig em0 192.168.1.2/20”

**2.3** Το PC1 βρίσκεται στο υποδίκτυο 192.168.0.0. Εάν εφαρμόσουμε τη μάσκα του υποδικτύου του στις διευθύνσεις των PC2, PC3 βλέπουμε πως το PC1 τα αντιλαμβάνεται σαν να ανήκουν στο ίδιο υποδίκτυο.

**2.4** Το ping δεν είναι επιτυχές.

**2.5** Πλέον το ping είναι επιτυχές, καθώς ο δρομολογητής λειτουργεί ως proxy, επομένως απαντάει με τη δική του MAC στα ARP requests του PC1, δεδομένου ότι το PC2 βρίσκεται σε υποδίκτυο στο οποίο ο R1 ξέρει πώς να δρομολογήσει πακέτα για εκεί.

**2.6** Αποτυγχάνει, καθώς για τον PC3, το PC1 φαίνεται πως είναι σε άλλο υποδίκτυο, οπότε δεδομένου ότι δεν έχει προκαθορισμένη πύλη ή κάποια πύλη για το υποδίκτυο του PC1, απλά απορρίπτει τα πακέτα που λαμβάνει χωρίς να απαντά.

**2.7** Εκτελούμε στο PC3 την εντολή “route add -net 192.168.1.0/24 192.168.2.1”.

**2.8** arp -a -d

**2.9** Στο R1 εκτελούμε σε μία κονσόλα “tcpdump -ei em0” και σε μία δεύτερη “tcpdump -ei em1”.

**2.10** Βλέπουμε πως το R1 απαντάει με τη MAC του em0 του, παρόλο που το ping έχει ως προορισμό το PC3.

**2.11** Προς τη MAC του R1

**2.12** Από τη MAC του R1 στο em1 (LAN2)

### 2.13

- Ο PC1 κάνει broadcast ARP request για να μάθει την MAC address της διεύθυνσης 192.168.2.3
- Το R1 απαντάει δίνοντάς του τη MAC της διεπαφής em0 ως MAC της 192.168.2.3, αφού το έχουμε δηλώσει ως proxy
- Το PC1 στέλνει στο em0 του R1 το ICMP echo request
- Το R1, μέσω της em1 κάνει broadcast ένα ARP request με σκοπό να μάθει την MAC της 192.168.2.3
- Το PC3 απαντάει στο παραπάνω broadcast με τη MAC διεύθυνσή του
- Το R1 προωθεί στο PC3 το ICMP echo request μέσω της em1
- Το PC3 απαντάει στην em1 του R1 με ICMP echo reply, με τελικό αποδέκτη τη διεύθυνση 192.168.1.2
- Το R1 κάνει broadcast ένα ARP Request μέσω της em0, ώστε να μάθει την MAC της 192.168.1.2
- Το PC1 απαντάει με τη MAC διεύθυνσή του στο R1
- Το R1 προωθεί το ICMP echo reply στο PC1

**2.14** Το ping θα επιτυγχάνει όσο το PC1 νομίζει πως το PC3 είναι στο ίδιο υποδίκτυο με αυτό. Επομένως, το μέγιστο μήκος προθέματος είναι 22, καθώς αν βάλουμε 23, το PC1 αντιλαμβάνεται πως το PC3 ανήκει στο 192.168.2.0/23, ενώ το ίδιο το PC1 ανήκει στο 192.168.0.0/23, άρα από τα 23 bits και μετά απαιτείται δρομολόγηση, για την οποία δεν έχουμε ορίσει κάποια πύλη στο PC1, οπότε και το ping θα αποτυγχάνει.

**2.15** Στο PC1: "ifconfig em0 192.168.1.2/23".

**2.16** Στο PC1: "route add -net 192.168.2.0/24 -interface em0"

**2.17** Με "netstat -rn" βλέπουμε πως ως πύλη για το δίκτυο 192.168.2.0/24 εμφανίζεται η MAC address της διεπαφής em0.

**2.18** Πλέον το ping επιτυγχάνει, καθώς το ταίριασμα μεγαλύτερου προθέματος γίνεται με το υποδίκτυο 192.168.2.0/24, οπότε και το em0 κάνει τα κατάλληλα ARP requests, ώστε να στείλει τα ICMP πακέτα και λαμβάνει απαντήσεις από το proxy ARP, δηλαδή το R1, το οποίο απαντάει σα να ήταν το PC3.

**2.19** sysctl net.link.ether.inet.proxyall=0

**2.20** route change -net 192.168.2.0/24 192.168.1.1

**2.21** Στο PC1: "ifconfig em0 192.168.1.2/24"

**2.22** Η διαδρομή διαγράφηκε.

**2.23** "route add -net 192.168.2.0/24 192.168.1.1

**2.24** ifconfig em0 delete

### Άσκηση 3

**1.3** sysrc ifconfig\_em1="inet 172.17.17.1 netmask 255.255.255.252"

service netif restart

**1.4** sysrc ifconfig\_em0="inet 172.17.17.2 netmask 255.255.255.252" και  
sysrc ifconfig\_em1="inet 192.168.2.1 netmask 255.255.255.0"

service netif restart

**3.3** gateway\_enable="YES" και service routing restart

**3.4**

```
root@PC:~ # ping -c 1 192.168.2.2
PING 192.168.2.2 (192.168.2.2): 56 data bytes
92 bytes from 192.168.1.1: Destination Host Unreachable
  vr  HL  TOS  Len   ID  Flg  off TTL  Pro  cks           Src           Dst
   4   5   00  0054 08ac   0 0000  3f   01 eea8 192.168.1.2 192.168.2.2

^C
--- 192.168.2.2 ping statistics ---
1 packets transmitted, 0 packets received, 100.0% packet loss
```

**3.5**

```
root@PC:~ # 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
01:33:52.975418 IP 192.168.1.2 > 192.168.2.2: ICMP echo request, id 39171, seq 0
, length 64
01:33:52.975442 IP 192.168.1.1 > 192.168.1.2: ICMP host 192.168.2.2 unreachable,
length 92
```

Παρατηρούμε πως στο WAN1 δε καταγράφεται κανένα πακέτο. Ο λόγος είναι πως στον πίνακα δρομολόγησης του R1 δεν υπάρχει default gateway, αλλά ούτε και εγγραφή προς το υποδίκτυο 192.168.2.0/24., με αποτέλεσμα να επιστρέφεται το μήνυμα λάθους .

**3.6**

```
root@PC:~ # traceroute 192.168.2.2
traceroute to 192.168.2.2 (192.168.2.2), 64 hops max, 40 byte packets
 1  192.168.1.1 (192.168.1.1)  0.268 ms  0.191 ms  0.175 ms
 2  192.168.1.1 (192.168.1.1)  0.206 ms !H  0.169 ms !H  0.156 ms !H
```

Το “!H” αναφέρεται στο Host Unreachable.

**3.7** Εκτελούμε στο R1 την εντολή “route add -net 192.168.2.0/24 172.17.17.2”.

**3.8** Πλέον, το ping του PC1 στο PC2 επιτυγχάνει ως Request, ωστόσο δε λαμβάνουμε πίσω το Reply, καθώς όταν στέλνει το Reply το PC2, αυτό αδυνατεί να προωθηθεί από το R2, επομένως στέλνεται ένα “ICMP host 192.168.1.2 unreachable” από το R2 στο PC2.

### 3.9

```
root@PC:~ # tcpdump -i em1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on em1, link-type EN10MB (Ethernet), capture size 262144 bytes
01:41:50.789450 IP 192.168.1.2 > 192.168.2.2: ICMP echo request, id 45315, seq 0,
length 64
01:41:50.789685 IP 192.168.2.2 > 192.168.1.2: ICMP echo reply, id 45315, seq 0,
length 64
01:41:50.789694 IP 192.168.2.1 > 192.168.2.2: ICMP host 192.168.1.2 unreachable,
length 92
```

- IP 192.168.1.2 > 192.168.2.2: ICMP echo request, το οποίο είναι το πακέτο του PC1 που το R2 προωθεί στο PC2
- IP 192.168.2.2 > 192.168.1.2: ICMP echo reply, το οποίο είναι το πακέτο που το PC2 στέλνει στο R2 με τελικό προορισμό το PC1
- IP 192.168.2.1 > 192.168.2.2: ICMP host 192.168.1.2 unreachable, το οποίο είναι η απάντηση που το R1 στέλνει στο PC2, ενημερώνοντας το πως δε μπορεί να προωθήσει το προηγούμενο reply

### 3.10

```
root@PC:~ # tcpdump -i em1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on em1, link-type EN10MB (Ethernet), capture size 262144 bytes
01:41:51.437523 IP 192.168.1.2 > 192.168.2.2: ICMP echo request, id 45315, seq 0,
length 64
01:46:26.597985 IP 192.168.1.2.33726 > 192.168.2.2.33438: UDP, length 12
01:46:31.653317 IP 192.168.1.2.33726 > 192.168.2.2.33439: UDP, length 12
01:46:36.752455 IP 192.168.1.2.33726 > 192.168.2.2.33440: UDP, length 12
01:46:41.778609 IP 192.168.1.2.33726 > 192.168.2.2.33441: UDP, length 12
```

Στο WAN1 δε παρατηρούμε πακέτα ICMP, παρά μόνο UDP, με αποστολέα το 192.168.1.2.33726 και παραλήπτη τη διεύθυνση 192.168.2.2 με διαφορετική κάθε φορά θύρα προορισμού προκειμένου ο κόμβος-παραλήπτης να μην επεξεργαστεί τα UDP packets. Καταγράφουμε τα εν λόγω πακέτα, καθώς αυτά αποστέλλονται μέσω του μέχρι να ληφθεί απάντηση ICMP time exceeded.

### 3.11

```
reachable, length 48
01:52:01.865788 IP 192.168.1.2.33732 > 192.168.2.2.33446: UDP, length 12
01:52:01.866137 IP 192.168.2.2 > 192.168.1.2: ICMP 192.168.2.2 udp port 33446 un-
reachable, length 48
01:52:06.914906 IP 192.168.1.2.33732 > 192.168.2.2.33447: UDP, length 12
01:52:06.915106 IP 192.168.2.2 > 192.168.1.2: ICMP 192.168.2.2 udp port 33447 un-
reachable, length 48
```

Στο LAN2 βλέπουμε να προωθούνται τα UDP πακέτα από το R2 στο PC2, ενώ επιπλέον βλέπουμε ως απάντηση από το PC2 (192.168.2.2) στο R2 (192.168.1.2) μηνύματα ICMP 192.168.2.2 udp port XXXXX unreachable, όπου XXXXX η εκάστοτε θύρα προορισμού.

**3.12** Είδαμε πως το PC2 αποκρίνεται στο R2 λέγοντας πως ήταν unreachable το port του μηνύματος που έλαβε. Δε παράγονται ICMP host unreachable μηνύματα ως απόκριση στα ICMP που παράγει το PC2, καθώς σε αυτή την περίπτωση θα προκαλούνταν loop στο σύστημα.

**3.13** Εκτελούμε στο R2 “route add -net 192.168.1.0/24 172.17.17.1”.

### 3.14

```
root@PC:~ # traceroute 192.168.2.2
traceroute to 192.168.2.2 (192.168.2.2), 64 hops max, 40 byte packets
 1  192.168.1.1 (192.168.1.1)  0.461 ms  0.170 ms  0.158 ms
 2  172.17.17.2 (172.17.17.2)  1.936 ms  0.388 ms  0.283 ms
 3  192.168.2.2 (192.168.2.2)  0.726 ms  0.492 ms  0.441 ms
```

Μπορούμε να κάνουμε traceroute. Στο WAN2 παράγονται μηνύματα τύπου ICMP time exceeded in-transit (172.17.17.2 > 192.168.1.2), ενώ επιπλέον καταγράφονται μηνύματα ICMP 192.168.2.2 udp port 33443 unreachable (192.168.2.2 > 192.168.1.2). Τα ICMP time exceeded μηνύματα που παρήχθησαν οφείλονται στο γεγονός ότι στη διεπαφή 172.17.17.2 του R2 μηδενίστηκε το TTL (TTL = 2) της δεύτερης τριάδας πακέτων που απεστάλησαν από το traceroute του PC1.

**3.15** Λαμβάνουμε ως απάντηση “no route to host”, πράγμα που οφείλεται στο γεγονός ότι ο πίνακας δρομολόγησης του PC2 δε περιλαμβάνει εγγραφές ούτε για το υποδίκτυο της διεύθυνσης 172.17.17.1, αλλά ούτε και έχει default gateway, ώστε το πακέτο να δρομολογηθεί από εκεί.

**3.16** route del 192.168.1.0/24

**3.17** Στο PC2 εκτελούμε την εντολή “route add default 192.168.2.1”.

**3.18** Το ping επιτυγχάνει.

**3.19** Στο πρώτο ping το PC2 αδυνατούσε να στείλει το πακέτο του, καθώς δεν είχε κάποια εγγραφή για το πού έπρεπε να το στείλει. Αφού ορίσαμε προκαθορισμένη πύλη, πλέον το πακέτο που στέλνει το PC2 δρομολογείται σε αυτή, και από εκεί, εφόσον υπάρχει εγγραφή για το υποδίκτυο του τελικού προορισμού πηγαίνει σε αυτόν.

## Άσκηση 4

**4.1** “ifconfig em0 192.168.2.3/24” και “route add -net 192.168.1.0/24 192.168.2.1”.

**4.2** sysrc ifconfig\_em2="inet 172.17.17.5 netmask 255.255.255.252"

service netif restart

**4.3** sysrc ifconfig\_em2="inet 172.17.17.9 netmask 255.255.255.252"

service netif restart

**4.4** sysrc ifconfig\_em0="inet 172.17.17.6 netmask 255.255.255.252"

sysrc ifconfig\_em1="inet 172.17.17.10 netmask 255.255.255.252"

service netif restart

**4.5** gateway\_enable="YES" και service routing restart

**4.6** "route add -net 192.168.2.0/24 172.17.17.2"

**4.7** "route add -net 192.168.1.0/24 172.17.17.1"

**4.8** "route add -net 192.168.1.0/24 172.17.17.5" και "route add -net 192.168.2.0/24 172.17.17.9"

**4.9** Εκτελούμε στο R1 "route add -host 192.168.2.3 172.17.17.6" και βλέπουμε στα αποτελέσματα της εντολής "netstat -r" τις σημαίες UGHS για τη συγκεκριμένη εγγραφή, εκ των οποίων η H δηλώνει πως αναφέρεται σε συγκεκριμένο host.

**4.10** Βλέπουμε συνολικά 3 βήματα:

```
root@PC:~ # traceroute 192.168.2.2
traceroute to 192.168.2.2 (192.168.2.2), 64 hops max, 40 byte packets
 1  192.168.1.1 (192.168.1.1)  0.467 ms  1.298 ms  0.235 ms
 2  172.17.17.2 (172.17.17.2)  0.464 ms  2.312 ms  0.602 ms
 3  192.168.2.2 (192.168.2.2)  1.083 ms  0.522 ms  0.439 ms
```

**4.11** Βλέπουμε πως το TTL που λάβαμε ως απάντηση έχει τιμή 62, επομένως μεσολάβησαν 2 ενδιάμεσοι κόμβοι, όπως και πριν.

**4.12** Βλέπουμε συνολικά 4 βήματα:

```
root@PC:~ # traceroute 192.168.2.3
traceroute to 192.168.2.3 (192.168.2.3), 64 hops max, 40 byte packets
 1  192.168.1.1 (192.168.1.1)  0.429 ms  0.195 ms  0.187 ms
 2  172.17.17.6 (172.17.17.6)  0.371 ms  0.335 ms  0.370 ms
 3  172.17.17.2 (172.17.17.2)  1.861 ms  0.502 ms  0.429 ms
 4  192.168.2.3 (192.168.2.3)  0.644 ms  0.697 ms  0.733 ms
```

**4.13** Λαμβάνουμε απάντηση με TTL = 62, επομένως, βλέπουμε 2 ενδιάμεσα βήματα.

**4.14** Το ICMP echo request ακολουθεί τη διαδρομή PC1 → R1 → R3 → R2 → PC3.

**4.15** Αντιθέτως, το ICMP echo reply ακολουθεί τη διαδρομή PC3 → R2 → R1 → PC1, δεδομένου πως είχαμε ορίσει στατική εγγραφή στον R2 ώστε να προωθούνται πακέτα προς το LAN1 μέσω του R1, ενώ είχαμε ορίσει επίσης στατική εγγραφή στο R1, έτσι ώστε πακέτα προς το PC3 να διέρχονται από το R3.

**4.16** Εκτελούμε στο R2 "tcpdump -i em1"

**4.17** Όχι, δε καταγράφουμε τίποτα, επομένως ούτε έφτασε κάτι, ούτε παρήχθησαν UDP στο PC2.

**4.18** Καταγράφουμε τα παρακάτω πακέτα, επομένως φτάνουν τα πακέτα από το PC1 στο PC3 και στη συνέχεια παράγονται πακέτα "ICMP 192.168.2.3 udp port XXXXX unreachable".

```

03:14:27.088289 IP 192.168.1.2.34300 > 192.168.2.3.33450: UDP, length 12
03:14:27.089996 IP 192.168.2.3 > 192.168.1.2: ICMP 192.168.2.3 udp port 33450 unreachable, length 48
03:14:32.188668 IP 192.168.1.2.34300 > 192.168.2.3.33451: UDP, length 12
03:14:32.188971 IP 192.168.2.3 > 192.168.1.2: ICMP 192.168.2.3 udp port 33451 unreachable, length 48
03:14:37.289404 IP 192.168.1.2.34300 > 192.168.2.3.33452: UDP, length 12
03:14:37.291123 IP 192.168.2.3 > 192.168.1.2: ICMP 192.168.2.3 udp port 33452 unreachable, length 48
03:14:42.389927 IP 192.168.1.2.34300 > 192.168.2.3.33453: UDP, length 12
03:14:42.390428 IP 192.168.2.3 > 192.168.1.2: ICMP 192.168.2.3 udp port 33453 unreachable, length 48
03:14:47.459991 IP 192.168.1.2.34300 > 192.168.2.3.33454: UDP, length 12
03:14:47.460230 IP 192.168.2.3 > 192.168.1.2: ICMP 192.168.2.3 udp port 33454 unreachable, length 48
03:14:52.490026 IP 192.168.1.2.34300 > 192.168.2.3.33455: UDP, length 12
03:14:52.490309 IP 192.168.2.3 > 192.168.1.2: ICMP 192.168.2.3 udp port 33455 unreachable, length 48
03:14:57.539362 IP 192.168.1.2.34300 > 192.168.2.3.33456: UDP, length 12
03:14:57.539622 IP 192.168.2.3 > 192.168.1.2: ICMP 192.168.2.3 udp port 33456 unreachable, length 48

```

**4.19** Ναι, ολοκληρώνεται

**4.20** route change -net 192.168.2.0/24 172.17.17.6 στον R1

route change -net 192.168.1.0/24 172.17.17.10 στον R2

**4.21**

```

root@PC:~ # route show 192.168.2.2
route to: 192.168.2.2
destination: 192.168.2.0
mask: 255.255.255.0
gateway: 172.17.17.6
fib: 0
interface: em2
flags: <UP,GATEWAY,DONE,STATIC>
rcvpipe sendpipe ssthresh rtt,msec mtu weight expire
0 0 0 0 1500 1 0
root@PC:~ # route show 192.168.2.3
route to: 192.168.2.3
destination: 192.168.2.3
gateway: 172.17.17.6
fib: 0
interface: em2
flags: <UP,GATEWAY,HOST,DONE,STATIC>
rcvpipe sendpipe ssthresh rtt,msec mtu weight expire
0 0 0 0 1500 1 0
root@PC:~ #

```

Στο PC2 ως destination είναι η διεύθυνση του υποδικτύου, ενώ στο PC3 η ίδια η διεύθυνση

**4.22** Μεταξύ των 2 τελευταίων εγγραφών, επιλέγεται η τελευταία, καθώς έχουμε ταίριασμα μήκους 32 bits.



## Routing tables

Internet:

Destination	Gateway	Flags	Netif	Expire
127.0.0.1	link#4	UH	lo0	
172.17.17.0/30	link#2	U	em1	
172.17.17.1	link#2	UHS	lo0	
172.17.17.4/30	link#3	U	em2	
172.17.17.5	link#3	UHS	lo0	
192.168.1.0/24	link#1	U	em0	
192.168.1.1	link#1	UHS	lo0	
192.168.2.0/24	172.17.17.6	UGS	em2	
192.168.2.3	172.17.17.6	UGHS	em2	

**4.23** Στο R3: route change -net 192.168.2.0/24 172.17.17.5

**4.24** Λαμβάνουμε σφάλμα "Time to live exceeded".

```
root@PC:~ # ping -c 1 192.168.2.2
PING 192.168.2.2 (192.168.2.2): 56 data bytes
92 bytes from 172.17.17.6: Time to live exceeded
^C
--- 192.168.2.2 ping statistics ---
1 packets transmitted, 0 packets received, 100.0% packet loss
```

**4.25** Από τη διεπαφή em0 του R3, με διεύθυνση 172.17.17.6

**4.26** "tcpdump -i em0 -e 'icmp[icmptype] == icmp-echo'".

**4.27** 64 πακέτα ICMP Echo request

**4.28** 32 πακέτα έχουν πηγή τον R1 και 32 τον R2

**4.29** tcpdump -i em2 -e 'icmp[icmptype]==icmp-echo' στο R1  
tcpdump -i em0 -e 'icmp[icmptype]==11' στον R3

**4.30** Εμφανίζονται 64 βήματα, ενώ η διαδρομή που καταγράφεται είναι: PC1 → R1 (192.168.1.1) → R3 (172.17.17.6) → R1 (192.168.1.1) → R3 (172.17.17.6) → R1 (192.168.1.1) → R3 (172.17.17.6) ... → R1 (192.168.1.1) → R3 (172.17.17.6).

**4.31** Στάλθηκαν 64 μηνύματα ICMP echo request από το PC1.

Στο πρώτο ICMP echo request του PC1, αναμένουμε πως δε καταγράφηκε τίποτα στο WAN2, καθώς το TTL ήταν 1, οπότε και απάντησε το R1 αμέσως. Για τα υπόλοιπα ICMP echo requests του PC1:

- 2<sup>ο</sup> ICMP echo request: Καταγράφεται 1 στο WAN2
- 3<sup>ο</sup> ICMP echo request: Καταγράφονται 2 στο WAN2
- 4<sup>ο</sup> ICMP echo request: Καταγράφονται 3 στο WAN2
- ν-οστό ICMP echo request: Καταγράφονται ν-1 στο WAN2

Συνεπώς, εφόσον έχουμε συνολικά 64 requests, ψάχνουμε το άθροισμα:  $0 + 1 + 2 + 3 + \dots + 63 = 63 \cdot 64 / 2 = 2016$ , όσα και τα πακέτα που καταγράψαμε.

**4.32** Εμφανίστηκαν 32 μηνύματα ICMP time exceeded. Εξετάζοντας το αποτέλεσμα, σκεφτόμαστε αντίστοιχα με πριν. Το πρώτο request θα λάβει ως απάντηση από ICMP time exceeded από το R1, δεδομένου ότι έχει TTL = 1, επομένως και δε θα καταγραφεί στο WAN2. Το δεύτερο ωστόσο, με TTL = 2, θα λάβει τέτοια απάντηση από το R3, επομένως θα καταγραφεί στο WAN2. Συνεχίζοντας αυτή τη συλλογιστική, βλέπουμε πως θα καταγραφούν μόνο ICMP time exceeded από πακέτα που είχαν ζυγό TTL έως και 64, άρα για TTL = 2, 4, 6, 8, ... 64, συνεπώς 32 πακέτα.

## Άσκηση 5

**5.1** Από το μπλοκ 172.17.17.0/24 μπορούμε να φτιάξουμε 2 υποδίκτυα που να χωράνε 126 hosts, τα 172.17.17.0/25 και 172.17.17.128/25. Ωστόσο, παρατηρούμε πως το υποδίκτυο 172.17.17.128/30 είναι δεσμευμένο από τα WANs. Επομένως, αναθέτουμε στο LAN1 το 172.17.17.0/25 προκειμένου να αποφύγουμε ενδεχόμενες μελλοντικές συγκρούσεις στα ταιριάσματα μήκους. Μας απομένει το 172.17.17.128/25.

**5.2** Από το προηγούμενο μπλοκ μας απομένει το 172.17.17.128/25, το οποίο διαθέτει 2 υποδίκτυα χωρητικότητας 62 κόμβων έκαστο, το 172.17.17.192/26 και το 172.17.17.128/26. Για τους λόγους που αναφέραμε και προηγουμένως, αναθέτουμε στο LAN2 το 172.17.17.192/26.

**5.3** Αντίστοιχα, το 172.17.17.128/26 που μας έμεινε μπορεί να σπάσει σε 172.17.17.128/27 και 172.17.17.160/27 με 30 υπολογιστές έκαστο. Αναθέτουμε το 172.17.17.160/27 στο LAN3

**5.4** sysrc ifconfig\_em1="inet 172.17.17.129 netmask 255.255.255.252"  
sysrc ifconfig\_em2="inet 172.17.17.133 netmask 255.255.255.252"

service netif restart

**5.5** Εκτελούμε "ifconfig em0 172.17.17.1/25" στο PC1 και "ifconfig em0 172.17.17.126/25" στο R1.

**5.6** sysrc ifconfig\_em0="inet 172.17.17.130 netmask 255.255.255.252"  
sysrc ifconfig\_em2="inet 172.17.17.138 netmask 255.255.255.252"

service netif restart

**5.7** Εκτελούμε "ifconfig em1 172.17.17.193/26" στο R2, "ifconfig em0 172.17.17.253/26" στο PC2 και "ifconfig em0 172.17.17.254/26" στο PC3.

**5.8** sysrc ifconfig\_em0="inet 172.17.17.134 netmask 255.255.255.252"

```
sysrc ifconfig_em1="inet 172.17.17.137 netmask 255.255.255.252"
```

```
service netif restart
```

**5.9** Εκτελούμε “ifconfig em0 172.17.17.161/27” στο PC4 και “ifconfig em0 172.17.17.190/27” στο R3.

#### **5.10**

PC1: “route add default 172.17.17.126”

PC2: “route add default 172.17.17.193”

PC3: “route add default 172.17.17.193”

PC4: “route add default 172.17.17.190”

**5.11** Στο R1: “route add -net 172.17.17.192/26 172.17.17.130” και “route add -net 172.17.17.160/27 172.17.17.130”

**5.12** Στο R2 “route add -net 172.17.17.0/25 172.17.17.137” και “route add -net 172.17.17.160/27 172.17.17.137”

**5.13** Στο R3 “route add -net 172.17.17.0/25 172.17.17.133” και “route add -net 172.17.17.192/26 172.17.17.133”

**5.14** Τα ping είναι επιτυχή

## **Άσκηση 6**

**6.1** PC2 : 08:00:27:ed:25:c4

PC3 : 08:00:27:fb:d3:06

**6.2** Εκτελούμε στο PC2 “ifconfig em0 172.17.17.254”

**6.3** Ναι, μήνυμα για χρήση της IP του από το PC3:

```
root@PC:~ # Mar 27 05:22:39 PC kernel: arp: 08:00:27:fb:d3:06 is using my IP address 172.17.17.254 on em0!
```

**6.4** Ναι, μήνυμα για χρήση της IP του από το PC2:

```
root@PC:~ # Mar 27 05:22:46 PC kernel: arp: 08:00:27:ed:25:c4 is using my IP address 172.17.17.254 on em0!
```

**6.5** Ναι, ορίστηκε κανονικά. Τα μηνύματα λάθους εμφανίζονται προκειμένου να προβούμε σε αλλαγές, ώστε να λύσουμε το ζήτημα των ίδιων IP σε ένα υποδίκτυο, ώστε να αποφύγουμε μελλοντικά προβλήματα με αποστολές από/σε αυτή τη διεύθυνση.

**6.6** Όχι, καθώς αλλάξαμε την IP του, επομένως και διαγράφηκε το default gateway

**6.7** Στο PC2: route add default 172.17.17.193

**6.8** Εκτελούμε στα PC2, PC3 και R3 “arp -a -d”.

**6.9** Στο R2: “tcpdump -i em1 -n arp”

**6.10** Στα PC2 και PC3: “tcpdump -n tcp”

**6.11**

```
root@PC:~ # ssh lab@172.17.17.254
Fssh_kex_exchange_identification: read: Connection reset by peer
Connection reset by 172.17.17.254 port 22
```

**6.12** Ναι είναι επιτυχής

**6.13** Δεν υπάρχει διαθέσιμη εγγραφή για το PC3, καταγράφεται μόνο το PC2 στη διεύθυνση 172.17.17.254.

**6.14** Πρώτα απάντησε το PC3 και στη συνέχεια το PC2

```
root@PC:~ # tcpdump -i em1 -n arp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on em1, link-type EN10MB (Ethernet), capture size 262144 bytes
10:36:29.608553 ARP, Request who-has 172.17.17.254 tell 172.17.17.193, length 28
10:36:29.608802 ARP, Reply 172.17.17.254 is-at 08:00:27:fb:d3:06, length 46
10:36:29.608836 ARP, Reply 172.17.17.254 is-at 08:00:27:ed:25:c4, length 46
```

**6.15** Ανήκει στο PC2

**6.16** Στον PC2

**6.17** Από την εντολή who στο μηχάνημα PC2, βλέπουμε σύνδεση με το χρήστη lab. Είτε μέσω του πίνακα ARP του R2, είτε μέσω του prompt που εμφανίζεται όταν κάνουμε ssh σύνδεση (lab@PC2), εφόσον έχουν ορίσει το hostname κατάλληλα στο PC2

**6.18** Την πρώτη φορά που πήγε να γίνει η σύνδεση, το R2 έκανε broadcast ARP request για να μάθει τη MAC της 172.17.17.254. Όπως είδαμε, έλαβε πρώτα απάντηση από το PC3, οπότε και η σύνδεση ξεκίνησε με το PC1 να στέλνει SYN και το PC3 να απαντάει SYN ACK. Ωστόσο επειδή όπως είπαμε ο πίνακας ARP του R2 άλλαξε, με το PC2 να αντικαθιστά το PC3, βλέπουμε ότι το PC1 στέλνει το ACK της τριπλής χειραψίας στον PC2, με το PC2 να απαντάει με το flag reset, καθώς κατάλαβε πως κάτι δε πάει καλά στη σύνδεση.

Την δεύτερη φορά γίνεται κανονικά σύνδεση ssh με το PC2 καθώς υπάρχει στο ARP table του R2 η MAC address του PC2 ως τελευταία καταχώρηση για την IP 172.17.17.254

**6.19** Παρατηρούμε ότι την πρώτη φορά το PC3 μετά το πρώτο SYN ACK προς το PC1 που απέτυχε, συνεχίζει να στέλνει τέτοια πακέτα στο PC1 (μετά τα 4 συνεχόμενα αποτυχημένα σταματάει). Ωστόσο ο PC1 ξέρει ότι η σύνδεση έχει διακοπεί καθώς έχει πάρει RST τεμάχιο από το PC2. Οπότε ο PC1 απαντά στα SYN ACK με RST 3 φορές, τα οποία και καταλήγουν στο PC2 μέσω του R2.

