

Ονοματεπώνυμο: Πυλιώτης Αθανάσιος		Όνομα PC: DESKTOP-5DLG3IF
Ομάδα: 1	Ημερομηνία: 25/03/23	

Εργαστηριακή Άσκηση 5

Στατική δρομολόγηση

Άσκηση 1: Δρομολόγηση σε ένα βήμα

1.1) PC1: **ifconfig em0 192.168.1.2/24**

PC2: **ifconfig em0 192.168.2.2/24**

R1: **ifconfig em0 192.168.1.1/24**

R1: **ifconfig em1 192.168.2.1/24**

1.2) R1: **vi /etc/rc.conf → gateway_enable="YES"**

sysctl net.inet.ip.forwarding=1

1.3) PC1: **route add -net 192.168.2.0/24 192.168.1.1**

1.4) PC1: **netstat -rn**

192.168.2.0/24 192.168.1.1 UGS em0

Οι σημαίες σημαίνουν τα εξής:

U: up, δηλαδή πως είναι ενεργή.

G: Ο προορισμός είναι πύλη

S: Η διαδρομή έχει οριστεί στατικά

1.5) PC1: **ping -c 1 192.168.2.2**

Το ping δεν επιτυγχάνει και απλά δεν λαμβάνουμε καμία απάντηση.

1.6) R1: **tcpdump -vvn -en -i em0 icmp (LAN1)**

tcpdump -vvn -en -i em1 icmp (LAN2)

Βλέπουμε πως στέλνεται μονάχα το ICMP Echo Request από το PC1. Αυτό συμβαίνει επειδή δεν έχει οριστεί στον PC2 διαδρομή για τον PC1 στον πίνακα δρομολόγησης και άρα δεν λαμβάνουμε ποτέ πίσω απάντηση.

1.7) PC2: **route add -net 192.168.1.0/24 192.168.2.1**

1.8) PC1: **ping -c 1 192.168.2.2**

Ναι είναι επιτυχές τώρα.

- 1.9) Δεν χρειάζεται να γίνει καμία αλλαγή στον πίνακα δρομολόγησης του R1 επειδή και το PC1 και το PC2 γνωρίζουν πως υπάρχει διαδρομή από το ένα υποδίκτυο στο άλλο. Η R1 έχει ήδη εγγραφές που δημιουργήθηκαν όταν αποδώσαμε διευθύνσεις στις διεπαφές του. Όταν ψάχνουν να επικοινωνήσουν μεταξύ τους ξέρουν πως αν βρουν το 192.168.x.1 (τις διεπαφές του R1) θα επικοινωνήσουν με το άλλο υποδίκτυο, άρα στέλνουν στον R1 που είναι η Gateway απευθείας και επικοινωνούν.

Άσκηση 2: Proxy ARP

2.1) PC1: **route del 192.168.2.0/24**

2.2) PC1: **ifconfig em0 192.168.1.2/20**

2.3) Από την προοπτική του PC1 τα PC2/PC3 βρίσκονται στο ίδιο υποδίκτυο, αφού μόνο τα πρώτα 20 bits χρησιμοποιούνται για το δίκτυο του, ενώ η διαφορά τους είναι στα τελευταία 12 και συγκεκριμένα στο 23^ο και στο 24^ο.

2.4) PC1: **ping -c 1 192.168.2.2, ping -c 1 192.168.2.3**

Όχι, καθώς για τα PC2/PC3 το PC1 δεν βρίσκεται στο υποδίκτυο τους, άρα ενώ πρέπει να λαμβάνουν το ICMP Echo Request, δεν στέλνουν ποτέ ICMP Echo Reply καθώς είναι σε άλλο δίκτυο και δεν έχουν route για εκείνο το δίκτυο, 192.168.1.0/20.

2.5) R1: **sysctl net.link.ether.inet.proxyall=1**

PC1: **ping -c 1 192.168.2.2**

Ναι πλέον είναι επιτυχές το ping. Αυτό συμβαίνει καθώς λειτουργεί ο δρομολογητής R1 σαν proxy, δηλαδή όταν έλαβε το ARP request και εφόσον γνωρίζει το δίκτυο του PC2, στέλνει απάντηση με τη δική του MAC. Στη συνέχεια προωθεί αντίστοιχα μετά το ICMP Request και στέλνει ARP Request, βρίσκει το PC2 και του στέλνει και πρακτικά λειτουργεί σαν μεσάζοντας. Επίσης, το PC2 έχει από πριν στατική εγγραφή στο route table του για το δίκτυο του PC1 που είναι μέσω του R1.

2.6) PC1: **ping -c 1 192.168.2.3**

Αποτυγχάνει επειδή το PC3 δεν έχει από πριν στατική εγγραφή στο route table του για το δίκτυο του PC1. Δεν υπάρχει το απαραίτητο route.

2.7) PC3: **route add -net 192.168.1.0/24 192.168.2.1**

2.8) PC1, R1, PC3: **arp -da**

2.9) R1: **tcpdump -vvv -en -i em0 (LAN1)**

tcpdump -vvv -en -i em1 (LAN2)

PC1: **ping -c 1 192.168.2.3**

2.10) Το R1 απαντάει πως Reply 192.168.2.3 is-at 08:00:27:24:d7:6a, η οποία είναι η δική του διεύθυνση MAC στη διεπαφή του em0, όπως αναμέναμε.

2.11) Προς τη διεύθυνση 08:00:27:24:d7:6a, δηλαδή τη MAC του R1 στο LAN1.

2.12) Το PC3 λαμβάνει το ICMP Echo Request από τη διεύθυνση 08:0027:60:72:35, η οποία

ανήκει στη διεπαφή στο LAN2 του R1. (αλλά η IP διεύθυνση αποστολής παραμένει 192.168.1.2)

2.13) Σχεδιασμός

PC1 → R1 (ARP Request για 192.168.2.3)

R1 → PC1 (ARP Reply με τη MAC της διεπαφής του στο LAN1)

PC1 → R1 (ICMP Echo Request for 192.168.2.3)

R1 → PC3 (ARP Request for 192.168.2.3)

PC3 → R1 (ARP Reply με τη MAC του)

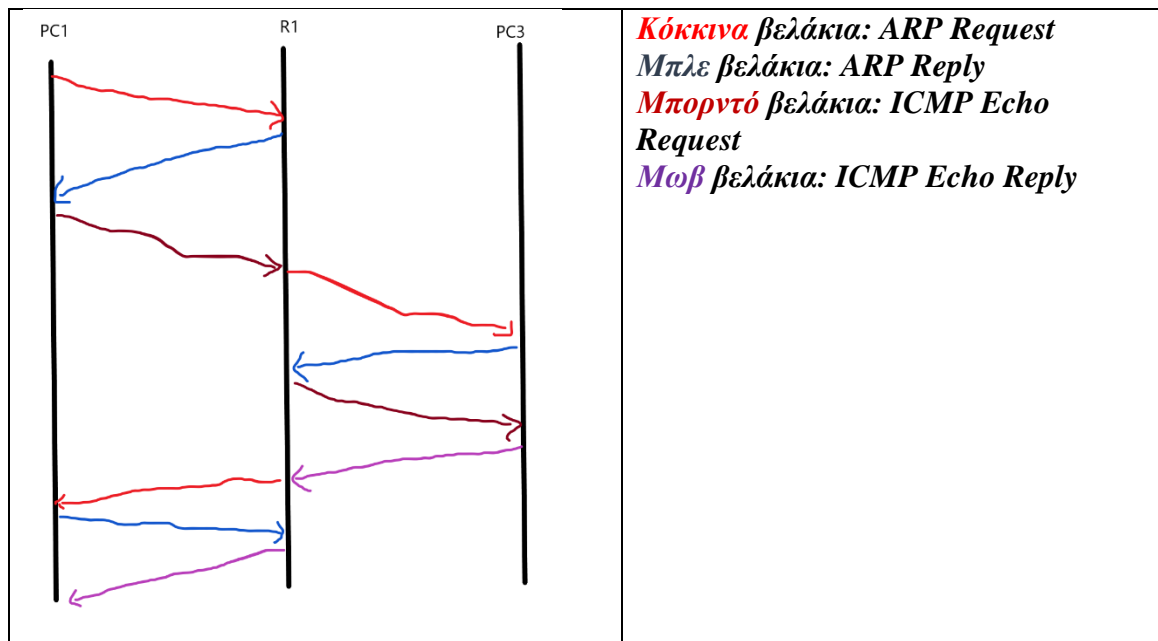
R1 → PC3 (ICMP Echo Request)

PC3 → R1 (ICMP Echo Reply)

R1 → PC1 (ARP Request για 192.168.1.2)

PC1 → R1 (ARP Reply με τη MAC του)

R1 → PC1 (ICMP Echo Reply)



2.14) Όπως αναφέρθηκε παραπάνω, η διαφορά μεταξύ των 2 υποδικτύων είναι στο 23^ο bit, άρα θα μπορούσε το PC1 να είχε τιμή μήκους προθέματος μέχρι και 22 bits.

2.15) PC1: **ifconfig em0 192.168.1.2/23**

Όντως αποτυγχάνει και εμφανίζει *no route to host*.

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

2.17) netstat -rn

Εμφανίζει τη MAC διεύθυνση της διεπαφής του στο LAN1.

2.18) Ναι είναι επιτυχές. Αυτό συμβαίνει επειδή η διεπαφή του στέλνει ARP Request για την IP3 και αυτό πετυχαίνει. Εφόσον το επόμενο βήμα είναι γνωστό, το ARP Request στέλνεται και ο δρομολογητής R1 κάνει τα υπόλοιπα, ξέρει πως είναι από το R1 η διαδρομή για το PC3.

2.19) R1: **sysctl net.link.ether.inet.proxyall=0**

2.20) PC1: **route change 192.168.2.0/24 192.168.1.1**

2.21) PC1: **ifconfig em0 192.168.1.2/24**

2.22) Παρατηρώ πως η διαδρομή διαγράφηκε από τον πίνακα προώθησης, όπως ξέραμε πως γίνεται όταν κάνουμε αλλαγές στην διεπαφή (διεύθυνση, netmask, οτιδήποτε) στο 2.21

2.23) PC1: **route add -net 192.168.2.0/24 192.168.1.1**

Άσκηση 3: Δρομολόγηση σε περισσότερα βήματα

3.1) R1: **ifconfig em0 192.168.1.1/24**

ifconfig em1 172.17.17.1/30

3.2) R2: **ifconfig em0 172.17.17.2/30**

ifconfig em1 192.168.2.1/24

sysctl net.inet.ip.forwarding=1

3.3) PC1: **ping 192.168.2.2**

Παρατηρώ destination host unreachable.

3.4) R1: **tcpdump -vvv -en -i em0 (LAN1)**

tcpdump -vvv -en -i em1 (WAN1)

Παρατηρώ πως στέλνεται ένα ICMP Echo Request και απαντάει ο R1 με ICMP host 192.168.2.2 unreachable και του στέλνει και το original Echo Request. Στο R1 δεν προωθούνται τα ICMP Echo request στο PC2 γιατί δεν έχει διαδρομή για αυτό.

3.5) PC1: **traceroute 192.168.2.2**

Παρατηρώ πως πηγαίνει προς το 192.168.1.1 2 φορές, και στο δεύτερο έχει !H, που σημαίνει destination host unreachable, όπως αναμέναμε.

3.6) R1: **route add -net 192.168.2.0/24 172.17.17.2**

3.7) PC1: **ping -c 1 192.168.2.2**

Όχι δεν μπορούμε να κάνουμε ping το PC2. Παρατηρούμε πως δεν λαμβάνουμε απάντηση.

3.8) Στο LAN2 το μόνο που παρατηρούμε είναι μηνύματα ICMP Echo Request πρώτα και μετά ICMP Echo Reply από το PC2 οφείλεται στο ping και ICMP host 192.168.1.2 unreachable. Είναι αυτό που αναμέναμε αφού δεν έχουμε ορίσει την πύλη μέσω της οποίας ο R2 προωθεί στο υποδίκτυο 192.168.1.0/24 πακέτα και για αυτό τον θεωρεί unreachable.

3.9) Όχι δεν παρατηρούμε μηνύματα ICMP Echo Request στο WAN1. Παρατηρήσαμε UDP Datagrams που στάλθηκαν από το PC1 προς το R1 (default gateway) και εκείνο τα προωθεί στο R2 (προς το WAN1)

3.10) Παρατηρούμε μόνο ICMP UDP port unreachable το οποίο είναι error message που δείχνει πως δεν μπορεί να στείλει και τον λόγο, αλλά και UDP Datagrams.

- 3.11) Δεν παρατηρούμε ICMP host unreachable. Ο R2 δεν προώθησε το μήνυμα παραπάνω (UDP Port unreachable) και δεν τον αφήνει το σύστημα να απαντήσει ICMP μηνύματα σφάλματος.
- 3.12) R2: **route add -net 192.168.1.0/24 172.17.17.1**
- 3.13) Ναι μπορούμε τώρα. Παρατηρούμε μηνύματα ICMP echo Request and ICMP Echo Reply και Time to live exceeded. Τα παρατηρούμε επειδή
- 3.14) PC2: **ping 172.17.17.1**
- Παρατηρούμε no route to host.
- 3.15) PC2: **route del 192.168.1.0/24**
- 3.16) PC2: **route add default 192.168.2.1**
- 3.17) PC2: **ping -c 1 172.17.17.1**
- Παρατηρούμε πως το ping ήταν επιτυχές τώρα.
- 3.18) Τη πρώτη φορά το PC2 έψαχνε να βρει το υποδίκτυο του PC1, γιατί δεν είχε κατάλληλη εγγραφή στον πίνακα προώθησης για το 172.17.17.0/30 και δεν μπορούσε να στείλει. Τώρα, όταν στέλνει πακέτο προς άλλο υποδίκτυο, πηγαίνει απευθείας στη default gateway R2 (αν δεν υπάρχει άλλη εγγραφή στον πίνακα προώθησης) και αυτή είναι που κάνει την αναζήτηση για το PC2 και όχι το ίδιο το PC2, που έχει στο πίνακα δρομολόγησης του κατάλληλη διαδρομή.

Άσκηση 4: Ένα πιο πολύπλοκο δίκτυο με εναλλακτικές διαδρομές

4.1) PC3: **ifconfig em0 up**

ifconfig em0 192.168.2.3/24

4.2) PC3: **route add -net 192.168.1.0/24 192.168.2.1**

4.3) Οι κάρτες δικτύου του R1 πρέπει να βρίσκονται: em0 → 192.168.1.1/24 (LAN1), em1 → 172.17.17.1/30 (WAN1), em2 → 172.17.17.5/30 (WAN2)

Τις ορίζουμε ως

R1: **ifconfig em0 192.168.1.1/24**

ifconfig em1 172.17.17.1/30

ifconfig em2 172.17.17.5/30

sysctl net.inet.ip.forwarding=1

4.4) Οι κάρτες δικτύου του R2 πρέπει να βρίσκονται: em0 → 172.17.17.2/30 (WAN1), em1 → 192.168.2.1/24 (LAN2), em2 → 172.17.17.9/30 (WAN3)

Τις ορίζουμε ως

R1: **ifconfig em0 172.17.7.2/30**

ifconfig em1 192.168.2.1/24

ifconfig em2 172.17.17.9/30

sysctl net.inet.ip.forwarding=1

4.5) Οι κάρτες δικτύου του R3 πρέπει να βρίσκονται: em0 → 172.17.17.6/30 (WAN2), em1 → 172.17.17.10/30 (WAN3)

Τις ορίζουμε ως

R1: **ifconfig em0 172.17.17.6/30**

ifconfig em1 172.17.17.10/30

sysctl net.inet.ip.forwarding=1

4.6) R1: **route add -net 192.168.2.0/24 172.17.17.2**

4.7) R2: **route add -net 192.168.1.0/24 172.17.17.1**

4.8) R3: **route add -net 192.168.1.0/24 172.17.17.5**

route add -net 192.168.2.0/24 172.17.17.9

4.9) R1: **route add -host 192.168.2.3 172.17.17.6**

Οι σημαίες που έχει είναι UGHS. Η σημαία H δείχνει πως είναι host, δηλαδή

υπολογιστής.

4.10) PC1: **tracert 192.168.2.2**

Παρατηρούμε 3 βήματα, 192.168.1.1, 172.17.17.2, 192.168.2.2

4.11) PC1: **ping -c 1 192.168.2.2**

tcpdump -i em0 -v -n icmp → TTL = 62, το οποίο σημαίνει πως έκανε 3 βήματα αφού ξεκινάει από το 64.

4.12) PC1: **tracert 192.168.2.3**

Παρατηρούμε 4 βήματα, 192.168.1.1, 172.17.17.6, 172.17.17.2, 192.168.2.3

4.13) PC1: **ping -c 1 192.168.2.3**

tcpdump -i em0 -v -n icmp → TTL = 62, το οποίο σημαίνει πως έκανε 3 βήματα αφού ξεκινάει από το 64.

4.14) Θα ακολουθήσει τη διαδρομή μέσω των WAN2, WAN3. Πρώτα πάει από το R1, και εφόσον έχει το PC3 στον πίνακα δρομολόγησης του το προωθεί μέσω του R3 για να πάει στο LAN2. Το επιβεβαιώνουμε εύκολα με ένα **tcpdump -i em0 -v -n icmp** στο R3.

4.15) Με βάση τα βήματα καταλαβαίνουμε πως το ICMP Reply ακολουθεί τη διαδρομή μέσω του WAN1 όπως είναι λογικό, αφού το R2 προωθεί μόνο προς το WAN1 και όχι προς το WAN3. Συνεπώς φαίνονται λιγότερα βήματα από όσα περιμέναμε.

4.16) Απενεργοποιούμε τη διεπαφή και στη συνέχεια

R2: tcpdump -i em1 -v -n icmp

4.17) PC1: **tracert 192.168.2.2**

Όχι δεν παρατηρώ κανενός είδους πακέτα στο LAN2, ούτε UDP

4.18) PC1: **tracert 192.168.2.3**

Παρατηρώ πως παράγονται πακέτα κάπου στα 3-4 βήματα. Είναι πακέτα ICMP 192.168.2.3 udp port 54321 unreachable με το 54321 να είναι ένας αριθμός port.

Παρατηρώ πως φτάνει μέχρι το 172.17.17.6 και μετά βγάζει ***. Είναι λογικό να μας στέλνει αυτά τα error messages αφού είναι unreachable. Δεν απαντάει με UDP, αλλά είναι μέσα στο ICMP. Παρατηρούμε και UDP προς το PC3.

4.19) R1: **route change 192.168.2.0/24 172.17.17.6**

R2: route change 192.168.1.0/24 172.17.17.10

PC1: tracert 192.168.2.2 – tracert 192.168.2.3

Δουλεύουν και τα δύο και μας δείχνουν κανονικά όλη τη διαδρομή.

4.20) R1: **route show 192.168.2.2**

route show 192.168.2.3

Παρατηρούμε πως το destination τους είναι διαφορετικό. Στο IP2 είναι 192.168.2.0 ενώ στο IP3 είναι 192.168.2.3, του ίδιου του host δηλαδή.

4.21) Η εγγραφή από τον πίνακα δρομολόγησης που επιλέγεται για το PC3, δεδομένου του longest prefix (μεγαλύτερο ταίριασμα), είναι η 192.168.2.3 αφού έχει 32/32 bits ίδια και ταιριάζει απόλυτα.

Άσκηση 5: Βρόχοι κατά τη δρομολόγηση

5.1) R3: **route change 192.168.2.0/24 172.17.17.5**

5.2) PC1: **ping -c 1 192.168.2.2**

Όχι δεν ήταν. Και λάβαμε πολλές απαντήσεις Redirect Host (λογικά 63) από το R1 ή το R3. και στο τέλος time to live exceeded από το R3.

5.3) Αυτό που συμβαίνει είναι πως στέλνεται το ICMP Echo Request από το PC1 στο R1 και μετά στο R3 και μετά πάλι στο R1 και κάνει $R1 \rightarrow R3 \rightarrow R1 \rightarrow R3 \rightarrow R1 \rightarrow \dots$ μέχρι να επιστρέψει time to live exceeded. Παρατηρούμε πως το Redirect Host στέλνεται τόσο από το 192.168.1.1 όσο και από το 172.17.17.6. Στο τέλος το time to live exceeded στέλνεται από το 172.17.17.6.

5.4) R1: **tcpdump -vvn -en -i em0 -l | tee data**

R3: **tcpdump -vvn -en -i em0 -l | tee data**

5.5) Παρατηρούμε:

R1: **grep 'x' data | wc -l** $\rightarrow x = \{\text{request, redirect, exceeded}\}$

R3: **grep 'x' data | wc -l**

LAN1: 1 ICMP Echo request, 62 ICMP redirect host και 1 ICMP Time to live exceeded

WAN2: 63 ICMP Echo request, 31 ICMP redirect host, 1 ICMP Time to live exceeded

5.6) R3: **tcpdump -vvn -XX -en -i em0 'icmp[0] == 8'**

5.7) PC1: **ping -c 1 192.168.2.2**

Παρατηρούμε πως captured έγιναν 63 πακέτα και λήφθηκαν 95 πακέτα, άρα έχουμε 63 ICMP Echo Request. Όλα είναι από το 192.168.1.2 για αυτό θα κοιτάξουμε και με βάση MAC. Παρατηρούμε πως πηγαίνει εναλλάξ και τελικά είναι 32 από το R1 και 31 από το R3, αφού ο R1 ξεκινάει. Είναι 63, μέχρι να λήξει το TTL πρακτικά.

5.8) R1: **tcpdump -vvn -XX -en -i em0 'icmp[0] == 5'**

R3: **tcpdump -vvn -XX -en -i em0 'icmp[0] == 5'**

5.9) PC1: **ping -c 1 192.168.2.2**

Παρατηρώ 62 ICMP Echo Redirect στο LAN1 και 31 ICMP Echo Redirect στο WAN2.

Ο R3 συνολικά λαμβάνει 31 ICMP Echo Request (βλέπε 5.5) τα οποία του προωθεί ο R3 πρακτικά

5.10) Εμφανίζονται 62 ICMP Echo Redirect στο LAN1. Είναι ένα για κάθε ICMP Echo

Request που υπάρχει. Τα 31 πρέπει να είναι από τον R3 και τα υπόλοιπα από τον R1, καθώς δια κάθε ICMP Echo Request που λαμβάνουν στέλνουν redirect. Το R1 όμως για το πρώτο δεν στέλνει ICMP Echo Redirect επειδή ξέρει πως πρέπει να το στείλει στο R3.

5.11) R1: **tcpdump -n -i em0 -l | tee data**

R3: **tcpdump -n -i em0 -l | tee data**

PC1: **traceroute -I -q 1 192.168.2.2**

Παρατηρούμε 64 βήματα (λογικό με βάση TTL = 64). Η διαδρομή που καταγράφεται είναι PC1→R1→R3→R1→R3→R1→...→R1→R3 και μετά επιστρέφει και τελειώνει

5.12) Ctrl+c.

R1: **grep 'request' data | wc -l**

R3: **grep 'request' data | wc -l**

Παρατηρούμε πως από το PC1 στάλθηκαν 64 ICMP Echo Request από το PC1 αφού είχαμε flag -q 1 (δηλαδή ένα probe ανά βήμα για τα 64 βήματα που είναι default) ενώ στο WAN2 βλέπουμε 2016 ICMP Echo Request πακέτα. Καταγράφονται περισσότερα από όσα περιμέναμε. Αυτό είναι επειδή στο πρώτο βήμα (TTL = 1) στέλνεται 1, στο δεύτερο (TTL = 2) 2, στο τρίτο 3 Και τελικά είναι $1+2+3+\dots+64 = 64*63/2 = 32*63 = 2016$ όπως και βρήκαμε.

5.13) R3: **grep 'exceeded' data | wc -l**

Παρατηρούμε 32 ICMP Time to live exceeded. Αυτά τα μηνύματα στέλνονται αποκλειστικά από τον R3 όταν λήγει, και άρα στέλνει 32 γιατί τόσα requests λήγουν.

5.14) **tcpdump -n -i em0 'icmp[0] == 8'** για το ICMP Echo Request και κωδικός 11 για το time to live exceeded. Μετά όταν κλείσουμε την καταγραφή βλέπουμε το πλήθος στα packets captured τα οποία είναι αυτά που θέλουμε σε πλήθος.

Άσκηση 6: Χωρισμός σε υποδίκτυα

Σε αυτό το σημείο θα χωρίσουμε το δίκτυο μας σε υποδίκτυα:

Note: στα δίκτυα περιλαμβάνονται και η IP δικτύου (όλα 0) και η IP broadcast (όλα 1)

LAN1	LAN2	LAN3
172.17.17.0 - 172.17.17.127	172.17.17.192 - 172.17.17.255	172.17.17.160 - 172.17.17.191
WAN1	WAN2	WAN3
172.17.17.128 - 172.17.17.131	172.17.17.132 - 172.17.17.135	172.17.17.136 - 172.17.17.138

LAN1: 00000000 prefix number → 25 (126 PC)

WAN1: 10000000 prefix number → 30 (2 PC)

WAN2: 10000100 prefix number → 30 (2 PC)

WAN3: 10001000 prefix number → 30 (2 PC)

LAN2: 11000000 prefix number → 26 (62 PC)

LAN3: 10100000 prefix number → 27 (30 PC)

Άδειο για μετέπειτα επέκταση: 1001000 prefix number → 28 (30 PC)

10001100 prefix number → 30 (2 PC)

LAN1	WAN1	WAN2	WAN3	Empty	LAN3	LAN2
------	------	------	------	-------	------	------

6.1) LAN1: 172.17.17.0/25

6.2) LAN2: 172.17.17.192/26

6.3) LAN3: 172.17.17.160/27

6.4) PC1: **ifconfig em0 172.17.17.1/25**

R1: ifconfig em0 172.17.17.126/25

ifconfig em1 172.17.17.129/30

ifconfig em2 172.17.17.133/30

6.5) PC4: **ifconfig em0 172.17.17.161/27**

R3: ifconfig em2 172.17.17.190/27

ifconfig em0 172.17.17.134/30

ifconfig em1 172.17.17.137/30

6.6) R2: **ifconfig em1 172.17.17.193/26**

ifconfig em0 172.17.17.130/30

ifconfig em2 172.17.17.138/30

PC2: **ifconfig em0 172.17.17.253/26**

PC3: **ifconfig em0 172.17.17.254/26**

6.7) 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**

6.8) R1: **route add -net 172.17.17.192/26 172.17.17.130**

R1: **route add -net 172.17.17.160/27 172.17.17.130**

6.9) R2: **route add -net 172.17.17.0/25 172.17.17.137**

R2: **route add -net 172.17.17.160/27 172.17.17.137**

6.10) R3: **route add -net 172.17.17.0/25 172.17.17.133**

R3: **route add -net 172.17.17.192/26 172.17.17.133**

6.11) PC1: **ping -c 1 172.17.17.253**

PC2: **ping -c 1 172.17.17.162**

PC3: **ping -c 1 172.17.17.1**

Επικοινωνούν κανονικά και δουλεύουν τα παραπάνω pings

Άσκηση 7: Ταυτόσημες διευθύνσεις IP

7.1) MAC PC2: 08:00:27:1e:69:ab

MAC PC3: 08:00:27:87:2a:44

7.2) PC2: **ifconfig em0 172.17.17.254/26**

7.3) Ναι, λάβαμε την ένδειξη PC kernel: arp: 08:00:27:87:2a:44 (MAC PC3) is using my IP address 172.17.17.254 on em0!

7.4) Ναι εμφανίστηκε η PC kernel: arp: 08:00:27:1e:69:ab (MAC PC2) is using my IP address 172.17.17.254 on em0!

7.5) Ναι η διεύθυνση έχει οριστεί. Το νόημα των μηνυμάτων λάθους είναι να μας ενημερώσουν για την ύπαρξη της ίδιας IP σε 2 μηχανήματα και να κάνουμε εμείς, σαν χρήστης/διαχειριστής τις απαραίτητες αλλαγές. Για να κάναμε αυτή την αλλαγή μάλλον είχαμε λόγο, οπότε απλά μας λέει πως αυτή τη στιγμή είναι λάθος και να συνεχίσουμε να το κάνουμε σωστό. Είναι πιο εύκολο να κάνεις τις αλλαγές έτσι, ειδικά όταν έχεις χρησιμοποιήσει όλες τις IP (που τότε, αν δεν μας άφηνε να το αλλάξουμε, θα έπρεπε να ορίσουμε εξ αρχής όλο το σύστημα)

7.6) Όχι δεν παρέμεινε, επειδή όπως έχουμε πει όταν κάνουμε αλλαγές στις IP και στο δίκτυο διαγράφει όλα τα routes.

7.7) PC2: **route add default 172.17.17.193**

7.8) R2, PC2, PC3: **arp -da**

7.9) R2: **tcpdump -vvv -en -i em1 arp**

7.10) PC2: **tcpdump -vvv -en -i em0 tcp**

PC3: **tcpdump -vvv -en -i em0 tcp**

7.11) PC1: **ssh lab@172.17.17.254**

Όχι δεν εμφανίστηκε ένδειξη λάθους Βλέπουμε ένα ARP Request και δύο ARP Reply. Βλέπουμε αρκετά TCP πακέτα στα PC2, PC3 αλλά τελικά δεν συνδέθηκε σε κάποιον. Στο PC2 βλέπουμε μηνύματα από 172.17.17.1.39268 > 172.17.17.254.22 (έρχονται όντως από R2 και πάνε στο PC2) ενώ στο PC3 172.17.17.254.22 > 172.17.17.1.39268 (είναι από PC3 και πάνε R2). Άρα το PC1 στέλνει στο PC2 και το PC3 απαντάει στο PC1 οπότε δεν επιτυγχάνει το ssh. Αυτό που συνέβη είναι πως το πρώτο πακέτο της τριπλής χειραψίας στάλθηκε στο PC3 που έστειλε πρώτο ARP Reply και μετά το τρίτο μέρος

στέλνεται στο PC2 επειδή άλλαξε το ARP Table του R2. Το PC3 παράλληλα συνεχίζει να στέλνει την απάντηση στη τριπλή χειραψία, αλλά ποτέ δεν είναι αυτή ολοκληρωμένη και από τα 2 μέρη, οπότε αποτυγχάνει.

7.12) Ναι καταφέραμε να συνδεθούμε.

7.13) R2: **arp -a**

172.17.17.254 at 08:00:27:1e:69:ab που είναι το PC2

Δεν υπάρχει εγγραφή για το PC3.

7.14) Παρατηρώ πως πρώτο απάντησε το PC3 και μετά το PC2 και αυτό που καταγράφηκε στον πίνακα είναι το PC2.

7.15) Η διεύθυνση MAC που έχει ο πίνακας ARP του R2 ανήκει στο PC2.

7.16) Συνδεθήκαμε στο PC2, αυτό στο οποίο προωθεί η R2

7.17) Μπορούμε να προσθέσουμε αρχείο και να δούμε την αλλαγή στο home folder του αντίστοιχου PC.

7.18) Τη πρώτη φορά δεν λειτούργησε επειδή η αρχική επικοινωνία έγινε με το PC3 το οποίο έστειλε πρώτο και οι υπόλοιπες από το PC1 έγιναν με το PC2 γιατί άλλαξε η MAC στην οποία προωθεί το R2 στον ARP Table. Συνεπώς την απάντηση την είχε το PC3 αλλά ο PC1 δεν την έλαβε ποτέ γιατί το R2 τη μπλόκαρε επειδή περίμενε να στείλει απάντηση από το PC2. Και τις επανα-προσπάθειες τις είχε στο PC2 από το PC1. Τη δεύτερη φορά λειτούργησε επειδή το ARP Table δεν άλλαξε κατά τη διάρκεια της τριπλής χειραψίας και ήταν επιτυχής η σύνδεση.