

Εργαστήριο Δικτύων Υπολογιστών

Εργαστηριακή Άσκηση 5

Δημήτριος Κόγιος

03119220

Όνομα PC: lekog-HP-Laptop-15s-fq1xxx

Άσκηση 1:

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) gateway_enable='YES'

1.3) route add -net 192.168.2.0/24 192.168.1.1

1.4) UGS : U : Η διαδρομή είναι ενεργή

G : Ο προορισμός είναι πύλη, που θα αποφασίσει για το πώς θα προωθήσει τα πακέτα περαιτέρω.

S : Η διαδρομή έχει οριστεί στατικά.

1.5) Φτάνουν ICMP echo requests στο PC2 αλλά δεν φτάνουν ICMP echo replies στο PC1 αφού το PC2 δεν έχει route για το PC1.

1.6) Βλέπουμε ICMP echo requests και στο LAN1 και στο LAN2 αλλά κανένα ICMP echo reply καθώς το PC2 δεν έχει route για το PC1.

1.7) route add -net 192.168.1.0/24 192.168.2.1

1.8) Ναι.

1.9) Γιατί όταν ορίσαμε IP διευθύνσεις για τις διεπαφές του, δημιουργήθηκαν αυτόματα εγγραφές στον πίνακα δρομολόγησής του για τα δίκτυα στα οποία ανήκουν οι κάρτες δικτύου του.

Άσκηση 2:

2.1) route del 192.168.2.0/24

2.2) ifconfig em0 192.168.1.2/20

2.3) Στο ίδιο γιατί τα πρώτα 20 bits και των τριών PC είναι 192.168.[0000|.....] .

2.4) Όχι, παίρνουμε μήνυμα "Host is down".

2.5) Ναι πλέον είναι επιτυχές γιατί ο R1 απάντησε στο ARP request του PC1 με ARP reply που περιείχε τη MAC διεύθυνση του R1. Αυτό συνέβη γιατί είχαμε ενεργοποιημένο το Proxy ARP και ο R1 απαντά σε οποιοδήποτε ARP request απευθύνεται σε υπολογιστή υποδικτύου για το οποίο διαθέτει εγγραφή στον πίνακα δρομολόγησής του.

2.6) Γιατί το PC3 δεν έχει route προς το PC1.

2.7) route add -net 192.168.1.0/24 192.168.2.1

2.8) arp -d -a

2.9) `tcpdump -i em{0,1} -e -v`
`ping -c 1 192.168.2.3`

2.10) Βάζει στο ARP reply τη δικιά του MAC διεύθυνση.

2.11) Στη MAC του R1.

2.12) Από τη MAC της διεπαφής του R1 στο LAN2.

2.13) PC1 -> Broadcast (ARP request) : το PC1 προσπαθεί να μάθει την MAC διεύθυνση του PC3 αφού νομίζει ότι βρίσκονται στο ίδιο υποδίκτυο.

R1 -> PC1 (ARP reply) : το R1 απαντά με τη δική του MAC λόγω ενεργοποίηση Proxy ARP.

PC1 -> R1 (ICMP echo request) : ping

R1 -> Broadcast (ARP request) : το R1 θέλει να μάθει τη MAC διεύθυνση του PC3 γιατί λαμβάνει πακέτο IPv4 με την IP διεύθυνσή του.

PC3 -> R1 (ARP reply)

R1 -> PC3 (ICMP echo request) : προώθηση

PC3 -> R1 (ICMP echo reply) : λόγω της στατικής εγγραφής του ερωτήματος 2.7

R1 -> Broadcast (ARP request) : το R1 θέλει να μάθει τη MAC διεύθυνση του PC1.

PC1 -> R1 (ARP reply)

R1 -> PC1 (ICMP echo request) : προώθηση

2.14) 22 γιατί τα PC2/3 έχουν IP της μορφής :
192.168.[000000|1x].0 και το PC1 πρέπει να νομίζει ότι βρίσκονται στο ίδιο υποδίκτυο με εκείνο.

2.15) `ifconfig 192.168.1.2/23`

2.16) route add -net 192.168.2.0/24 -interface em0

2.17) Η MAC διεύθυνση της διεπαφής em0.

2.18) Ναι είναι επιτυχές γιατί λόγω της εντολής στο 2.16 , το PC1 συνεχίζει να θεωρεί ότι το PC3 βρίσκεται στο ίδιο υποδίκτυο και συμβαίνει το ίδιο με τα προηγούμενα ερωτήματα.

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) ifconfig em0 192.168.1.2/24

2.22) Εξαφανίστηκε.

2.23) route add -net 192.168.2.0/24 192.168.1.1

Άσκηση 3:

3.1) ifconfig em0 192.168.1.1/24
ifconfig em1 172.17.17.1/30

3.2) ifconfig em0 172.17.17.2/30
ifconfig em1 192.168.2.1/24

3.3) “Destination Host Unreachable” από το 192.168.1.1

3.4) Στο WAN1 δεν παράγεται καθόλου κίνηση γιατί ο R1 δεν έχει κάποια εγγραφή ώστε να προωθήσει τα echo requests.
Στα LAN1 παράγονται μηνύματα ICMP echo request και ICMP host unreachable.

3.5) Παίρνουμε μήνυμα “failed to connect to peer for src addr selection”. Σημαίνει ότι δεν έχει ρυθμιστεί κάτι καλά στο default gateway, συγκεκριμένα όπως είπαμε ο R1 δεν έχει κάποια εγγραφή για το PC2.

3.6) `route add -net 192.168.2.0/24 172.17.17.2`

3.7) Όχι δεν λαμβάνουμε απάντηση. Γιατί ο R2 δεν έχει εγγραφή για το PC1.

3.8) Παρατηρούμε ICMP echo requests, ICMP echo replies, ICMP host unreachable. Τα ICMP echo requests παράγονται λόγω του ping, τα ICMP echo replies τα στέλνει το PC2 προς το R2 αφού αυτό το route είχαμε ορίσει στην προηγούμενη άσκηση και τα ICMP host unreachable τα στέλνει ο R2 στο PC2 γιατί δεν έχει εγγραφή για το PC1.

3.9) Παράγονται UDP datagrams καθώς στο FreeBSD η traceroute στέλνει UDP by default.

3.10) UDP port unreachable.

3.11) Γιατί δεν παράγεται μήνυμα σφάλματος σε μήνυμα σφάλματος.

3.12) `route add -net 192.168.1.0/24 172.17.17.1`

3.13) Ναι πλέον επιτυγχάνει με ttl = 3. Στο WAN1 παρατηρώ τρία UDP πακέτα, τρία μηνύματα ICMP time exceeded τα οποία στέλνει ο R2 προς το PC1 όταν το αρχικό ttl είναι 2. Όταν όμως το TTL γίνει 3 βλέπω μόνο UDP πακέτα συνολικά 6 από R1 προς R2

και από R2 προς R1 με αρχικούς και τελικούς προορισμούς τα PC1, PC2.

3.14) Μηνύματα “No route to host”.

3.15) `route del 192.168.1.0/24`.

3.16) `route add default 192.168.2.1`.

3.17) Πλέον είναι επιτυχές.

3.18) Στο ερώτημα 17, τα ICMP echo requests του PC2 πηγαίνουν στο default gateway και από εκεί στο τοπικό δίκτυο WAN1.

Στο ερώτημα 14, δεν είχαμε ορισμένο default gateway για το PC2 και συνεπώς δεν έκανε match κανένα prefix.

Άσκηση 4:

4.1) Συνδέουμε το καλώδιο

```
ifconfig em0 192.168.2.0/24
```

4.2) `route add -net 192.168.1.0/24 192.168.2.1`

4.3) LAN1, WAN1, WAN2

```
ifconfig em{0,1,2} <IP>
```

4.4) WAN1, LAN2, WAN1

```
ifconfig em{0,1,2} <IP>
```

4.5) WAN2, WAN3

```
ifconfig em{0,1} <IP>
```

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
route add -net 192.168.2.0/24 172.17.17.9

4.9) route add -host 192.168.2.3 172.17.17.6

Η σημαία H.

4.10) 3 βήματα (192.168.1.1 , 172.17.17.2, PC2)

4.11) ttl = 62.

4.12) 4 βήματα (192.168.1.1, 172.17.17.6, 172.17.17.2, 192.168.2.3)

4.13) ttl = 62.

4.14) PC1 -> R1 -> R3 -> R2 -> PC3

4.15) PC3 -> R2 -> R1 -> PC1.

Γιατί το R2 έχει εγγραφή για το δίκτυο του PC1 μέσω του R1 από το ερώτημα 4.7.

4.16) tcpdump -i em1

4.17) Όχι.

4.18) Ναι φτάνει ένα πακέτο στο PC3 και παράγονται πακέτα
udp port unreachable από 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

Υπάρχει επικοινωνία.

4.20) route get <IP>

Για το PC2 έχουμε τη διεύθυνση του υποδικτύου του ως destination ενώ για το PC3 έχουμε την ίδια του την διεύθυνση.

4.21) 192.168.2.3 -> 172.17.17.6

Άσκηση 5:

5.1) route change 192.168.2.0/24 172.17.17.5

5.2) Όχι.

5.3) Γίνονται συνεχόμενα redirects μεταξύ 192.168.1.1 και 172.17.17.6 γιατί το PC1 ξεκινάει στέλνοντας το ICMP echo request στο R1 το οποίο το προωθεί στο R3 το οποίο όμως λόγω του ερωτήματος 5.1 το ξαναπροωθεί στο R1 κ.λπ κ.λπ. Μέχρι που κάποια στιγμή μηδενίζεται το TTL. Το μήνυμα Time to live exceeded έρχεται από 172.17.17.6.

5.4) R1 : tcpdump -i em0 -e -l | tee R1

R3 : tcpdump -i em0 -e -l | tee R3

5.5) Στο LAN1 : 62 redirects, 1 requests, 1 ttl exceeded

Στο WAN2 : 31 redirects, 63 requests, 1 ttl exceeded

5.6) tcpdump -i em0 -e '(icmp[0]=0x8)'

5.7) 63 ICMP Echo requests εμφανίστηκαν στο WAN2.

32 από το R1 και 31 από το R3.

Λογικό το R1 να παράγει ένα παραπάνω αφού από εκεί ξεκινάει το πακέτο μετά το PC1.

5.8) R1 : tcpdump -i em0 -vvn '(icmp[0]=0x5)'

R3 : tcpdump -i em0 -vvn '(icmp[0]=0x5)'

5.9) 31 πακέτα στο WAN2 δηλαδή ένα λιγότερο από το ερώτημα 5.7.

5.10) 62 δηλαδή ένα λιγότερο από 5.7.

5.11) 192.168.1.1 -> 172.17.17.6 -> 192.168.1.1 -> 172.17.17.6

64 βήματα

5.12) 64 ICMP echo requests έστειλε το PC1.

2016 στο WAN2. Αυτό συμβαίνει γιατί $1 + 2 + \dots + 63 = 2016$.

5.13) 32 γιατί το traceroute αυξάνει κάθε φορά το TTL κατά ένα οπότε τα μισά από όσα έστειλε το PC1 θα “πεθάνουν” στο R3 και τα άλλα μισά στο R1.

5.14) tcpdump <flags> | grep {request,exceeded} | wc

Άσκηση 6:

6.1) LAN1 : 172.17.17.0/25 (θέλουμε 7 bits για hosts)

6.2) LAN2 : 172.17.17.192/26 (θέλουμε 6 bits για hosts και το τελευταίο octet ξεκινάει με 11 και όχι 10 για να μην έχουμε επικάλυψη με τα WAN)

6.3) LAN3 : 172.17.17.160/27 (θέλουμε 5 bits για hosts)

6.4) PC1 : 172.17.17.1/25

R1 : 172.17.17.126/25 (emo)

6.5) PC4 : 172.17.17.161/27

R3: 172.17.17.190/27 (emo)

6.6) R2 : 172.17.17.193/26 (em1)

PC2 : 172.17.17.253/26

PC3 : 172.17.17.254/26

6.7) PC1 : route add default 172.17.17.126

PC2, PC3 : route add default 172.17.17.193

PC4: route add default 172.17.17.190

6.8) route add -net 172.17.17.192/26 172.17.17.130

route add -net 172.17.17.160/27 172.17.17.130

6.9) route add -net 172.17.17.0/25 172.17.17.137

route add -net 172.17.17.160/27 172.17.17.137

6.10) route add -net 172.17.17.0/25 172.17.17.133

route add -net 172.17.17.192/26 172.17.17.133

6.11) Όλα τα ping είναι επιτυχή.

Άσκηση 7:

7.1) PC2 : 08:00:27:6f:8d:26

PC3 : 08:00:27:8a:03:d9

7.2) ifconfig em0 172.17.17.254/26

7.3) Ναι βλέπω μήνυμα ότι η MAC του PC3 χρησιμοποιεί την .254.

7.4) Ναι αντίστοιχα βλέπω μήνυμα ότι η MAC του PC2 χρησιμοποιεί την .254.

7.5) Ναι έχει οριστεί. Το μήνυμα δρα ως προειδοποίηση.

7.6) Όχι, το R2 δεν εμφανίζεται ως default gateway πλέον. Ο λόγος είναι ότι θα μπορούσαμε να είχαμε ορίσει στο PC2 μία άσχετη IPv4 διεύθυνση πχ 1.1.1.1/24 και τότε άμα η εγγραφή default παρέμενε ως είχε, το PC2 θα βρισκόταν σε διαφορετικό δίκτυο από την εγγραφή default.

7.7) route add default 172.17.17.193

7.8) arp -d -a

7.9) tcpdump -i em1 -n '(arp)'

7.10) tcpdump -n '(tcp)'

7.11) ssh lab@172.17.17.254

Connection reset by peer.

Δεν με αφήνει να συνδεθώ.

7.12) Ναι.

7.13) 172.17.17.254 at 08:00:27:6f:8d:26

7.14) Πρώτα το PC3 και μετά το PC2.

7.15) Στο PC2 που απάντησε δεύτερο.

7.16) Στο PC2 (κάνουμε ifconfig από την κονσόλα του ssh και βλέπουμε την MAC).

7.17) Μέσω της εντολής who από την κονσόλα του PC2 ή με netstat για την port 22.

7.18) Το PC3 απάντησε πρώτο στο ARP request και έτσι πήρε το πρώτο πακέτο της τριπλής χειραψίας (SYN), μετά το PC3 έστειλε στο PC1 το δεύτερο πακέτο της χειραψίας (SYN, ACK) και το PC1 έστειλε στην διεύθυνση 172.17.17.254 το τρίτο πακέτο της χειραψίας (ACK) όμως μόλις αυτό το τρίτο πακέτο έφτασε στον R2, η εγγραφή του πίνακα ARP είχε αλλάξει και πλέον το τρίτο πακέτο της χειραψίας έφτασε στο PC2 το οποίο δεν γνώριζε τίποτα για τη σύνδεση αυτή για αυτό και την έκανε reset.

