LAB-04 (IPv4, Ping, Fragmentation)

Ex. 1 | Latency

1.1 | `ping www.mit.edu -n 3 -4`

1.2 | Καταγράφεται μόνο η unicast κίνηση του δικτύου, δηλαδή μόνο όσα μηνύματα έχουν συγκεκριμένο παραλήπτη και αποστολέα.

1.3 | 0% loss, Average = 60ms

1.4 | Minimum = 51ms, Maximum = 79ms, Average = 60ms

1.5 |
Echo 1 RTT : 0.079712 s
Echo 2 RTT : 0.052519 s
Echo 3 RTT : 0.051504 s
Ναί, συμφωνούν (minimum → echo 3, maximum → echo 1)

1.6 | Display filter : ip

1.7 | Display filter : icmp.type == 8 or icmp.type == 0
(8 → request, 0 → reply)

1.8 | Type: 8 (Echo (ping) request)

1.9 | Source Address: 192.168.1.8, Destination Address: 184.30.212.47

1.10 | Type: 0 (Echo (ping) reply)

1.11 | Source Address: 184.30.212.47, Destination Address: 192.168.1.8

1.12 | Η IPv4 διεύθυνση του ιστότοπου www.mit.edu έχει αλλάξει από 18.7.22.83 σε 184.30.212.47

Ex. 2 | Ping

2.1 | ping <address> -n 5 -4

2.2 | Έχουν καταγραφεί μόνο 5 (αυτά που στάλθηκαν στο default gateway)

2.3 | Destination : 192.168.1.1 (default gateway)

2.4 | Όχι, δεν παρατήρησα,. Αυτό διότι τα πακέτα ICMP αυτά περνάνε από τον οδηγό loopback (σύμφωνα με το σχήμα) και δεν βγαίνουν ποτέ στο τοπικό δίκτυο οπότε το wireshark δεν τα εντοπίζει.

2.5 | Όχι , δεν παρατήρησα. Τα ICMP μηνύματα αυτά οδηγούνται και αυτά στον οδηγό loopback, οπότε και πάλι δεν παιρνούν από την ουρά εισόδου IPv4 για να τα δει το wireshark.

2.6 | Στο ping στο 192.168.1.8 το πακέτο εισέρχεται στον οδηγό Ethernet και αυτός το στέλνει στον οδηγό loopback. Ενώ, στο ping στο 127.0.0.1 το πακέτο εισέργεται κατευθείαν στον οδηγό loopback και στέλνεται πίσω στην είσοδο πακέτων IPv4.

DIRECTORY

Ex. 1 | Latency
Ex. 2 | Ping
Ex. 3 | IPv4 Headers
Ex. 4 | Fragmentation

DICTIONARY

ping command
Χρησιμοποιείται για διαγνωστικούς λόγους και για μετρήσεις επίδοσης

Echo Request / Αίτηση Ηχούς
μήνυμα που στέλνεται από μία IP σε μία άλλη

Echo Reply / Απάντηση Ηχούς

RTT (Round-Trip Time)

Βρόχος επιστροφής / loopback
Εικονική διεπαφή που χρησιμοποιείται για την επικοινωνία διεργασιών (συναντάται ως 127.0.0.1

MTU (Maximum Transmission Unit)
Μέγεθος μεγαλύτερου πακέτου IPv4 που μπορεί να μεταδοθεί χωρίς θρυμματισμό

2.7 | Όταν κάνω ping το www.netflix.com δεν υπάρχουν ping replies σε αντίθεση με το www.amazon.com και το πιο πιθανό είναι οτι το Netflix (ή κάποιος άλλος ενδιάμεσος) έχει ενεργοποιήσει κάποιο firewall που μπλοκάρει τα ICMP πακέτα.

Ex. 3 | IPv4 Headers

3.1 | Capture filter : host 147.102.40.15

3.2 | Display filter : ip.src == 192.168.1.8

3.3 |
1. Version (4 bits)
2. Header Length (4 bits)
3. Differentiated Services Field (1 byte)
4. Total Length (2 bytes)
5. Identification (2 bytes)
6. Flags (1 byte)
7. Fragment Offset (1 byte)
8. Time to Live (1 byte)
9. Protocol (1 byte)
10. Header Checksum (2 bytes)
11. Source Address (4 bytes)
12. Destination Address (4 bytes)

3.4 | Αλλάζουν τα πεδία 'Total Length' και 'Identification'

3.5 | Ναί, είναι (20 bytes).

3.6 | Το μικρότερο είναι : 40 bytes και το μεγαλύτερο : 66 bytes

3.7 | Παίρνει τις τιμές 00 (HEX) → CS0 : Standard Service class και b8 (HEX) → EF PHB : Telephony Service class

3.8 | Αυξάνονται με έναν μετρητή

3.9 | Έχει τιμή 1

3.10 | Έχει τιμή 0

3.11 | Έχει τιμή 06 (HEX) και αντιστοιχεί στο πρωτόκολλο TCP

3.12 | Γιατί αναπαριστά το άθροισμα των λέξεων που περιέχονται στο IPv4 header και από τη στιγμή που υπάρχουν πεδία που αλλάζουν (πχ το identification) αλλάζει και αυτό.

Ex. 4 | Fragmentation

4.1 | `ping <address> -n 1 -4 -f -l <size>`

4.2 | Η μέγιστη τιμή είναι 1472 bytes

4.3 | Η ελάχιστη τιμή για θρυμματισμό είναι τα 1473 bytes

4.4 | Capture filter : not multicast and not broadcast

4.5 | Display filter : ip.addr == 192.168.1.2

4.6 | Όχι, δεν παράγονται. Γιατί το πακέτο που πάει να μεταδοθεί ξεπερνάει το μήκος της MTU και δεν μεταδίδεται.

4.7 | Το μέγεθος MTU είναι επομένως 1514 γιατί αυτό είναι το συνολικό μέγεθος του πακέτου IPv4 όπως το κατέγραψε το

wireshark

4.8 | Από την επικεφαλίδα ICMP και πεδίο Data προκύπτει maximum Length : 1472 bytes

4.9 | Για μήκος δεδομένων 1472 και χωρίς τη παράμετρο -f επιτυχγάνεται το ping

4.10 | Το μεγαλύτερο πακέτο IPv4 (επικεφαλίδες Ethernet II, IPv4, ICMP) έχει μήκος 1514 bytes

4.11 | Όχι, έχει μεταφερθεί ώς πολλά

4.12 | Χρειάστηκαν 5 πακέτα γιατί το κάθε ένα έχει μέγιστο μήκος ICMP 1480 bytes και επειδή ceiling(6000/1480) = 5 δημιουργούνται 5 πακέτα

4.13 |
(Identification, Don't Fragment Bit, More Fragments Bit, Fragment Offset) → { (cb52, 0, 1, 0), (cb52, 0, 1, 1480), (cb52, 0, 1, 2960), (cb52, 0, 1, 4440), (cb52, 0, 0, 5920) }

4.14 | To flag 'More Fragments Bit'

4.15 | Το πεδίο 'Fragment Offset' το οποίο είναι 0 (δηλαδή δεν υπάρχει προηγούμενο)

4.16 | Μήκος σε δεδομένα είναι 1480 bytes και μήκος πακέτου είναι 1514 bytes

4.17 | Το πεδίο 'Fragment Offset' το οποίο δεν είναι 0

4.18 | Ναι, ακολουθούν

4.19 | Είναι ενεργοποιημένο το flag 'More fragments'

4.20 | Μόνο το πεδίο 'Fragment Offset'

4.21 | Για το πρωτελευταίο η τιμή είναι 4440 = 3 * 1480 δηλαδή έχουν προηγηθεί 3 θραύσματα με μήκος δεδομένων 1480. Για το τελευταίο η τιμή είναι 5920 = 4 *1480 και προκύπτει αντίστοιχα.

4.22 | Τα πεδία : 'Fragment Offset', 'More fragments' και 'Total Length'