

Όνοματεπώνυμο: Παναγιώτης Σταματόπουλος		Ομάδα: 6
Όνομα PC/ΛΣ: TakisAsus/Windows 10		Ημερομηνία: 31/10/23
Διεύθυνση IP: 192.168.2.2	Διεύθυνση Mac: D4-5D-64-59-27-5F	

AM: el20096

Εργαστηριακή Άσκηση 4

Πρωτόκολλο IPv4 και Θρυμματισμός

Άσκηση 1:

*Στον προσωπικό υπολογιστή

- 1.1: ping www.mit.edu -n 3 -4
 - 1.2: Lost = 0 (0% loss), Average = 47ms
 - 1.3: P1: time=48ms, P2: time=48ms, P3: time=47ms
 - 1.4: Έχουμε 3 πακέτα αντί για 4, άλλη IP address για την ιστοσελίδα, μικρότερα RTT και μεγαλύτερα TTL
 - 1.5: Με το φίλτρο συλλαμβάνουμε μόνο unicast κίνηση από και προς τον υπολογιστή
 - 1.6: ip
 - 1.7: icmp
 - 1.8: Echo (ping) request
 - 1.9: Source IP Address: 192.168.2.2
Destination IP Address: 92.122.34.235
 - 1.10: Echo (ping) reply
 - 1.11: Source IP Address: 92.122.34.235
Destination IP Address: 192.168.2.2
 - 1.12: P1: 0.047947 s \approx 0.048 s = 48 ms
P2: 0.047844 s \approx 0.048 s = 48 ms
P3: 0.047654 s \approx 0.048 s = 48 ms
- Οι τιμές διαφωνούν ελάχιστα (για το τρίτο πακέτο και στη μέση τιμή) με τις αντίστοιχες του παραθύρου εντολών

Άσκηση 2:

*Στον προσωπικό υπολογιστή

2.1: ping 192.168.2.1 -n 5 -4, ping 192.168.2.2 -n 5 -4, ping 127.0.0.1 -n 5 -4

2.2: 5

2.3: To Default Gateway

2.4: Όχι, γιατί τα πακέτα αυτά τοποθετούνται στην ουρά εισόδου και δεν φεύγουν προς το τοπικό δίκτυο

2.5: Όχι, καθώς αυτά στέλνονται κατευθείαν στον οδηγό loopback

2.6: Όταν κάνουμε ping στη διεπαφή του υπολογιστή, το πακέτο περνάει από τον οδηγό Ethernet πριν μπει στον οδηγό Loopback, ενώ όταν κάνουμε ping στη διεύθυνση Loopback, το πακέτο μπαίνει κατευθείαν στον οδηγό loopback

2.7: Όλα τα πακέτα που στέλνονται στο Netflix είναι timed out (100% loss), ενώ όλα τα πακέτα που στέλνονται στην Amazon δέχονται reply (0% loss). Αυτό μπορεί να οφείλεται σε κάποιο firewall που έχει το Netflix ώστε να μπλοκάρει τα ICMP μηνύματα

Άσκηση 3:

*Στον προσωπικό υπολογιστή

3.1: host 147.102.40.15

3.2: ip.src_host == 192.168.2.2

3.3: Version 4 Bits

Header Length 4 Bits

Differentiated Services Field 8 Bits

Total Length 16 Bits

Identification 16 Bits

Flags 3 Bits

Fragment Offset 13 Bits

Time to Live 8 Bits

Protocol 8 Bits

Header Checksum 16 Bits

Source Address 32 Bits

Destination Address 32 Bits

3.4: Identification, Total Length και Header Checksum

3.5: Ναι

3.6: Μικρότερο Total Length = 40 bytes

μέγιστο Total Length = 66 bytes

3.7: 0x00 → Default Forwarding, Standard Service Class

3.8: Από το πρώτο έως το τελευταίο IPv4 πακέτο που στέλνουμε, η τιμή του Identification αυξάνεται κατά 1 για κάθε πακέτο

3.9: 1 Set

3.10: 0

3.11: 6 → TCP

3.12: Το Header Checksum αλλάζει από πακέτο σε πακέτο επειδή αλλάζουν τα πεδία Identification και Total Length

Άσκηση 4:

*Στον υπολογιστή του εργαστηρίου

4.1: ping <A.B.C.D> -n 1 -4 -f -l size

4.2: 1472

4.3: 1473

4.4: not broadcast and not multicast

4.5: ip and ip.addr == 147.102.38.11

4.6: Όχι, γιατί το μέγεθος του πακέτου είναι μεγαλύτερο από την τιμή της MTU και δεν προωθείται (100% loss)

4.7: Για τη μέγιστη τιμή δεδομένων 1472 bytes καταγράφεται πακέτο μεγέθους 1514 bytes. Άρα MTU = 1514 bytes

4.8: $\text{dataICMP} = \text{maxIP} - \text{headerIP} - \text{headerICMP} = 65535 - 20 - 8 = 65507$

4.9: Όχι, επιτυγχάνει για 65500 bytes

4.10: 65528 bytes

4.11: Όχι, έχει υποστεί θρυμματισμό

4.12: 5 πακέτα με τα πρώτα 4 να έχουν 1480 bytes δεδομένων (όριο πακέτου) και το 5ο πακέτο έχει 80 bytes δεδομένων

4.13: Identification: Request → 0xcff4 (53236),
Reply → 0x0907 (2311)

Don't Fragment Bit: 0 Not set

More Fragments Bit: Πρώτα 4 πακέτα 1 Set, 5ο 0 Not Set

Fragment Offset: 0 → 1480 → 2960 → 4440 → 5920

4.14: Flags: 001. → More fragments: Set

4.15: Fragment Offset: 0

4.16: 1480 bytes δεδομένων, 1514 bytes μήκος πακέτου

4.17: Fragment Offset: 1480

4.18: Ναι, από το πεδίο More Fragments: Set

4.19: Fragment Offset, Total Length και More Fragments

4.20: Version, Header Length, Differentiated Services Field,
Identification, Reserved Bit, Don't Fragment, Time to Live,
Protocol, Source/Destination Address

*Header Checksum [validation disabled]

4.21: Το Fragment Offset του εκάστοτε θραύσματος ισούται με το άθροισμα των μηκών των δεδομένων των προηγούμενων θραυσμάτων

4.22: Τα 4 πρώτα πακέτα μεταφέρουν 1480 bytes δεδομένων + 20 bytes IP Header + 14 bytes Ethernet Header = 1514 bytes πακέτου. Το τελευταίο θραύσμα μεταφέρει 80 bytes δεδομένων + 8 bytes ICMP Header + 20 bytes IP Header + 14 bytes Ethernet Header = 122 bytes πακέτου.