

Εργαστηριακή Άσκηση 4

Πρωτόκολλο IPv4 και Θρυμματισμός

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΚΟΥΣΤΕΝΗΣ ΧΡΙΣΤΟΣ (03120227)

ΟΜΑΔΑ: 3

ΟΝΟΜΑ PC/ΛΣ: LAPTOP-TK5Q3T95 / WINDOWS 11

ΗΜΕΡΟΜΗΝΙΑ:

ΔΙΕΥΘΥΝΣΗ IP: 147.102.239.26(PC LAB) / 192.168.1.14(HOME NETWORK) / 147.102.136.45(NTUA VPN)

ΔΙΕΥΘΥΝΣΗ MAC: B4-B5-B6-79-4B-09

1 – Μετρήστε την καθυστέρηση

Έγιναν στο οικιακό δίκτυο.

1.1 Ποια η ακριβής σύνταξη της εντολής ping που χρησιμοποιήσατε;

ping www.mit.edu -n 3 -4

Με βάση τα αποτελέσματα από την εκτέλεση της εντολής στο παράθυρο εντολών καταγράψτε:

Pinging e9566.dscb.akamaiedge.net [104.96.133.24] with 32 bytes of data:

Reply from 104.96.133.24: bytes=32 time=30ms TTL=56

Reply from 104.96.133.24: bytes=32 time=27ms TTL=56

Reply from 104.96.133.24: bytes=32 time=27ms TTL=56

Ping statistics for 104.96.133.24:

Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 27ms, Maximum = 30ms, Average = 28ms

1.2 Το ποσοστό απωλειών πακέτων και τη μέση καθυστέρηση.

Μέση καθυστέρηση(Average) = 28 ms.

Lost = 0 (0% loss),

1.3 Τις τιμές του RTT.

Approximate round trip times in milli-seconds:

Minimum = 27ms, Maximum = 30ms, Average = 28ms

1.4 Τι έχει αλλάξει σε σχέση με την καταγραφή του παρελθόντος;

Έχουμε μεγαλύτερες ταχύτητες άρα τα RRT times είναι μικρότερα. Επίσης, παρατηρούμε μικρότερη τιμή TTL και διαφορετική τιμή στη διεύθυνση του εξυπηρετητή.

104.96.133.24(new) - 18.7.22.83(old)

Παρατηρώντας την καταγεγραμμένη κίνηση στο Wireshark απαντήστε στα παρακάτω ερωτήματα:

1.5 Ποια η σημασία του φίλτρου σύλληψης που εφαρμόσατε; [Υπόδειξη: Ανατρέξτε στην ιστοσελίδα παραδειγμάτων για φίλτρα σύλληψης <https://wiki.wireshark.org/CaptureFilters>].

Δεν καταγράφονται multicast και broadcast πακέτα αλλά μόνο unicast με συγκεκριμένο παραλήπτη και αποστολέα.

1.6 Ποιο φίλτρο απεικόνισης (Display Filter) πρέπει να εφαρμόσετε προκειμένου να παρατηρείτε μόνο πακέτα IPv4;

ip

1.7 Ποιο φίλτρο απεικόνισης πρέπει να εφαρμόσετε προκειμένου να παρατηρείτε μόνο την κίνηση ICMP που προκάλεσε η εντολή ping;

icmp and ip.addr = 147.102.239.26

1.8 Τι είδος μηνυμάτων ICMP στάλθηκαν από τον υπολογιστή σας κατά την εκτέλεση της εντολής ping; ping requests

1.9 Ποιες οι διευθύνσεις IPv4 πηγής και προορισμού των παραπάνω μηνυμάτων;

Requests

- Source : 147.102.239.26
- Destination : 104.96.133.24

1.10 Τι είδος μηνυμάτων ICMP ελήφθησαν από τον υπολογιστή σας κατά την εκτέλεση της εντολής ping; ping replies

1.11 Ποιες οι διευθύνσεις IPv4 πηγής και προορισμού των παραπάνω μηνυμάτων;

Replies

- Source : 104.96.133.24
- Destination : 147.102.239.26

1.12 Να καταγραφούν οι τιμές του RTT για κάθε ζεύγος Echo Request – Echo Reply, με τη βοήθεια του Wireshark. Συμφωνούν με τις αντίστοιχες στο παράθυρο εντολών [Υπόδειξη: Από το μενού View -> Time Display Format επιλέξτε Seconds Since Previous Displayed Packet.]

Ναι συμφωνούν μέχρι το 3^ο δεκαδικό ψηφίο σε μονάδα μέτρησης second.

2 – Περισσότερα για το Ping

Έγιναν στο PC lab.

2.1 Ποια είναι η ακριβής σύνταξη της εντολής ping που χρησιμοποιήσατε;

ping <address> -n 5 -4

2.2 Πόσα από τα μηνύματα ICMP Echo request που έχουν αποσταλεί από τον υπολογιστή σας έχει καταγράψει το Wireshark;

No.	Time	Source	Destination	Protocol	Info
10	0.000000	147.102.239.26	147.102.236.200	ICMP	Echo (ping) request id=0x0001, seq=298/10753, ttl=128 (reply in 11)
11	0.001855	147.102.236.200	147.102.239.26	ICMP	Echo (ping) reply id=0x0001, seq=298/10753, ttl=255 (request in 10)
12	1.010743	147.102.239.26	147.102.236.200	ICMP	Echo (ping) request id=0x0001, seq=299/11009, ttl=128 (reply in 13)
13	0.002133	147.102.236.200	147.102.239.26	ICMP	Echo (ping) reply id=0x0001, seq=299/11009, ttl=255 (request in 12)
14	1.013591	147.102.239.26	147.102.236.200	ICMP	Echo (ping) request id=0x0001, seq=300/11265, ttl=128 (reply in 15)
15	0.002198	147.102.236.200	147.102.239.26	ICMP	Echo (ping) reply id=0x0001, seq=300/11265, ttl=255 (request in 14)
17	1.012972	147.102.239.26	147.102.236.200	ICMP	Echo (ping) request id=0x0001, seq=301/11521, ttl=128 (reply in 18)
18	0.003468	147.102.236.200	147.102.239.26	ICMP	Echo (ping) reply id=0x0001, seq=301/11521, ttl=255 (request in 17)
21	1.012522	147.102.239.26	147.102.236.200	ICMP	Echo (ping) request id=0x0001, seq=302/11777, ttl=128 (reply in 22)
22	0.003681	147.102.236.200	147.102.239.26	ICMP	Echo (ping) reply id=0x0001, seq=302/11777, ttl=255 (request in 21)

5 από τα $3 \times 5 = 15$ μηνύματα request.

2.3 Ποιος ήταν ο προορισμός τους;

Το default gateway στο δίκτυο του PC lab.

2.4 Παρατηρήσατε αποστολή μηνυμάτων ICMP Echo request στο δίκτυο με πηγή και προορισμό τη διεύθυνση IPv4 του υπολογιστή σας; Εξηγήστε.

Όχι, γιατί αυτά ICMP μηνύματα εισέρχονται στον οδηγό ethernet σύντομα όμως γίνεται αντιληπτό ότι κατευθύνονται στον ίδιο τον αποστολέα και έτσι μεταβιβάζονται άμεσα στο οδηγό loopback.

2.5 Παρατηρήσατε αποστολή μηνυμάτων ICMP Echo request προς τη διεύθυνση του βρόχου επιστροφής; Εξηγήστε.

Όχι, γιατί σύμφωνα με το σχήμα αυτά τα ICMP μηνύματα πηγαίνουν απευθείας στον οδηγό loopback και δεν εισέρχονται στο τοπικό δίκτυο ώστε να τα εντοπίσει το Wireshark.

2.6 Ποια η διαφορά όταν κάνετε ping στη διεπαφή του υπολογιστή σε σχέση με ping στη διεύθυνση loopback αυτού 127.0.0.1; [Υπόδειξη: Η απάντηση σχετίζεται με το ρόλο του βρόχου επιστροφής σε ένα δικτυωμένο σταθμό εργασίας.]

Με το ping στη διεπαφή του υπολογιστή το πακέτο εισέρχεται πρώτα στον οδηγό Ethernet και μετά στον οδηγό loopback. Με ping στη διεύθυνση loopback αυτού 127.0.0.1 το πακέτο εισέρχεται κατευθείαν στον οδηγό loopback (ώστε να πάει στη συνέχεια στην είσοδο πακέτων IPv4).

Ανοίξτε τον φυλλομετρητή της αρεσκείας σας και επισκεφτείτε την ιστοσελίδα της Netflix (<https://www.netflix.com>). Μόλις η σελίδα φορτωθεί πλήρως, χρησιμοποιήστε την εντολή ping ώστε να παράγονται πακέτα IPv4/ICMP με προορισμό τον εξυπηρετητή www.netflix.com. Στη συνέχεια επισκεφτείτε την ιστοσελίδα της Amazon (<https://www.amazon.com>). Μόλις η σελίδα φορτωθεί πλήρως, χρησιμοποιήστε όπως πριν την εντολή ping με προορισμό τον εξυπηρετητή www.amazon.com.

2.7 Τι παράδοξο παρατηρείτε και τι μπορείτε να υποθέσετε για να το εξηγήσετε;

```

C:\Windows\System32>ping www.netflix.com -n 5 -4

Pinging apiproxy-website-nlb-prod-2-22bf9dee8ebc92ff.elb.us-east-1.amazonaws.com [54.237.226.164] with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 54.237.226.164:
    Packets: Sent = 5, Received = 0, Lost = 5 (100% loss),

C:\Windows\System32>ping www.amazon.com -n 5 -4

Pinging d3ag4hukkh62yn.cloudfront.net [52.85.155.200] with 32 bytes of data:
Reply from 52.85.155.200: bytes=32 time=253ms TTL=247
Reply from 52.85.155.200: bytes=32 time=18ms TTL=247
Reply from 52.85.155.200: bytes=32 time=21ms TTL=247
Reply from 52.85.155.200: bytes=32 time=20ms TTL=247
Reply from 52.85.155.200: bytes=32 time=19ms TTL=247

Ping statistics for 52.85.155.200:
    Packets: Sent = 5, Received = 5, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 18ms, Maximum = 253ms, Average = 66ms

```

Παρατηρούμε ότι η αποστολή requests στον εξυπηρετητή netflix αποτυγχάνει ενώ σε αυτόν του amazon επιτυγχάνει. Πιθανόν, το netflix έχει απενεργοποιήσει τα ping replies για λόγους ασφαλείας (ping flooding).

3 – Επικεφαλίδες IPv4

```

ftp> ls
200 PORT command successful
150 Opening ASCII mode data connection for file list
FreeBSD10.4.ova
PCATTCP.exe
lab6.cap
router.ova
FreeBSD.ova
firewall.ova
MagicAdb.exe
Asterisk.ova
TDIQ.exe
MacAddr2.exe
putty.exe
FreeBSD11.3.ova
psftp.exe
pcattcp.pcap
icmpv6.pcap
DMZ.ova
FreeBSD11.4.ova
1984.txt
FreeBSD12.3.ova
R1-BSD11.4-FRR7.ova
Asterisk18.ova
FreeBSD12.4.ova
SecurityLab.rar
226 Transfer complete
ftp: 324 bytes received in 0.02Seconds 20.25Kbytes/sec.

```

3.1 Ποια είναι η σύνταξη του φίλτρου σύλληψης που χρησιμοποιήσατε;
host 147.102.40.15

3.2 Εφαρμόστε φίλτρο απεικόνισης ώστε να παραμείνουν μόνο τα πακέτα IPv4 που έστειλε ο υπολογιστής σας. Ποια είναι η σύνταξή του;

Η σύνταξη του ζητούμενου φίλτρου είναι η ακόλουθη: ***ip and ip.src == 192.168.1.14***

Με κλικ στο σύμβολο '>' στα αριστερά της επικεφαλίδας Internet Protocol Version 4 (στο παράθυρο με τις λεπτομέρειες) αναπτύξετε τα περιεχόμενά της.

3.3 Καταγράψτε τα ονόματα και το μήκος σε bit των πεδίων της επικεφαλίδας του πακέτου IPv4 και σημειώστε στο σχήμα τις θέσεις τους. Χρησιμοποιώντας το πλήκτρο ↓ (κάτω βέλος) μετακινηθείτε από το πρώτο στο τελευταίο μήνυμα της σειράς πακέτων IPv4 που έστειλε ο υπολογιστής σας.

1. Version(4 bits)
2. Header Length(4 bits)
3. Differentiated Services Field(8 bits)
4. Total Length(16 bits)
5. Identification(16 bits)
6. Flags(3 bits)
7. Fragment Offset(13 bits)
8. Time to Live(8 bits)
9. Protocol(8 bits)
10. Header Checksum(16 bits)
11. Source Address(32 bits)
12. Destination Address(32 bits)

3.4 Ποια πεδία της επικεφαλίδας IPv4 αλλάζουν τιμές;

Τα πεδία της επικεφαλίδας IPv4 που αλλάζουν τιμές είναι:

- Total Length
- Identification
- Header Checksum

3.5 Είναι το μήκος της επικεφαλίδας IPv4 το ίδιο σε όλα τα πακέτα;

Ναι, 20 bytes.

3.6 Ποιο είναι το μικρότερο και ποιο το μεγαλύτερο μήκος πακέτου IPv4 που παρατηρήσατε;

40 bytes το μικρότερο και το μεγαλύτερο 66 bytes.

3.7 Τι τιμή έχει το πεδίο Differentiated Services Field και σε ποια ποιότητα υπηρεσίας αντιστοιχεί; [Υπόδ.

Δείτε https://en.wikipedia.org/wiki/Differentiated_services.]

Πάρνει παντού την τιμή 0x00 και αντιστοιχεί στο standard service class.

3.8 Τι παρατηρείτε για τις τιμές του πεδίου Identification;

Αυξάνονται κατά ένα για κάθε επόμενο πακέτο που γίνεται capture.

3.9 Τι τιμή έχει η σημαία Don't Fragment;

$010_2 = 2_{10}$

3.10 Τι τιμή έχει το πεδίο Fragment Offset;

0

3.11 Τι τιμή έχει το πεδίο Protocol και σε ποιο πρωτόκολλο αντιστοιχεί;
6₁₀ και αντιστοιχεί στο TCP πρωτόκολλο.

3.12 Γιατί σε κάθε πακέτο IPv4 αλλάζει η τιμή του πεδίου Header Checksum;
Γιατί αλλάζουν οι τιμές Total Length και Identification της επικεφαλίδας και άρα αλλάζει και το checksum που χρησιμοποιείται για να επαληθεύσει τις τιμές αυτής.

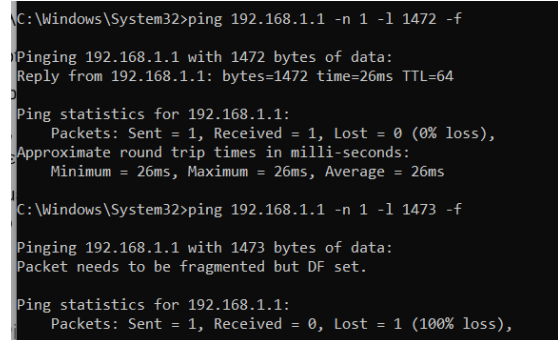
4 – Θρυμματισμός (Fragmentation) στο IPv4

Έγιναν με σύνδεση στο ntua VPN μέσω OpenVPN και ip βλέπε αρχή.

4.1 Ποια είναι η ακριβής σύνταξη της εντολής ping που πρέπει να χρησιμοποιήσετε ώστε να στείλετε χωρίς θρυμματισμό ένα μόνο πακέτο IPv4 που να μεταφέρει μήνυμα ICMP Echo request με συγκεκριμένο μέγεθος δεδομένων; [Υπόδειξη: Αναζητείστε στην τεκμηρίωση της εντολής ping επιλογή για Don't fragment flag ή για ενεργοποίηση της MTU discovery.]

```
ping <address> -n 1 -l <size> -f
```

Εφαρμόζοντας την παραπάνω σύνταξη της εντολής δοκιμάστε στους υπολογιστές του εργαστηρίου διάφορες τιμές για το μέγεθος δεδομένων ICMP στην περιοχή των 1480 byte κάνοντας ping με προορισμό τη διεύθυνση IPv4 κάποιου ενεργού κόμβου στο τοπικό σας δίκτυο



```
C:\Windows\System32>ping 192.168.1.1 -n 1 -l 1472 -f
Pinging 192.168.1.1 with 1472 bytes of data:
Reply from 192.168.1.1: bytes=1472 time=26ms TTL=64
Ping statistics for 192.168.1.1:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 26ms, Maximum = 26ms, Average = 26ms
C:\Windows\System32>ping 192.168.1.1 -n 1 -l 1473 -f
Pinging 192.168.1.1 with 1473 bytes of data:
Packet needs to be fragmented but DF set.
Ping statistics for 192.168.1.1:
    Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
```

4.2 Ποια είναι η μέγιστη τιμή για την οποία επιτυγχάνει η αποστολή;
Η μέγιστη τιμή για την οποία επιτυγχάνεται η αποστολή χωρίς θρυμματισμό είναι 1472 bytes.

4.3 Ποια η μικρότερη τιμή για την οποία απαιτείται θρυμματισμός;
Η μικρότερη τιμή για την οποία απαιτείται θρυμματισμός είναι 1473 bytes.

Στη συνέχεια χρησιμοποιήστε το Wireshark με φίλτρο σύλληψης ώστε να καταγράφονται μόνο πλαίσια μονο-εκπομπής (unicast) για να παρατηρήσετε τι ακριβώς συμβαίνει. Επαναλάβετε τα ping με απαίτηση μη θρυμματισμού για τις δύο τιμές που προσδιορίσατε προηγουμένως στα ερωτήματα 4.2 και 4.3, αντίστοιχα. Μόλις ολοκληρωθεί η καταγραφή, εφαρμόστε ένα φίλτρο απεικόνισης ώστε να παραμείνουν μόνο πακέτα IPv4 από και προς τη διεύθυνση IPv4 όπου κάνετε ping.

4.4 Γράψτε τη σύνταξη του φίλτρου σύλληψης που χρησιμοποιήσατε. [Υπόδειξη: Συμβουλευτείτε τη σελίδα <https://wiki.wireshark.org/CaptureFilters>]
not broadcast and not multicast

4.5 Γράψτε τη σύνταξη του φίλτρου απεικόνισης που εφαρμόσατε.
ip.addr == 192.168.1.1

4.6 Παράγονται πακέτα IPv4 όταν χρησιμοποιείτε την τιμή της ερώτησης 4.3; Γιατί;
Όχι, δεν παράγονται γιατί απορρίπτονται λόγω του ότι είναι μεγαλύτερα από το MTU του τοπικού μας δικτύου.

4.7 Ποιο είναι το μέγεθος της MTU της διεπαφής του υπολογιστή σας; Αιτιολογήστε.
Η MTU της διεπαφής του υπολογιστή μου έχει μέγεθος 1500 bytes όπως στα περισσότερα windows PCs αφού το total length του ipv4 packet ισούται με 1500 bytes για Data = 1472 bytes + ICMP(= 8 bytes) + IPv4 Header(= 20 bytes) είναι 1500 bytes.

Εναλλακτικά, τρέχουμε την εντολή:

netsh interface ipv4 show subinterfaces

4.8 Ποια τιμή του μεγέθους δεδομένων ICMP οδηγεί σε πακέτο IPv4 μέγιστου μήκους;
Maximum packet size = 65535 bytes.

ICMP = 8 bytes.

IPv4 header = 20 bytes.

Data maximum size = 65535 - 20 - 8 = 65507 bytes.

4.9 Για την προηγούμενη τιμή μεγέθους δεδομένων ICMP και με απαίτηση μη θρυμματισμού, επιτυγχάνει το ping προς τη διεύθυνση IPv4 του υπολογιστή σας; Εάν όχι, ποια είναι η μέγιστη τιμή για την οποία είναι επιτυχής;

Όχι, δεν επιτυγχάνεται. Το maximum data size για το οποίο επιτυγχάνεται είναι 1472 bytes.

4.10 Τι μέγεθος έχει το μεγαλύτερο πακέτο IPv4 που μπορεί να παράγει η εντολή ping;

Κατόπιν με τα ίδια φίλτρα σύλληψης και απεικόνισης ξεκινήστε μια καταγραφή και κάντε ping προς προορισμό εντός του τοπικού σας δικτύου στέλνοντας ένα μόνο μήνυμα ICMP με μέγεθος δεδομένων 6.000, χωρίς την απαίτηση μη θρυμματισμού του πακέτου IPv4.

4.11 Βρείτε το πρώτο μήνυμα ICMP Echo Request που έστειλε ο υπολογιστής σας. Έχει μεταφερθεί μήνυμα αυτό ως ένα πακέτο IPv4;

Όχι, έχει θρυμματιστεί.

4.12 Εάν όχι, πόσα πακέτα IPv4 χρειάστηκαν και γιατί;

Έχει μεταφερθεί ως 5 πακέτα IPv4 αφού $6000/4 = 1500$ και το MTU του τοπικού δικτύου επιτρέπει το πολύ 1472 bytes δεδομένων.

4.13 Για καθένα από αυτά τα πακέτα IPv4, καταγράψτε τις τιμές των πεδίων της επικεφαλίδας που σχετίζονται με τον θρυμματισμό (Identification, Don't Fragment Bit, More Fragments Bit, Fragment Offset).

1. Packet_1
 - a. 0xcef0
 - b. 0
 - c. 1
 - d. 0
2. Packet_2
 - a. 0xcef0
 - b. 0
 - c. 1
 - d. 1480
3. Packet_3
 - a. 0xcef0
 - b. 0
 - c. 1
 - d. 2960
4. Packet_4
 - a. 0xcef0
 - b. 0
 - c. 1
 - d. 4440
5. Packet_5
 - a. 0xcef0
 - b. 0
 - c. 0
 - d. 5920

4.14 Επιλέξτε το πρώτο από τα παραπάνω πακέτα IPv4 (το πρώτο θραύσμα). Ποια πληροφορία της επικεφαλίδας IPv4 δηλώνει ότι το πακέτο έχει θρυμματιστεί;

Το More fragments bit που είναι ίσο με 1.

4.15 Ποια πληροφορία της επικεφαλίδας IPv4 δηλώνει ότι αυτό είναι το πρώτο θραύσμα και όχι ένα μεταγενέστερο;

Το `fragment offset` που είναι ίσο με 0.

4.16 Ποιο είναι το μήκος του πρώτου θραύσματος;

1514 bytes = `ethernet_frame`

Και $1514 - 14(\text{Ethernet header}) - 20(\text{IPv4 header}) = 1480$ το `payload` του IPv4 packet.

4.17 Επιλέξτε το δεύτερο από τα παραπάνω πακέτα IPv4 (το δεύτερο θραύσμα). Ποια πληροφορία της επικεφαλίδας IPv4 δηλώνει ότι δεν είναι το πρώτο θραύσμα;

Fragment Offset: 1480

4.18 Ακολουθούν άλλα θραύσματα; Πώς το αναγνωρίζετε από τις πληροφορίες της επικεφαλίδας;

Ναι. Φαίνεται από το `More fragments bit` που είναι ίσο με 1.

4.19 Ποια πεδία της επικεφαλίδας IPv4 αλλάζουν μεταξύ των θραυσμάτων;

- Fragment offset
- More fragments(0 μόνο στο τελευταίο θραύσμα)
- Header checksum

4.20 Ποια πεδία της επικεφαλίδας IPv4 δεν αλλάζουν μεταξύ των θραυσμάτων;

- Identification
- Source address
- Destination address
- Don't fragment

4.21 Ποια η σχέση της τιμής του πεδίου Fragment Offset και του μήκους των θραυσμάτων που προηγήθηκαν;

$\text{Fragment offset} = n * 1480$ όπου n η σειρά του πακέτου 1^ο ,2^ο ,3^ο κ.ο.κ.

4.22 Δικαιολογήστε το μήκος του πακέτου IPv4 που μεταφέρει το τελευταίο θραύσμα που στάλθηκε.

Τα προηγούμενα packets είχαν μήκος IPv4 packet size = 1500 bytes.

Έχουμε $1480 * 4 = 5920$ bytes of IPv4 dataload.

$6008 - 5920 = 88$ bytes of dataload remaining + IPv4 header έχουμε Total length τελευταίου πακέτου ίσο με 108.