Ονοματεπώνυμο: Μάρκος Δεληγιάννης		Ομάδα: 3
Όνομα PC/ΛΣ: LAPTOP-THP9FHNU / W11	Ημερομηνία: 23 / 11 / 22	
Διεύθυνση IP: 192.168.1.9	Διεύθυνση MAC: 00 – 45 – E2 – A3 – 54 – 95	

Εργαστηριακή Άσκηση 7
Πρωτόκολλα TCP και UDP

Απαντήστε στα ερωτήματα στον χώρο που σας δίνεται παρακάτω και στην πίσω σελίδα εάν δεν επαρκεί. Το φυλλάδιο αυτό θα παραδοθεί στον επιβλέποντα.

1

1.1 "host 192.168.1.9"

1.2 "ip.dst == 1.1.1.1 or ip.dst == 2.2.2.2 or ip.dst ==147.102.40.1"

1.3 Στην θύρα 23, η οποία αντιστοιχεί στο telnet.

1.4 "tcp.port == 23"

1.5 Η σημαία "Syn".

1.6 Και στις δύο περιπτώσεις ο υπολογιστής μας κάνει 5 προσπάθειες.

1.7 Η χρονική απόσταση είναι 1, 2, 4 και 8 δευτερόλεπτα, δηλαδή διπλασιάζεται με κάθε αποτυχημένη προσπάθεια. Αυτό συμβαίνει και στις δύο περιπτώσεις.

.....

1.8 Παρατηρούμε ότι οι περιπτώσεις Α και Β ταυτίζονται ως προς τα παρατηρούμενα αποτελέσματα. Πιο συγκεκριμένα και στις δύο περιπτώσεις ο υπολογιστής μας σταματά να προσπαθεί μετά από 5 αποτυχημένες προσπάθειες, χωρίς να έχει λάβει καμία απάντηση από τον προορισμό. Επιπλέον, και στις δύο περιπτώσεις έχουμε την ίδια χρονική απόσταση μεταξύ των τεμαχίων που αποστέλλουμε.

1.9 Μόνο το πρώτο, στο οποίο ο υπολογιστής μας αποστέλει τεμάχιο SYN.

1.10 Ο υπολογιστής μας εγκαταλείπει μετά από πολλές αποτυχημένες προσπάθειες.

1.11 "tcp and ip.addr == 147.102.40.1"

1.12 5 προσπάθειες.

1.13 Στην περίπτωση Γ λαμβάνουμε ως απάντηση ένα τεμάχιο RST, ACK μετά από κάθε τεμάχιο SYN. Επιπλέον, τα τεμάχια SYN τώρα αποστέλλονται κάθε περίπου 0.5 δευτερόλεπτα, αντί για 1,2,4 και 8.

1.14 Τις σημαίες Accurate ECN, Congestion Window Reduced, ECN-Echo, Urgent, Acknowledgment, Push, Reset, Syn και Fin.

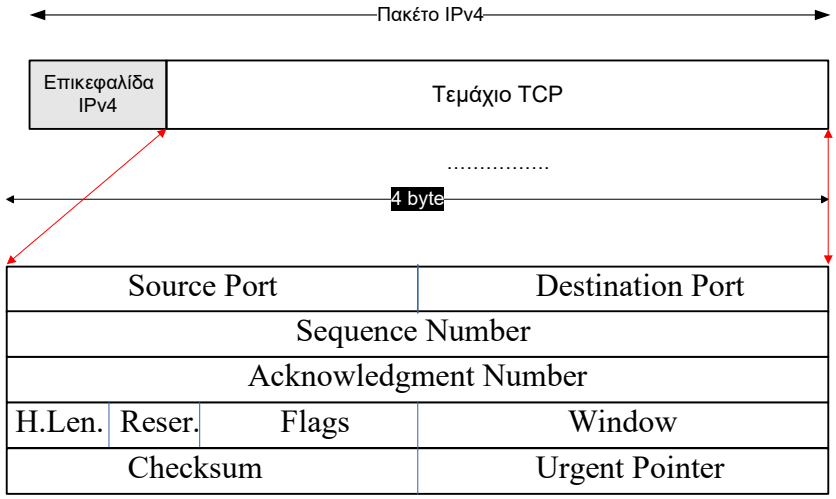
1.15 Η σημαία Reset.

1.16 Το μέγεθος της επικεφαλίδας είναι 20 bytes. Το μέγεθος του πεδίου δεδομένων είναι 0.

(*Τα τελευταία 6 bytes που είναι 0 αντιστοιχούν σε padding του πλαισίου Ethernet)*

1.17

1) Source Port	(2 bytes)	2) Destination Port	(2 bytes)
3) Sequence Number	(4 bytes)	4) Acknowledgement Number	(4 bytes)
5) Header Length	(4 bit)	6) Reserved	(3 bit)
7) Flags	(9 bit)	8) Window	(2 bytes)
9) Checksum	(2 bytes)	10) Urgent Pointer	(2 bytes)



1.18 Το όνομα του πεδίου σύμφωνα με τη δοσμένη ιστοσελίδα είναι **Data Offset**. Αυτό το πεδίο σύμφωνα με το Wireshark λέγεται **Header Length**.

1.19 Η τιμή του πεδίου Data offset αντιστοιχεί στο μήκος της επικεφαλίδας σε **32-bit λέξεις (= 4 bytes)**. Συνεπώς αρκεί να πολλαπλασιάσουμε επί 4 για να λάβουμε το μήκος σε bytes (εν προκειμένω 5*4=20).

1.20 Δεν υπάρχει τέτοιο πεδίο.

1.21 TCP Segment Length = IPv4 Total Length – IPv4 Header Length. Ο λόγος που μπορούμε να βρούμε το μέγεθος όπως αναφέραμε είναι ότι το TCP segment ενθυλακώνεται **ολόκληρο** και **μόνο αυτό** σε ένα πακέτο IPv4.

1.22 Το μέγεθος είναι 28 bytes.

1.23 Υπάρχει διαφορά: Το τεμάχιο SYN που στέλνουμε έχει επιπλέον το πεδίο Options (8 bytes), με παραμέτρους για τη διαπραγμάτευση, ενώ το RST, ACK δεν έχει καθόλου Options.

2

2.1 “tcp”

2.2 Στη θύρα 21.

2.3 Με τη θύρα 20.

2.4 “tcp.port == 21”

2.5 3 τεμάχια: Ένα SYN από εμάς, ένα SYN,ACK από τον server και ένα ACK από εμάς.

2.6 2 σημαίες: Η σημαία SYN και η σημαία ACK.

2.7 Τεμάχια SYN και SYN,ACK: 28 bytes Τεμάχιο ACK: 20 bytes.

2.8 Μηδέν και στα τρία τεμάχια.

2.9 11.183ms.

2.10 Συμφωνεί απολύτως.

2.11 Υπολογιστής μας: 0 (σχετικός) 2898239371 (απόλυτος)
Εξυπηρετητής: 0 (σχετικός) 646217266 (απόλυτος)

2.12 Ο αριθμός επιβεβαίωσης είναι 1, αφού ο εξυπηρετητής ζητά το πρώτο μας τεμάχιο.

2.13 Αριθμός επιβεβαίωσης = 1 (όπως στο 2.12)
Αριθμός σειράς = 1 = Αριθμός επιβεβαίωσης προηγούμενου πακέτου

2.14 Το μήκος δεδομένων είναι 0 και για τα τρία τεμάχια.

2.15 Αφού κωδικοποιούνται με 32 bits έχουμε μέγιστη τιμή: $2^{32}-1=4,294,967,295$

2.16 "((tcp.ack==1 and tcp.seq==1) or tcp.seq==0) and tcp.len==0"

2.17 Μέγεθος παραθύρου = 8192.

2.18 65535

2.19 Στο πεδίο Window.

2.20 Ο υπολογιστής μας ανακοινώνει: 0 (πολλαπλασιασμός επί 1)
Ο εξυπηρετητής ανακοινώνει: 6 (πολλαπλασιασμός επί 64)

2.21 Στο πεδίο Options και πιο συγκεκριμένα στο TCP Option – Window scale.

2.22 1460.

2.23 Γνωρίζουμε ότι η MTU της διεπαφής του υπολογιστή μας είναι 1500, οπότε (δεδομένου ότι χρησιμοποιείται IPv4) MSS = MTU – 40 = 1460.

2.24 Στο πεδίο Options και πιο συγκεκριμένα στο TCP Option – Maximum segment size.

2.25 MSS = 536

2.26 Πλήρως αντίστοιχα με το 2.23 είναι MSS = MTU – 40 = 576 – 40 = 536.

2.27 Το μέγιστο μήκος δεδομένων του τεμαχίου TCP είναι το μικρότερο από τα MSS, δηλαδή 536 bytes. Υπολογίζοντας επικεφαλίδα TCP μεγέθους 20 bytes έχουμε μέγιστο συνολικό μήκος του τεμαχίου ίσο με 556 bytes.

2.28 Η σημαία FIN.

2.29 "tcp.flags.fin==1"

2.30 Ο εξυπηρετητής.

2.31 4 τεμάχια συνολικά.

2.32 32 bytes για τη σύνδεση δεδομένων και 20 bytes για τη σύνδεση ελέγχου.

2.33 Το TCP τεμάχιο FIN που έστειλε ο εξυπηρετητής για τον τερματισμό της σύνδεσης δεδομένων φέρει 132 bytes δεδομένων ftp. Τα υπόλοιπα έχουν μήκος δεδομένων 0.

2.34 Σύνδεση δεδομένων: 52 bytes = 20 bytes (επικεφαλίδα IPv4) + 20 bytes (υποχρεωτικά επικεφαλίδας TCP) + 12 bytes (επιλογές / timestamps) + 0 bytes (δεδομένα).
Σύνδεση ελέγχου: 40 bytes (δεν περιλαμβάνονται timestamps).

2.35 Σύνδεση δεδομένων: 40 bytes (όπως στο 2.34)
Σύνδεση ελέγχου: 40 bytes (όπως στο 2.34)

2.36 Από εμάς: 377
Από τον εξυπηρετητή: 126

2.37 Θεωρούμε τα τελευταία τεμάχια που έστειλε κάθε πλευρά (από/προς τη θύρα 21) και εστιάζουμε στο relative SN τους. Προσθέτουμε τα δεδομένα του τελευταίου τμήματος (εδώ 0) και λαμβάνουμε το αποτέλεσμα.

2.38 "tcp.port == 20"

2.39 Υπολογιστής μας: 1460 Εξυπηρετητής: 536

2.40 MSS εξυπηρετητή + TCP header length = 1480.

2.41 1.564 ms

2.42 Όχι, στέλνει για μερικά από τα πακέτα που λαμβάνει, αφού τα ACK δρουν συσσωρευτικά.

2.43 118 τεμάχια με δεδομένα.

2.44 24 τεμάχια.

2.45 1048832.

2.46 Δεν είναι ίδια. Προκύπτει από την τιμή window του τεμαχίου πολλαπλασιασμένη με το window scale που ανακοίνωσε ο υπολογιστής μας στην χειραψία TCP (4097 * 256 = 1048832).

2.47 Όχι, δεν αλλάζει. Η μικρότερη τιμή είναι η αρχική (στην χειραψία), 65535 bytes.

2.48 Ο εξυπηρετητής θα σταματούσε να στέλνει τεμάχια TCP, γνωρίζοντας ότι κανένα δεν θα γίνει αποδεκτό.

2.49 Μέγεθος πλαισίου: 590 bytes Μέγεθος επικ. Ethernet: 14 bytes
Μέγεθος επικ. IP: 20 bytes Μέγεθος επικ. TCP: 32 bytes

2.50 Το μέγεθος των δεδομένων TCP (524) είναι κάτω του MSS, όπως αναμέναμε.

2.51 Το πακέτο θα θρυμματιστεί.

2.52 Προς εμάς: 61442 (AN) + 0 (data τελευταίου τεμαχίου) - 1 = 61441 Από εμάς: 0 (ομοίως)

2.53 Ρυθμός μετ. = Δεδομένα / χρόνο = 61441 bytes / 0.011551 sec ≈ 5319.1 kbyte/sec

2.54 Όχι, δεν υπήρχαν, καθώς δεν παρατηρήσαμε τα ειδικά επισημασμένα πακέτα στο wireshark.

3

3.1 “tcp.port == 20”

3.2 94.65.141.44

3.3 14.674 ms, το οποίο είναι σημαντικά μεγαλύτερο από αυτό του 2.41.

3.4 Τα τεμάχια στέλνονται σε ριπές και με κάθε ριπή στέλνονται εκθετικά περισσότερα πακέτα. Οι ριπές οφείλονται στο ότι ο αποστολέας τακτικά υπερβαίνει το μέγεθος παραθύρου και σταματά την εκπομπή πακέτων. Ο αυξανόμενος αριθμός πακέτων ανα ριπή οφείλεται στον μηχανισμό slow start.

3.5 Έστειλε 4 τεμάχια, σε συμφωνία με το RFC 5681, δεδομένου ότι SMSS = 536 < 1095.

3.6 Δεύτερο διάστημα: 6 τεμάχια
Τρίτο διάστημα: 10 τεμάχια
Τέταρτο διάστημα: 16 τεμάχια

3.7 Πρώτο διάστημα: 1 τεμάχια
Δεύτερο διάστημα: 2 τεμάχια
Τρίτο διάστημα: 3 τεμάχια
Παρατηρούμε ότι σε αντίθεση με το 3.6 τα τεμάχια αυξάνονται κατά ένα.

3.8 Είναι παρόμοιο με το αντίστοιχο που δόθηκε. Η σημαντικότερη διαφορά είναι ότι στην καταγραφή μας παρατηρούνται περισσότερα τεμάχια από τον δρομολογητή ανά RTT (π.χ. το initial window είναι 10, αντί για 4). Αυτό βρίσκεται σε συμφωνία με τον τύπο της παραγράφου 2 του RFC 6928, ο οποίος στην περίπτωσή μας δίνει τιμή 10 τεμαχίων ως αρχικό μέγεθος παραθύρου.

4

4.1 “udp”

4.2 Source Port: 2 bytes Destination Port: 2 bytes
Length: 2 bytes Checksum: 2 bytes

4.3 8 bytes.

4.4 98 bytes (από το πεδίο Payload Length του IPv6 πακέτου στο οποίο ενθυλακώνεται).

4.5 Το μήκος του UDP πακέτου (μαζί με την επικεφαλίδα) σε bytes.

4.6 8 bytes, όσα και το μήκος της επικεφαλίδας του UDP δεδομενογράμματος.

4.7 Ελάχιστο: ελάχιστο Length – μήκος επικεφαλίδας UDP = 8 – 8 = 0 bytes.
Μέγιστο (αγνοώντας το IPv4): μέγιστο Length – μήκος επικεφαλίδας UDP = 2^16 – 8 = 65528 bytes.
Το μέγιστο μέγεθος όμως ενός IPv4 πακέτου είναι 65535 bytes, οπότε το μέγιστο μέγεθος μηνύματος UDP με ενθυλάκωση σε IPv4 είναι 65535 – 20 (IPv4 Header) – 8 (UDP header) = 65508 bytes.

4.8 576 – 20 (IPv4 header) – 8 (UDP header) = 548 bytes.

4.9 Ναι, παρατηρήσαμε μηνύματα του πρωτοκόλλου MDNS.

4.10 “dns”

4.11 fe80::1

4.12 Θύρα προέλευσης: 52834 Θύρα προορισμού: 53

4.13 Θύρα προέλευσης: 53 Θύρα προορισμού: 52834

4.14 Η θύρα 53 αντιστοιχεί στο πρωτόκολλο εφαρμογής DNS.