Εργαστηριακή Άσκηση 7

Πρωτόκολλα TCP και UDP

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΚΟΥΣΤΕΝΗΣ ΧΡΙΣΤΟΣ (03120227)

ΟΜΑΔΑ: 3

ΟΝΟΜΑ PC/ΛΣ: LAPTOP-TK5Q3T95 / WINDOWS 11

ΗΜΕΡΟΜΗΝΙΑ: 21/11/2023

ΔΙΕΥΘΥΝΣΗ IP: 192.168.1.14

ΔΙΕΥΘΥΝΣΗ MAC: B4-B5-B6-79-4B-09

1 Μετάδοση δεδομένων με TCP

```
C:\Users\koust>telnet 1.1.1.1
Connecting To 1.1.1.1...Could not open connection to the host, on port 23: Connect failed

C:\Users\koust>telnet 2.2.2.2
Connecting To 2.2.2.2...Could not open connection to the host, on port 23: Connect failed

C:\Users\koust>telnet 147.102.40.1
Connecting To 147.102.40.1...Could not open connection to the host, on port 23: Connect failed

C:\Users\koust>
```

1.1 Καταγράψτε τη σύνταξη του φίλτρου σύλληψης που χρησιμοποιήσατε ώστε να συλλαμβάνονται μόνο τα πακέτα που περιλαμβάνουν τη διεύθυνση IPv4 του υπολογιστή σας.

Capture filter: ***ip and host 192.168.1.14***

1.2 Εφαρμόστε φίλτρο απεικόνισης ώστε να παραμείνουν μόνο πακέτα προς κάποιον από τους παραπάνω προορισμούς. Ποια είναι η σύνταξή του;

Display filter:

ip.addr == 1.1.1.1 or ip.addr == 2.2.2.2 or ip.addr == 147.102.40.1

1.3 Σε ποια θύρα (του άλλου υπολογιστή) προσπαθεί να συνδεθεί ο δικός σας υπολογιστής;

Προσπαθέι να συνδεθεί στη θύρα 23 που είναι η default θύρα για επικοινωνία μέσω telnet.

1.4 Εφαρμόστε ένα φίλτρο απεικόνισης ώστε να παραμείνουν μόνο τα τεμάχια TCP που σχετίζονται με τη θύρα αυτή. Ποια είναι η σύνταξή του;

Display filter: ***tcp.port == 23***

1.5 Ποια σημαία μήκους 1 bit ενεργοποιείται για την εκκίνηση της εγκατάστασης της σύνδεσης TCP;

Η σημαία / flag 'SYN'

1.6 Πόσες προσπάθειες κάνει ο υπολογιστής σας προκείμενου να εγκαταστήσει σύνδεση TCP στις Περιπτώσεις Α και Β;

Ο υπολογιστής μας κάνει 5 προσπάθειες στις περιπτώσεις Α και Β προκειμένου να εγκαταστήσει τη σύνδεση.

1.7 Καταγράψτε τη χρονική απόσταση μεταξύ των διαδοχικών προσπαθειών εγκατάστασης σύνδεσης.

No.	Time	Source	Destination	Protoc
18	0.000000	192.168.1.14	1.1.1.1	TCP
19	1.003385	192.168.1.14	1.1.1.1	TCP
37	2.014018	192.168.1.14	1.1.1.1	TCP
45	4.001634	192.168.1.14	1.1.1.1	TCP
63	8.003460	192.168.1.14	1.1.1.1	TCP
90	15.978928	192.168.1.14	2.2.2.2	TCP
96	1.011209	192.168.1.14	2.2.2.2	TCP
112	2.002816	192.168.1.14	2.2.2.2	TCP
361	4.013831	192.168.1.14	2.2.2.2	TCP
407	8.000887	192.168.1.14	2.2.2.2	TCP

Περίπτωση Α & Περίπτωση Β

Χρονικές αποστάσεις:

- 1η --> 2η = 1 sec
- 2η --> 3η = 1 sec
- 3η --> 4η = 2 sec
- 4η --> 5η = 4 sec

Δηλαδή έχουμε εκθετική αύξηση του διαστήματος μεταξύ διαδοχικών προσπαθειών.

1.8 Τι παρατηρείτε συγκρίνοντας τα αποτελέσματα των περιπτώσεων Α και Β;

Έχουν όμοια αποτελέσματα με εξαίρεση το Sequence number και Source Port.

1.9 Ποια βήματα της τριπλής χειραψίας παρατηρήσατε;

Παρατηρώ μόνο το πρώτο βήμα όπου SEQ = 0, SYN = 1, ACK = 0

1.10 Ο υπολογιστής σας απολύει τη σύνδεση ή απλώς εγκαταλείπει την προσπάθεια;

Εγκαταλείπει την προσπάθεια γιατι το RST είναι απενεργοποιημένο

1.11 Ποια είναι η σύνταξή του;

Display filter: ***tcp.port == 23 and ip.host == 147.102.40.1***

1.12 Πόσες προσπάθειες κάνει ο υπολογιστής σας προκείμενου να εγκαταστήσει σύνδεση TCP;

Μία.

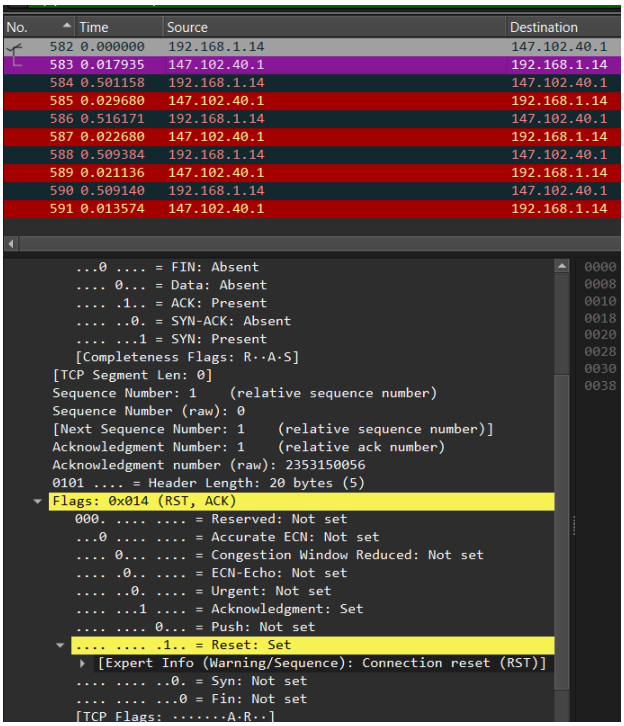
1.13 Συγκρίνοντας με την απάντηση σας στο ερώτημα 1.8, ποιες διαφορές παρατηρείτε;

Η διαφορά με το ερώτημα 1.8 είναι ότι πλέον υπάρχουν και απαντήσεις ACK από το άλλο άκρο, στις οποίες είναι ενεργοποιημένο και το flag RST, δηλαδή της απόρριψης σύνδεσης. Επίσης οι προσπάθειες γίνονται πλέον κάθε 5 δευτερόλεπτα.

1.14 Ποιες σημαίες μήκους 1 bit περιλαμβάνει;

Επιλέγοντας το πακέτο 583 που αποτελεί την πρώτη απάντηση TCP της διεύθυνσης 147.102.40.1 στον υπολογιστή μας εντοπίζουμε τα εξής bit flags:

- Reserved
- Accurate ECN
- Congestion Window Reduced
- ECN-ECHO
- Urgent
- Acknowledgment
- Push
- Reset
- Syn
- Fin



1.15 Ποια εξ αυτών δηλώνει άρνηση της εγκατάστασης σύνδεσης TCP;

Το flag RST που έχει τιμή 1.

1.16 Ποιο είναι το μέγεθος της επικεφαλίδας και ποιο το μέγεθος του πεδίου δεδομένων αυτού του τεμαχίου TCP;

Header Length : 20 bytes

Data : 0 bytes

1.17 Καταγράψτε τα ονόματα και το μήκος σε bit των πεδίων της επικεφαλίδας του τεμαχίου TCP και σημειώστε στο σχήμα τις θέσεις τους.

Source Port: 16 bits(2 bytes)

Destination Port: 16 bits(2 bytes)

Sequence Number: 32 bits(4 bytes)

Acknowledgment Number: 32 bits(4 bytes)

Header Length: 4 bits

Flags: 12 bits

Window: 16 bits

Checksum: 16 bits

Urgent Pointer: 16 bits

Source Port(16 bits)	Destination Port(16 bits)
Sequence Number(32 bits)	
Acknowledgment Number(32 bits)	
Header Length(4 bits)| Flags(12 bits)	Window(16 bits)
Checksum(16 bits)	Urgent Pointer(16 bits)

1.18 Ποιο είναι το όνομα του πεδίου που προσδιορίζει το μέγεθος της επικεφαλίδας TCP σύμφωνα με την ιστοσελίδα http://www.networksorcery.com/enp/protocol/tcp.htm που θα αναζητήσετε στο Internet Archive; Ποιο όνομα χρησιμοποιεί το Wireshark για το πεδίο αυτό της επικεφαλίδας TCP στο παράθυρο με τις λεπτομέρειες του επιλεγμένου πακέτου;

Σύμφωνα με την ιστοσελίδα, το μέγεθος της επικεφαλίδας TCP προσδιορίζεται από το πεδίο Data Offset το οποίο όπως λέει μας δίνει την τιμή σε λέξεις των 32 bits. Αντιθέτως, στο Wireshark το πεδίο αυτό ονομάζεται Header Length και για το συγκεκριμένο πακέτο έχει τιμή $0101_2 = 5_{10}$, το οποίο μας δίνει 20 bytes.

1.19 Πώς προκύπτει το μήκος της επικεφαλίδας TCP από την τιμή που παρατηρείτε στα περιεχόμενα πακέτου σε δεκαεξαδική τιμή;

Το Header length πεδίο του τεμαχίου παρατηρούμε από τα δεδομένα ότι έχει τιμή 5_{16}.
Άρα, 5 * 4 = 20 bytes.

1.20 Υπάρχει πεδίο της επικεφαλίδας TCP που να δηλώνει το μήκος του τεμαχίου;

Όχι, δεν υπάρχει.

1.21 Πώς προκύπτει το μήκος αυτό με βάση τα στοιχεία των επικεφαλίδων IPv4 και TCP;

Το μήκος του τεμαχίου TCP μπορεί να βρεθεί εάν από το συνολικό μήκος τουIPv4 πακέτου (πεδίο Total Length) αφαιρέσουμε το μήκος της IPv4 επικεφαλίδας (πεδίο Header Length). Συγκεκριμένα, 40(Total length) - 20(Header length) = 20 bytes.

1.22 Ποιο είναι το μέγεθος της επικεφαλίδας του πρώτου ή μοναδικού τεμαχίου TCP που στέλνει ο υπολογιστής σας στον 147.102.40.1 για την εγκατάσταση σύνδεσης TCP;

Header Length: 32 bytes ($8_{16} = 1000_2$)

1.23 Υπάρχει διαφορά στο μέγεθος της επικεφαλίδας TCP των δύο παραπάνω τεμαχίων; Εάν ναι, που οφείλεται;
Παρατηρούμε πως υπάρχει διαφορά στο μήκος των παραπάνω 2 τεμαχίων κατά 12 bytes, η οποία και οφείλεται στο πεδίο Options το οποίο δεν υπήρχε στην απάντηση από τον 147.102.40.1, ενώ καταλαμβάνει 12 bytes στο πρώτο TCP τεμάχιο που εμείς αποστέλουμε και περιλαμβάνει τις ρυθμίσεις της αιτούμενης TCP σύνδεσης.

2 Εγκατάσταση σύνδεσης, μεταφορά δεδομένων και απόλυση σύνδεσης TCP

2.1 Ποιο φίλτρο σύλληψης χρησιμοποιήσατε για την καταγραφή της κίνησης;
tcp and ip host 147.102.40.15

Εγκατάσταση σύνδεσης

Παρατηρήστε τα τεμάχια TCP που ανταλλάχθηκαν και εντοπίστε τα σχετικά με την τριπλή χειραψία. Θα βρείτε δύο τριπλές χειραψίες: μία για την εγκατάσταση της σύνδεσης ελέγχου FTP και μία για τη μεταφορά δεδομένων FTP.
2.2 Σε ποια θύρα (ελέγχου FTP) του edu-dy.cn.ntua.gr προσπαθεί να συνδεθεί ο υπολογιστής σας για να αρχίσει η επικοινωνία με τον εξυπηρετητή FTP;
Port 21.

2.3 Με ποια θύρα (δεδομένων FTP) του υπολογιστή edu-dy.cn.ntua.gr γίνεται η σύνδεση για τη μεταφορά δεδομένων (του αρχείου PCATTCP.exe);
Port 20.

Εφαρμόστε ένα φίλτρο απεικόνισης της μορφής tcp.port ώστε να παραμείνουν μόνο τα τεμάχια TCP που σχετίζονται με τη θύρα ελέγχου FTP.

2.4 Ποια είναι η σύνταξη του φίλτρου;
Display filter : tcp.port == 21

2.5 Πόσα τεμάχια TCP ανταλλάσσονται για την εγκατάσταση της σύνδεσης ελέγχου FTP;
Ανταλλάσσονται 3 τεμάχια όπως ήταν αναμενόμενο.

2.6 Ποιες σημαίες χρησιμοποιούνται για την εγκατάσταση της σύνδεσης TCP;

Αρχικά η σημαία SYN στο πρώτο τεμάχιο από εμάς προς τον σέρβερ, στη συνέχεια οι σημαίες SYN και ACK κατά την απόκριση του σέρβερ και τέλος η σημαία ACK από εμάς προς τον σέρβερ, οπότε συνολικά χρησιμοποιούνται οι σημαίες SYN και ACK, όπως προβλέπει το μοντέλο της τριπλής χειραψίας.

2.7 Ποιο είναι το μέγεθος των επικεφαλίδων TCP των τεμαχίων αυτών;

Το μέγεθος των επικεφαλίδων TCP των παραπάνω τεμαχίων είναι 32 bytes για τα πρώτα τεμάχια και 20 bytes για το τελευταίο.

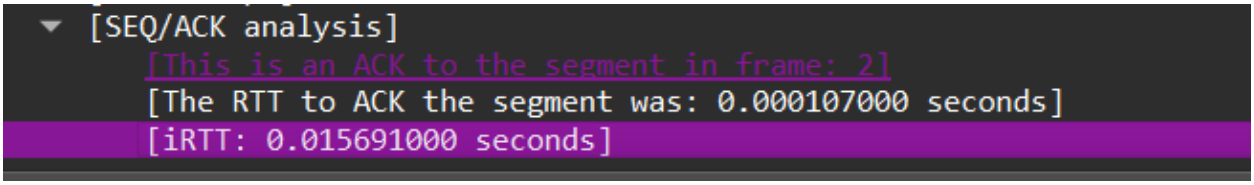
2.8 Ποιο είναι το μέγεθος δεδομένων των τεμαχίων αυτών;

Έχουν μηδενικό μέγεθος δεδομένων αφού κατά την τριπλή χειραψία δεν ανταλλάσσονται δεδομένα.

2.9 Πόσο διαρκεί η διαδικασία της τριπλής χειραψίας;

Με βάση τη χρονική στιγμή άφιξης του 3ου και τελευταίου πακέτου της χειραψίας διαρκεί 0.0157 sec.

2.10 Συμφωνεί η τιμή που βρήκατε προηγουμένως με το iRTT που εμφανίζει το Wireshark κάτω από το [SEQ/ACK analysis] στο παράθυρο με τις λεπτομέρειες επικεφαλίδας;



Με βάση την παραπάνω εικόνα διαπιστώνουμε ότι συμφωνεί.

Κατά την εγκατάσταση της σύνδεσης, ο πελάτης TCP και ο εξυπηρετητής TCP αναγγέλλουν ο ένας στον άλλο τους αύξοντες αριθμούς που θα χρησιμοποιήσουν κατά τη μετάδοση δεδομένων. Το πεδίο Sequence Number (αριθμός σειράς) στην επικεφαλίδα TCP δείχνει τον αύξοντα αριθμό του πρώτου byte στο πεδίο δεδομένων που αποστέλλονται και το πεδίο Acknowledgement Number (αριθμός επιβεβαίωσης) δείχνει τον αριθμό σειράς του επόμενου byte δεδομένων που αναμένεται.

2.11 Ποιοι είναι οι αρχικοί αριθμοί σειράς (Sequence Number) που ανακοινώνει η κάθε πλευρά;

Η πλευρά του υπολογιστή μας ανακοινώνει τους παρακάτω sequence numbers(σχετικό και απόλυτο):

Sequence Number: 0 (relative sequence number)

Sequence Number (raw): 1250482696

Η πλευρά του εξυπηρετητή ανακοινώνει τους παρακάτω αριθμούς sequence numbers(σχετικό και απόλυτο) :

Sequence Number: 0 (relative sequence number)

Sequence Number (raw): 442581172

2.12 Πώς προκύπτει ο αριθμός επιβεβαίωσης (Acknowledgement Number) του τεμαχίου TCP με το οποίο ο εξυπηρετητής FTP δηλώνει ότι αποδέχεται τη σύνδεση;

Παρατηρώντας το τεμάχιο TCP με το οποίο ο ftp server δηλώνει πως αποδέχεται τη σύνδεση, βλέπουμε το relative ACK number ταυτίζεται Next Sequence Number και αφού πριν είχαμε ως Sequence Number το 0, επόμενη τιμή είναι το 0+1 = 1.

Σε απόλυτες τιμές, προκύπτει πως αν η τιμή του Sequence Number είναι x η τιμή του ACK number που λαμβάνει θα είναι x+1.

2.13 Πώς προκύπτουν ο αριθμός σειράς και ο αριθμός επιβεβαίωσης (Sequence Number και Acknowledgement Number) του τελευταίου τεμαχίου TCP της τριπλής χειραψίας με το οποίο ολοκληρώνεται η εγκατάσταση της σύνδεσης;

Αναφορικά με το 3ο τεμάχιο της τριπλής χειραψίας, το raw Sequence Number του είναι το raw Acknowledgment Number του προηγούμενου τεμαχίου, ενώ το raw Acknowledgment Number του είναι το raw Sequence Number του προηγούμενου τεμαχίου αυξημένο κατά 1.

2.14 Ποιο είναι το μήκος δεδομένων των τριών τεμαχίων της τριπλής χειραψίας;

Το μήκος δεδομένων των τριών τεμαχίων της τριπλής χειραψίας είναι μηδενικό, αφού όπως γνωρίζουμε δεν ανταλλάσσονται δεδομένα κατά τη διαδικασία αυτή.

2.15 Ποια είναι η μέγιστη τιμή που μπορεί να λάβουν οι αριθμοί σειράς και επιβεβαίωσης;

Το πεδίο τους καταλαμβανει 4 bytes. Επομένως, 4*8 = 32 bits μπορεί να είναι η μέγιστη τιμή τους σε δυαδικό άρα 2^{32} - 1 = 4.294.967.295 είναι η μέγιστη τιμή που μπορούν να πάρουν.

2.16 Εφαρμόστε φίλτρο απεικόνισης ώστε να παραμείνουν μόνο τα τεμάχια TCP για την τριπλή χειραψία. Ποια η σύνταξή του; [Υποδ. Εκτός από τη σημαία SYN θα χρειαστεί να φιλτράρετε για τεμάχια ACK με τους σωστούς σχετικούς αύξοντες αριθμούς σειράς, επιβεβαίωσης και μήκος τεμαχίου.]

Display filter: ***tcp.port == 21 and (tcp.flags.syn == 1 or (tcp.dstport == 21 and tcp.ack == 1 and tcp.seq == 1))***

Το TCP χρησιμοποιεί έλεγχο ροής με ολισθαίνον παράθυρο. Σε κάθε τεμάχιο TCP, η κάθε πλευρά ανακοινώνει στην άλλη το μέγεθος του παραθύρου (window), δηλαδή, το μέγιστο πλήθος byte που μπορεί να δεχθεί, ή ισοδύναμα, να στείλει η άλλη πλευρά. Ο έλεγχος ροής επιτυγχάνεται ως εξής: ο αποδέκτης διαφημίζει στην επικεφαλίδα της επαλήθευσης ένα παράθυρο λήψης ίσο με τον ελεύθερο χώρο προσωρινής μνήμης που διαθέτει και ο αποστολέας δεν στέλνει περισσότερα ανεπιβεβαίωτα byte από όσο ορίζει το παράθυρο αυτό.

2.17 Προσδιορίστε το μέγεθος του παραθύρου λήψης που ανακοινώνει ο υπολογιστής σας κατά τη διάρκεια της τριπλής χειραψίας στις συνδέσεις ελέγχου και δεδομένων FTP.

Window my PC announces: 8192 bytes

2.18 Ποιο είναι το αντίστοιχο μέγεθος παραθύρου λήψης που ανακοινώνει ο εξυπηρετητής;

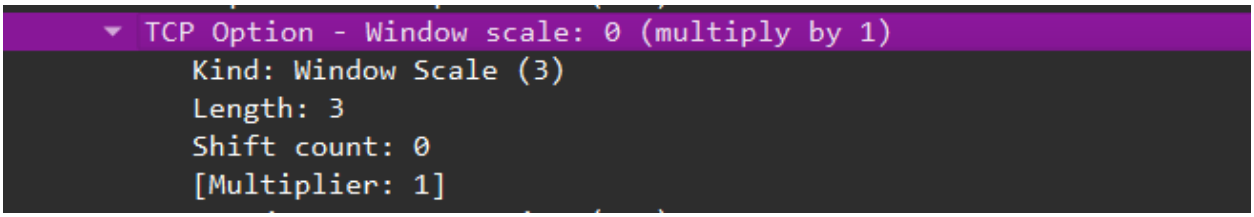
Window the server announces: 65535 bytes

2.19 Σε ποιο πεδίο της επικεφαλίδας μεταφέρεται η σχετική πληροφορία;

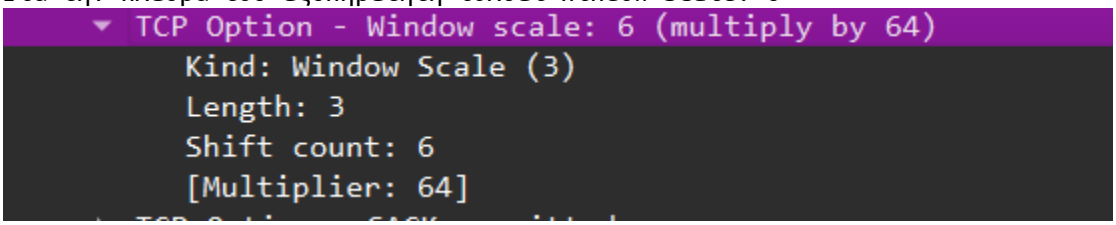
Στο πεδίο «Window».

2.20 Ποια τιμή κλίμακας παραθύρου (window scale) ανακοινώνουν οι δύο πλευρές σε κάθε σύνδεση;

- Για την πλευρά του υπολογιστή μας ισχύει Window scale : 0



- Για την πλευρά του εξυπηρετητή ισχύει Window scale: 6



2.21 Σε ποιο πεδίο της επικεφαλίδας μεταφέρεται η σχετική πληροφορία;
Στο TCP Option - Window scale όπως φαίνεται από τις παραπάνω εικόνες

Το TCP προσπαθεί να αποφύγει τον θρυμματισμό (fragmentation) των πακέτων IPv4. Όταν εγκαθίσταται η σύνδεση TCP, γίνεται ανακοίνωση του μέγιστου μεγέθους τεμαχίου (MSS – Maximum Segment Size). Το MSS είναι το μέγιστο μέγεθος δεδομένων (σε byte) του στρώματος εφαρμογής που μπορεί να δεχθεί ο προορισμός από την πηγή σε ένα πακέτο IP, δηλαδή, η τιμή της MTU μείον το ελάχιστο μήκος επικεφαλίδας των πρωτοκόλλων IP και TCP. Επομένως, MSS = MTU – 40 στην περίπτωση IPv4 και MSS = MTU – 60 στην περίπτωση IPv6. Τόσο ο πελάτης όσο και ο εξυπηρετητής TCP μπορούν να ανακοινώσουν το MSS σε μία επιλογή (option) της επικεφαλίδας TCP του πρώτου τεμαχίου TCP που μεταδίδεται. Έτσι, ο αποστολέας της πληροφορίας στέλνει τεμάχια μεγέθους τέτοιου ώστε να γίνονται αποδεκτά από τον παραλήπτη χωρίς να απαιτηθεί θρυμματισμός στη διεπαφή αυτού. Η ανταλλαγή των MSS αφορά μόνο στα δύο άκρα, και όχι στους ενδιάμεσους δρομολογητές από όπου θα διέλθει το πακέτο IPv4. Για να διαπιστωθεί η μικρότερη MTU (Maximum Transmission Unit) της διαδρομής από τον αποστολέα ως τον παραλήπτη, το TCP χρησιμοποιεί τη διαδικασία Path MTU Discovery που είδατε στην Εργαστηριακή Άσκηση 6. Με βάση την προηγούμενη καταγραφή της κίνησης FTP, απαντήστε στα παρακάτω ερωτήματα.

2.22 Ποια τιμή του MSS ανακοινώνει ο υπολογιστής σας κατά την εγκατάσταση της σύνδεσης ελέγχου FTP. [Υπόδειξη: Αναζητήστε μεταξύ των παραμέτρων που εμφανίζονται στο παράθυρο με τη λίστα καταγεγραμμένων πακέτων].
MSS που ανακοινώνει ο υπολογιστής μου : 1460 bytes

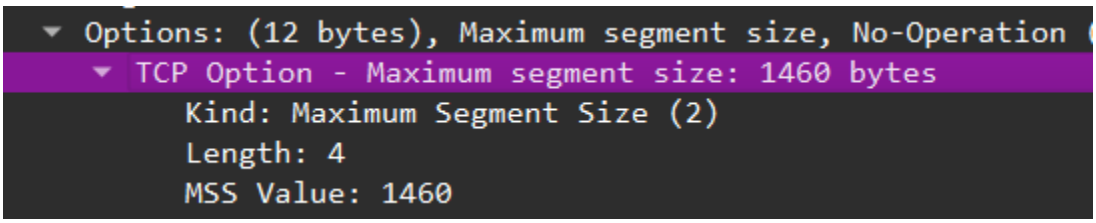
2.23 Πώς προκύπτει η παραπάνω τιμή από την MTU της διεπαφής του υπολογιστή σας;
1) Αναζητούμε την MTU του υπολογιστή μας.

```
C:\Windows\System32>netsh interface ipv4 show subinterfaces

      MTU  MediaSenseState      Bytes In     Bytes Out  Interface
----------  ---------------  ------------  ------------  -------------
4294967295                1             0        212010  Loopback Pseudo-Interface 1
      1500                5             0             0  Local Area Connection
      1500                1     269221456       7345821  Wi-Fi
      1500                5             0             0  Ethernet
      1500                5             0             0  Local Area Connection* 1
      1500                5             0             0  Local Area Connection* 2
```

2) Αφαιρούμε από την MTU τις IPv4 και TCP επικεφαλίδες που έχουν μήκος 40 bytes
3) Προκύπτει, MSS = 1500-40 = 1460 bytes.

2.24 Σε ποιο πεδίο της επικεφαλίδας TCP μεταφέρεται η τιμή του MSS;
Πεδιο Option --> TCP Option – Maximum segment size



2.25 Ποια τιμή του MSS ανακοινώνει ο edu-dy.cn.ntua.gr.
MSS που ανακοινώνει ο edu-dy.cn.ntua.gr : 536 bytes

2.26 Πώς προκύπτει αυτή από την MTU (576 byte) της διεπαφής του edu-dy.cn.ntua.gr;
MSS = MTU – IPv4 Header Length – TCP Header Length = 576 – 40 = 536 bytes.

2.27 Ποιο είναι το μέγεθος του μεγαλύτερου τεμαχίου TCP που μπορεί να στείλει ο υπολογιστής σας προς τον εξυπηρετητή στη σύνδεση που εγκαταστάθηκε κατά την τριπλή χειραψία TCP στη θύρα ελέγχου του FTP;
Είναι ίδιο με την MSS του εξυπηρετητή, δηλαδή 536 bytes.

Απόλυση σύνδεσης

Στη συνέχεια, αφού ακυρώσετε το τρέχον φίλτρο απεικόνισης, εφαρμόστε αυτό της ερώτησης 2.4 ώστε να εντοπίσετε τα τεμάχια TCP που σχετίζονται με την απόλυση της σύνδεσης ελέγχου FTP.

168	0.002130	147.102.40.15	192.168.1.14	TCP	21 → 60953 [FIN, ACK] Seq=376 Ack=116 Win=65920 Len=0
169	0.000038	192.168.1.14	147.102.40.15	TCP	60953 → 21 [ACK] Seq=116 Ack=377 Win=7817 Len=0
170	0.000686	192.168.1.14	147.102.40.15	TCP	60953 → 21 [FIN, ACK] Seq=116 Ack=377 Win=7817 Len=0
171	0.015708	147.102.40.15	192.168.1.14	TCP	21 → 60953 [ACK] Seq=377 Ack=117 Win=65920 Len=0

2.28 Ποια σημαία μήκους 1 bit ενεργοποιείται για την εκκίνηση της απόλυσης της σύνδεσης TCP;
Η σημαία FIN.

2.29 Ποια πλευρά εκκινεί τη διαδικασία απόλυσης;
Η πλευρά του εξυπηρετητή.

2.30 Πόσα τεμάχια TCP ανταλλάσσονται συνολικά;

4 τεμάχια σχετίζονται με την απόλυση της σύνδεσης.

2.31 Ποιο είναι το μέγεθος των επικεφαλίδων TCP των τεμαχίων αυτών;

TCP header length : 20 bytes.

2.32 Ποιο είναι το μέγεθος δεδομένων των τεμαχίων αυτών;

Μηδενικό.

2.33 Δικαιολογήστε το μήκος του πακέτου IPv4 και του πλαισίου Ethernet που μεταφέρει το τεμάχιο TCP με το οποίο απολύει τη σύνδεση ο υπολογιστής σας.

Το πακέτο αποτελείται μόνο από επικεφαλίδες / headers, άρα το συνολικό μήκος είναι Ethernet Header + IPv4 Header + TCP Header = 14 + 20 + 20 = 54 bytes.

2.34 Δικαιολογήστε το μήκος του πακέτου IPv4 και του πλαισίου Ethernet που μεταφέρει το αντίστοιχο τεμάχιο TCP από τον edu-dy.cn.ntua.gr.

Το πακέτο αποτελείται μόνο από επικεφαλίδες / headers, άρα το συνολικό μήκος είναι Ethernet Header + IPv4 Header + TCP Header = 20 + 20 + 20 = 60 bytes.

Σημειώνεται ότι το Ethernet header είναι 6 bytes μεγαλύτερο στην καταγραφή του Wireshark για τα εισερχόμενα πακέτα γιατί περιλαμβάνεται και το Padding.

Ο λόγος για τον οποίο τα εξερχόμενα πακέτα είναι μικρότερα οφείλεται στο ότι το padding προστίθεται από το NIC. Καθώς τα εξερχόμενα πακέτα περνούν libpcap/winpcap/npcap στο δρόμο προς το NIC, η αναπλήρωση δεν έχει πραγματοποιηθεί ακόμα, με αποτέλεσμα μικρότερα frames.

2.35 Πόσα byte μεταδόθηκαν συνολικά στη σύνδεση ελέγχου FTP από κάθε πλευρά;

Η πλευρά του εξυπηρετητή μετέδωσε συνολικά 1047 bytes, ενώ ο υπολογιστής μας μετέδωσε 1045 bytes.

Αν υπολογίσοουμε μόνο τα μήκη των TCP τεμαχίων τότε προκύπτει:

Από τον υπολογιστή μου στον εξυπηρετητή: 116 bytes

Από τον εξυπηρετητή στον υπολογιστή μου: 377 bytes

2.36 Με ποιο τρόπο προσδιορίσατε το πλήθος τους;

Για να βρούμε όσα μας έστειλε ο εξυπηρετητής εφαρμόσαμε το φίλτρο ***tcp.port==21 and ip.dst==192.168.1.14*** και αθροίσαμε τα bytes της στήλης Length. Για να βρούμε το αντίστροφο, αλλάξαμε το φίλτρο σε ***tcp.port==21 and ip.dst==147.102.40.15*** και ακολουθήσαμε την ίδια διαδικασία.

Για τη δεύτερη προσέγγιση του 2.35 χρησιμοποιήσαμε του relative sequence number των αντίστοιχων τελευταίων frames από κάθε πλευρά.

Μεταφορά δεδομένων

Εφαρμόστε νέο φίλτρο απεικόνισης της μορφής tcp.port ώστε να παραμείνουν μόνο τα τεμάχια TCPπου σχετίζονται με τη θύρα δεδομένων FTP.

2.37 Ποια είναι η σύνταξή του φίλτρου αυτού;
Display filter : ***tcp.port == 20***

2.38 Ποια τιμή του MSS ανακοινώνει η κάθε πλευρά κατά την τριπλή χειραψία TCP στη θύρα δεδομένων FTP;
MSS που ανακοινώνει ο υπολογιστή μας : 1460 bytes

MSS που ανακοινώνει ο εξυπηρετητής : 536 bytes

2.39 Ποιο είναι το μέγεθος του μεγαλύτερου τεμαχίου TCP που μπορεί να στείλει ο εξυπηρετητής προς τον υπολογιστή σας στη σύνδεση που εγκαταστάθηκε κατά την τριπλή χειραψία TCP στη θύρα δεδομένων του FTP;

Το μέγεθος του μεγαλύτερου τεμαχίου TCP που μπορεί να στείλει ο σέρβερ στον υπολογιστή μας ανέρχεται σε 1480 bytes, 1460 του payload (MSU) και επιπλέον 20 της επικεφαλίδας TCP.

2.40 Ποια είναι η τιμή του RTT (Round Trip Time) όπως αυτή προκύπτει από την ανταλλαγή των δύο πρώτων τεμαχίων της τριπλής χειραψίας;
0.000108 sec

2.41 Ο υπολογιστής σας στέλνει επιβεβαιώσεις για κάθε τεμάχιο TCP που λαμβάνει;
Όχι, δεν στέλνει για κάθε τεμάχιο. Στέλνει ανά τυχαίο αριθμό τεμαχίων(π.χ. 3, 5, 20, 11)

2.42 Εφαρμόστε φίλτρο ώστε να παραμείνουν μόνο τεμάχια της σύνδεσης δεδομένων TCP με πηγή τον εξυπηρετητή. Πόσα τεμάχια με δεδομένα έστειλε ο εξυπηρετητής;
Display filter : ***tcp.port == 20 and ip.src == 147.102.40.15 and tcp.payload***

118 τεμάχια έστειλε ο εξυπηρετητής.

2.43 Εφαρμόστε φίλτρο ώστε να παραμείνουν μόνο τεμάχια της σύνδεσης δεδομένων TCP με προορισμό τον εξυπηρετητή. Πόσα τεμάχια ACK έστειλε ο υπολογιστής σας για τα δεδομένα που έλαβε;
Display filter : ***tcp.port == 20 and ip.src == 192.168.1.14 and tcp.ack***

21 τεμάχια ACK έστειλε ο υπολογιστής μας.

2.44 Ποια τιμή παραθύρου (window) ανακοινώνει ο υπολογιστής σας στο πρώτο μετά την τριπλή χειραψία τεμάχιο ACK;
Window: 4097

[Calculated window size: 1048832]

2.45 Είναι ίδια με αυτή που προσδιορίσατε προηγουμένως στην ερώτηση 2.17 για τη σύνδεση δεδομένων; Εάν όχι, πώς προκύπτει;

Όχι, δεν είναι ίδια. Το ότι δεν αλλάζουν οι τιμές, σημαίνει πως δεν υπάρχει υπερφόρτωση του buffer του υπολογιστή μας.

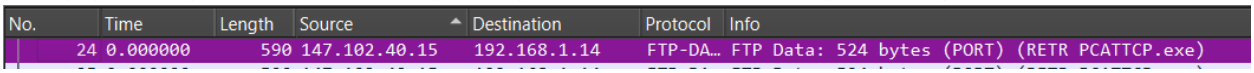
2.46 Αλλάζει η τιμή του παραθύρου καθώς προχωρά η μεταφορά του αρχείου; Ποια είναι η μικρότερη τιμή που παρατηρήσατε;

Όχι, δεν αλλάζει και παραμένει ίδια σε όλα τα πακέτα μετά από αυτό που περιγράφεται στο 2.44 ερώτημα.

2.47 Εάν ο υπολογιστής σας ανακοίνωνε μηδενική τιμή για το παράθυρο, τι θα έκανε ο εξυπηρετητής;

Αν συμβεί αυτό σημαίνει ότι ο υπολογιστής μας δεν μπορεί να διαχειριστεί αυτή τη στιγμή επιπλέον δεδομένα γιατί έχει γεμίσει ο buffer του από την κίνηση στο δίκτυο. Στην περίπτωση αυτή ο εξυπηρετητής θα κάνει retransmission των πιο πρόσφατων δεδομένων που έστειλε ώσπου να λάβει 0> window size σε Ack μήνυμα του υπολογιστή μας και να συνεχίσει σε επόμενα δεδομένα.

Εφαρμόστε φίλτρο απεικόνισης ftp-data ώστε να εμφανίζονται μόνο τα τεμάχια TCP που αφορούν μεταφορά δεδομένων FTP και επιλέξτε το πρώτο που στέλνει ο edu-dy.cn.ntua.gr.



No.	Time	Length	Source	Destination	Protocol	Info
24	0.000000	590	147.102.40.15	192.168.1.14	FTP-DA…	FTP Data: 524 bytes (PORT) (RETR PCATTCP.exe)

2.48 Να καταγραφεί το μέγεθος πλαισίου (frame) σε byte και το μήκος των επικεφαλίδων Ethernet,IP και TCP.

Frame size : 590 bytes

Ethernet header : 14 bytes

Ipv4 header : 20 bytes

TCP header : 32 bytes

FTP Data : 524 bytes

2.49 Είναι το μέγεθος των δεδομένων του τεμαχίου TCP το αναμενόμενο βάσει της τιμής του ερωτήματος 2.39;

Όχι, δεν είναι. Είναι μικρότερο πιθανόν λόγω της μικρότερης MTU ενδιάμεσων κόμβων.

2.50 Εάν για κάποιο λόγο έπρεπε ο εξυπηρετητής να αποστείλει δεδομένα μεγαλύτερα από την τιμή που βρήκατε πριν, τι θα συνέβαινε; [Υπόδειξη: Αναζητήστε Source Fragmentation στο RFC 879.]

Σε αυτήν την περίπτωση, η πύλη πρέπει να κατακερματίσει το datagram, εκτός εάν φέρει την ένδειξη "don't fragment", οπότε απορρίπτεται, με την επιλογή αποστολής ενός μηνύματος ICMP στην πηγή που αναφέρει το πρόβλημα).

2.51 Πόσα byte δεδομένων μεταδόθηκαν συνολικά στη σύνδεση δεδομένων από κάθε πλευρά;

[Υπόδειξη: Χρησιμοποιήστε τους αριθμούς επιβεβαίωσης των τεμαχίων για τον υπολογισμό.]

Το πρωτο τεμάχιο που έστειλε ο υπολογιστής μας μετά το τελευταίο FTP-data τεμάχιο έχει:

Acknowledgment Number: 61442   (relative ack number)

Άρα προς τον υπολογιστή μας στάλθηκαν 61442 bytes.

Αντίστοιχα, βρίσκουμε ότι στάλθηκαν προς τον εξυπηρετητή μόνο 2 bytes δεδομένων αφού

Acknowledgment Number: 2   (relative ack number)

Στο τελευταίο ACK πακέτο που στέλνει ο server.

2.52 Ποιος ήταν ο ρυθμός μεταφοράς δεδομένων σε kbyte/sec από τον εξυπηρετητή στο PC σας;
Ρυθμός μεταφοράς = (61442 bytes/ 0.079228 sec)=775508.7 bytes/sec = 775.51 kbytes/sec

Για τον υπολογισμό του χρόνου αφαιρέσαμε την χρονική στιγμή άφιξης του πρώτου από το τελεταίο πακέτο FTP-DATA.

2.53 Υπήρξαν αναμεταδόσεις τεμαχίων κατά τη μεταφορά δεδομένων; Εάν ναι, πώς το αντιληφτήκατε;
Δεν εντοπίστηκε κάποιο Retransmission TCP πακέτο, επομένως δεν υπήρξαν αναμεταδόσεις τεμαχίων.

3 Αποφυγή συμφόρησης στο TCP

Καθώς το διαδίκτυο άρχισε να διαδίδεται παρατηρήθηκαν φαινόμενα συμφόρησης που οφείλονταν στην απώλεια πακέτων λόγω υπερχείλισης των χώρων αποθήκευσης στους δρομολογητές. Για να προληφθούν τέτοιες καταστάσεις τροποποιήθηκε εκ των υστέρων η λειτουργία του TCP με την εισαγωγή ενός αλγόριθμου ελέγχου και αποφυγής συμφόρησης. Προς τούτο χρησιμοποιήθηκε ο μηχανισμός ολισθαίνοντος παραθύρου του TCP που αρχικά είχε σκοπό τον έλεγχο ροής, δηλαδή, να μην στέλνει η πηγή πιο γρήγορα από ότι μπορεί να δεχθεί ο προορισμός. Η κεντρική ιδέα είναι να εκτιμά η πηγή τη φόρτιση του δικτύου μέσω ενός παραθύρου συμφόρησης και να μην στέλνει περισσότερα byte από όσα μπορεί να προωθήσει το δίκτυο και δεχθεί ο προορισμός. Η εκτίμηση του παραθύρου συμφόρησης βασίζεται στην ανάδραση που λαμβάνει η πηγή από το δίκτυο. Μια μέθοδος είναι ο μηχανισμός εκκίνησης που αποκαλείται αργή αρχή (slow start) και περιγράφεται στο RFC 5681. Συγκεκριμένα, η πηγή ξεκινά με αρχικό παράθυρο μερικών MSS, δύο έως τέσσερα, ανάλογα με το μέγεθος του MSS, και στη συνέχεια για κάθε ACK που λαμβάνει, αυξάνει το παράθυρο συμφόρησης κατά ένα MSS.

Για το μέρος αυτό της άσκησης θα χρησιμοποιήσετε μια έτοιμη καταγραφή και αυτήν που αποθηκεύσατε προηγουμένως. Συνδεθείτε όπως πριν με ftp στον edu-dy.cn.ntua.gr και, αφού αλλάξετε σε δυαδικό τρόπο μεταφοράς, κατεβάστε το αρχείο pcattcp.pcap. Το αρχείο περιέχει μια καταγραφή, από την πλευρά του edu-dy.cn.ntua.gr, του κατεβάσματος με ftp του αρχείου PCATTCP.exe.

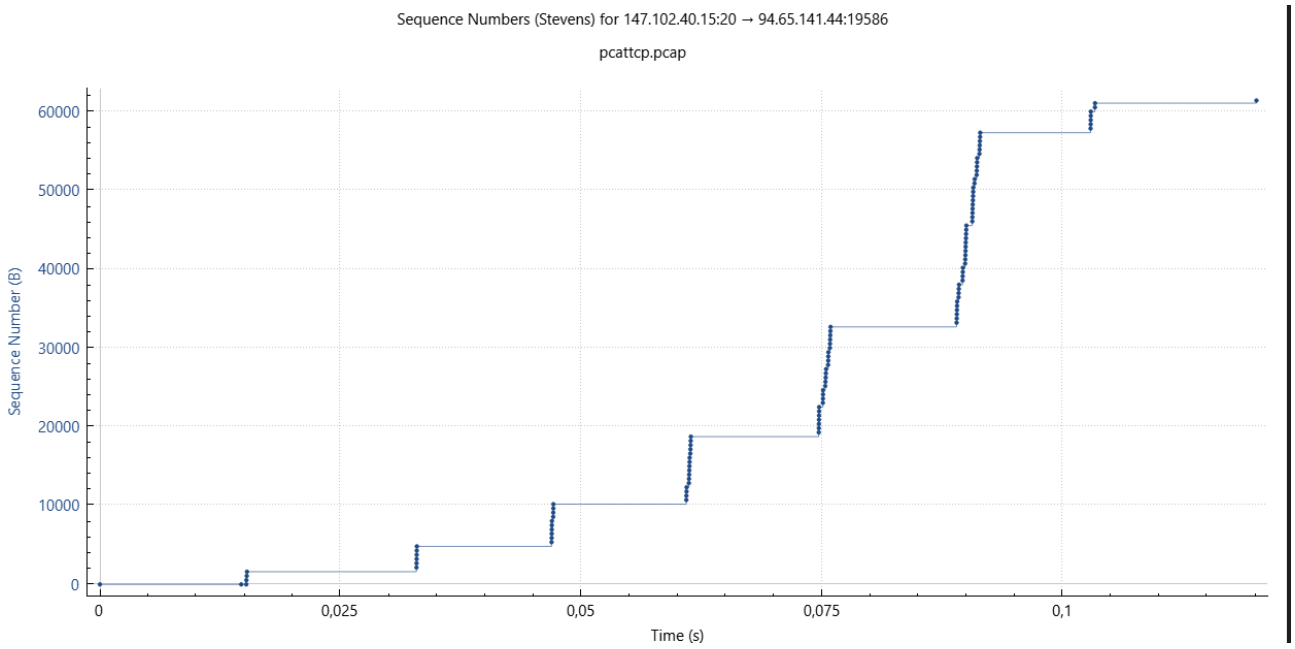
3.1 Ανοίξτε το αρχείο pcattcp.pcap στο Wireshark και εφαρμόστε φίλτρο απεικόνισης ώστε να παραμείνουν μόνο τα τεμάχια TCP που σχετίζονται με τη θύρα δεδομένων FTP. Ποια η σύνταξή του;
Φίλτρο απεικόνισης: ***tcp.port == 20***

3.2 Εντοπίστε τα τεμάχια της τριπλής χειραψίας. Ποια η διεύθυνση IPv4 του υπολογιστή που κατέβασε το αρχείο PCATTCP.exe;
Παρακάτω βλέπουμε τα πακέτα της τριμερούς χειραψίας, επομένως η IP του υπολογιστή που κατέβασε το PCATTCP.exe είναι 94.65.141.44.

3.3 Ποιο είναι το RTT της σύνδεσης όπως προκύπτει από την ανταλλαγή των δύο πρώτων τεμαχίων της τριπλής χειραψίας; Συγκρίνετε με αυτήν που βρήκατε προηγουμένως στο ερώτημα 2.40.

Είναι RTT: 0.014674000 seconds και είναι μεγαλύτερο από εκείνο του 2.40 ερωτήματος.

Εμφανίστε το διάγραμμα αριθμών σειράς συναρτήσει του χρόνου από το μενού Statistics--> TCP Stream Graphs--> Time Sequence (Stevens). Κάθε τελεία στο διάγραμμα αντιστοιχεί ένα τεμάχιο TCP. Λόγω του μηχανισμού ολισθαίνοντος παραθύρου τα τεμάχια τείνουν να συσσωρεύονται ανά RTT. Όπου οι τελείες εμφανίζονται η μία πάνω από την άλλη, τα αντίστοιχα τεμάχια στάλθηκαν το ένα πίσω από το άλλο περίπου την ίδια χρονική στιγμή. Όταν ληφθούν ACK για αυτά θα αρχίσει μια νέα αντίστοιχη αποστολή, κοκ.



3.4 Από το κουμπί Switch Direction επιλέξτε ως πηγή των τεμαχίων τον edu-dy.cn.ntua.gr.Παρατηρώντας προσεκτικά το διάγραμμα, τι συμπεραίνετε σχετικά με τον τρόπο που στέλνονται τα τεμάχια TCP από τον edu-dy.cn.ntua.gr;

Από το παραπάνω διάγραμμα παρατηρούμε ότι κάθε φορά που γίνεται ACK από τον παραλήπτη το μέγεθος του παραθύρου μεγαλώνει εκθετικά διπλασιάζεται κάθε φορά.

3.5 Πόσα τεμάχια έστειλε ο edu-dy.cn.ntua.gr στο πρώτο RTT; Είναι το πλήθος τους σύμφωνο με ότι προβλέπει το RFC 5681 στην παρ. 3.1;

```
If SMSS > 2190 bytes:
       IW = 2 * SMSS bytes and MUST NOT
be more than 2 segments
   If (SMSS > 1095 bytes) and (SMSS <=
2190 bytes):
       IW = 3 * SMSS bytes and MUST NOT
be more than 3 segments
   if SMSS <= 1095 bytes:
       IW = 4 * SMSS bytes and MUST NOT
be more than 4 segments
```

https://www.rfc-editor.org/rfc/rfc5681

Έστειλε αρχικά 4 τεμάχια. Άρα το πλήθος αυτό είναι σύμφωνο με τον RFC 5681 αφού

SMSS <= 1095 bytes.

3.6 Πόσα τεμάχια έστειλε κατά το δεύτερο, τρίτο και τέταρτο RTT;

Στο 2ο έστειλε 6 τεμάχια, στο 3ο έστειλε 10 τεμάχια και στο 4ο 16 τεμάχια.

Αυτό που στην πραγματικότητα συμβαίνει είναι πως σε κάθε RTT αποστολής δεδομένων στέλνονται ολοένα και περισσότερα ACK σχεδόν ταυτόχρονα δίνοντάς μας την ψευδαίσθηση ότι στέλνονται σε μία ριπή περισσότερα πακέτα, ενώ στην πραγματικότητα η ριπή αυτή αποτελείται από αυξανόμενο αριθμό μικρότερων. Αυτό επιβεβαιώνει και τον Slow Start αλγόριθμο, καθώς ξεκινάει με μικρό ρυθμό μετάδοσης για να δοκιμάσει το δίκτυο και αυξάνει σταδιακά.

3.7 Δείτε το αντίστοιχο διάγραμμα για την κίνηση από την άλλη πλευρά κάνοντας κλικ στο Switch Direction. Πόσα ACK στάλθηκαν στο πρώτο, δεύτερο και τρίτο RTT; Τι παρατηρείτε σε σχέση με την απάντησή σας στο προηγούμενο ερώτημα; [Σημ. Δείτε παρ. 2.3 στο RFC 3465.]

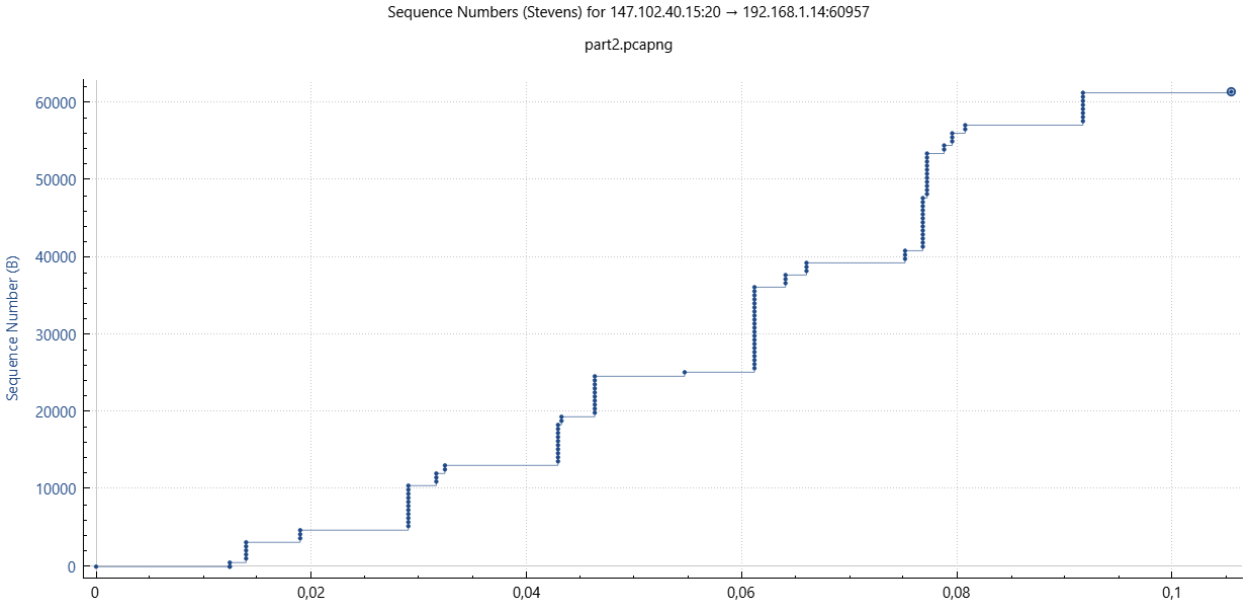
Στο πρώτο 1, στο δεύτερο 2, στο τρίτο 3(4ο 5, 5ο 11).

> This document specifies that TCP implementations MAY use L=2*SMSS bytes and MUST NOT use L > 2*SMSS bytes. This choice balances between being conservative (L=1*SMSS bytes) and being potentially very aggressive. In addition, L=2*SMSS bytes exactly balances the negative impact of the delayed ACK algorithm (as discussed in more detail in section 3.2). Note that when L=2*SMSS bytes cwnd growth is roughly the same as the case when the standard algorithms are used in conjunction with a receiver that transmits an ACK for each incoming segment [All98] (assuming no or small amounts of ACK loss in both cases).
>
> The exception to the above suggestion is during a slow start phase that follows a retransmission timeout (RTO). In this situation, a TCP MUST use L=1*SMSS as specified in RFC 2581 since ACKs for large amounts of previously unacknowledged data are common during this phase of a transfer. These ACKs do not necessarily indicate how much data has left the network in the last RTT, and therefore ABC cannot accurately determine how much to increase cwnd. As an example, say segment N is dropped by the network, and segments N+1 and N+2 arrive successfully at the receiver. The sender will receive only two duplicate ACKs and therefore must rely on the retransmission timer (RTO) to detect the loss. When the RTO expires, segment N is retransmitted. The ACK sent in response to the retransmission will be for segment N+2. However, this ACK does not indicate that three segments have left the network in the last RTT, but rather only a single segment left the network. Therefore, the appropriate cwnd increment is at most 1*SMSS bytes.
>
> https://www.rfc-editor.org/rfc/rfc3465#section-3.2

Με βάση την παραπάνω πηγή διαπιστώνουμε ότι για τα πρώτα RRT αυξάνεται μόλις κατά ένα το πλήθος των απεσταλμένων ACKs ενώ στη συνέχει ακολουθεί πολιτική διπλασιασμού.

3.8 Στη δική σας καταγραφή επιλέξτε το πρώτο τεμάχιο δεδομένων FTP από τον edu-dy.cn.ntua.gr και εμφανίστε το αντίστοιχο διάγραμμα αριθμών σειράς συναρτήσει του χρόνου από το edu-dy.cn.ntua.gr προς τον υπολογιστή σας. Είναι παρόμοιο με αυτό του αρχείου που κατεβάσατε;

Συγκρίνετε με τις απαντήσεις στα προηγούμενα ερωτήματα; [Σημ. Δείτε παρ. 2 στο RFC 6928.]



Είναι παρόμοιο παρουσιάζοντας ωστόσο κοντά στις μεγάλες ριπές ανα RRT και κάποιες γετονικές χρονικά πιο μικρές. Αυτό ενδεχομένως να οφείλεται στο ότι κάνουμε την καταγραφή από το οικιακό μας δίκτυο και μπορεί να μεσολαβήσει κάποιο traffic μεταξύ των κόμβων που διακόπτει στιγμιαία τη συνεχή αποστολή πακέτων ανά RRT.

4 Μετάδοση δεδομένων με UDP

Το πρωτόκολλο μεταφοράς UDP παρέχει μια υπηρεσία “καλύτερης προσπάθειας” χωρίς σύνδεση (connectionless) που δεν εγγυάται την παράδοση των δεδομένων. Είναι μια μινιμαλιστική επέκταση της υπηρεσίας “best-effort” του IP που δίνει στις εφαρμογές άμεση πρόσβαση σε μια υπηρεσία δεδομενογραμμάτων. Τα δεδομενογράμματα UDP μπορεί να χαθούν (μη αξιόπιστη μετάδοση) ή να παραδοθούν εκτός σειράς στο ανώτερο στρώμα. Κάθε δεδομενόγραμμα UDP αντιμετωπίζεται ανεξάρτητα από τα άλλα. Χρησιμοποιείται από εφαρμογές που δεν απαιτούν το επίπεδο υπηρεσίας που προσφέρει το TCP ή θέλουν να χρησιμοποιήσουν υπηρεσίες χωρίς σύνδεση (π.χ. εκπομπή ή πολλαπλή διανομή). Το UDP είναι ένα λιτό πρωτόκολλο μεταφοράς για να στέλνει κανείς όσο γρήγορα μπορεί. Οι μόνες επιπλέον υπηρεσίες που παρέχει σε σχέση με το IP είναι το πεδίο ελέγχου για τα δεδομένα και η πολυπλεξία μέσω της θύρας UDP. Έτσι οποιαδήποτε εφαρμογή το χρησιμοποιεί πρέπει να χειρισθεί απευθείας τα από άκρο σε άκρο προβλήματα της επικοινωνίας εάν αυτό είναι απαραίτητο. Για περισσότερες λεπτομέρειες σχετικά με τα πεδία της επικεφαλίδας του UDP ανατρέξτε στην ιστοσελίδα http://www.networksorcery.com/enp/protocol/udp.htm που θα αναζητήσετε στο Internet Archive.

Με τη βοήθεια του Wireshark να καταγράψετε την κίνηση ενώ κάνετε χρήση της υπηρεσίας DNS.

Εφαρμόστε φίλτρο σύλληψης για να παρατηρείτε μόνο κίνηση του πρωτοκόλλου UDP και ξεκινήστε την καταγραφή. Ανοίξτε ένα παράθυρο εντολών και καθαρίστε την προσωρινή μνήμη DNS (DNS cache) που διατηρεί ο υπολογιστής. Εάν χρησιμοποιείτε Windows, σε ένα παράθυρο εντολών εκτελέστε την εντολή ipconfig /flushdns. Σε Ubuntu εκτελέστε την εντολή sudo systemd-resolve--flush-caches. Σε συστήματα Unix/Linux, εν γένει δεν χρησιμοποιείται προσωρινή αποθήκευση για την επίλυση ονομάτων. Εάν όμως την έχετε ενεργοποιήσει, διαγράψτε τα περιεχόμενά της επανεκκινώντας την αντίστοιχη υπηρεσία, π.χ. nscd, unbound, κλπ.

Στη συνέχεια τρέξτε το πρόγραμμα nslookup σε περιβάλλον Windows, dig σε περιβάλλον Linux ή vhost σε περιβάλλον Unix, για να ζητήσετε τη διεύθυνση IP του edu-dy.cn.ntua.gr.

4.1 Ποια είναι η σύνταξη του φίλτρου σύλληψης που χρησιμοποιήσατε;
Capture filter : ***udp***

Παρατηρήστε το πρώτο δεδομενόγραμμα UDP που αποστάλθηκε από τον υπολογιστή σας.

4.2 Καταγράψτε τα ονόματα και το μήκος των πεδίων της επικεφαλίδας δεδομενογράμματος UDP.

- ✓ Source Port (2 bytes)
- ✓ Destination Port (2 bytes)
- ✓ Length (2 bytes)
- ✓ Checksum (2 bytes)

4.3 Ποιο είναι το συνολικό μέγεθος της επικεφαλίδας UDP;
UDP Header Length : 8 bytes

4.4 Ποιο είναι το μήκος του συγκεκριμένου δεδομενογράμματος βάσει του μεγέθους του πακέτου IPv4 ή IPv6 εντός του οποίου ενθυλακώνεται;
Το IPv4 πακέτο έχει συνολικό μέγεθος 198 bytes όπως φαίνεται στο πεδίο Total length της IPv4 επικεφαλίδας. Άρα το UDP datagram θα έχει μέγεθος

198 bytes- IPv4 Header(=20 bytes) = 178 bytes

4.5 Τι εκφράζει το πεδίο μήκος (Length) της επικεφαλίδας UDP;
Εκφράζει το μήκος των δεδομένων + το μήκος της επικεφαλίδας UDP

4.6 Ποια είναι η ελάχιστη τιμή του πεδίου μήκους της επικεφαλίδας UDP;
0.

4.7 Ποιο είναι το ελάχιστο και ποιο το μέγιστο μέγεθος μηνύματος που μπορεί να μεταφερθεί από ένα πακέτο IPv4 χρησιμοποιώντας το πρωτόκολλο UDP; Αιτιολογήστε την απάντησή σας.
Το ελάχιστο μέγεθος μηνύματος είναι προφανώς 0 bytes δεδομένων(payload) και περιλαμβάνοντας στο IPv4 πακέτο τα 8 bytes του UDP Header με την προσθήκη των άλλων 20 bytes της επικεφαλίδας IPv4 έχουμε σύνολο 28 bytes για minimum packet size.

Σχετικά με τη μέγιστη μέγεθος είδαμε πως το πεδίο Length παραπάνω αφορά το συνολικό μήκος και ότι είναι 2 bytes = 16bits, επομένως, θα περιμέναμε η μέγιστη τιμή που μπορεί να λάβει είναι 2^{16} – 1 = 65535 bytes. Ωστόσο, το πεδίο Total Length της IPv4 επικεφαλίδας είναι επίσης 2 bytes, με μέγιστη τιμή 65.535 bytes, επομένως η μέγιστη τιμή που μπορεί να λάβει ένα UDP Datagram είναι 65535 – 20 = 65515 bytes, όπου 20 bytes το ελάχιστο μέγεθος μιας IP επικεφαλίδας. Άρα αν αφαιρέσουμε και 8 bytes της UDP επικεφαλίδας το payload θα είναι 65507 bytes.

Το όριο των 65535 bytes είναι θεωρητικό και ενδέχεται να μην είναι πρακτικό σε όλα τα δίκτυα λόγω του MTU.

4.8 Δοθέντος ότι όλοι οι κόμβοι στο διαδίκτυο οφείλουν να δέχονται πακέτα IPv4 μεγέθους μέχρι 576 byte (θρυμματισμένα ή μη), ποιο είναι το μέγιστο μέγεθος μηνύματος που μπορεί να σταλεί και παραληφθεί με βεβαιότητα χρησιμοποιώντας το πρωτόκολλο UDP.
Το πεδίο Header Length ενός IPv4 πακέτου αποτελείται από 4 bits, επομένως παίρνει μέγιστη τιμή 2^4 -1 = 15_{10} και δεδομένου ότι μετράει το μέγεθος σε λέξεις των 4 bytes, το μέγιστο IPv4 header είναι 60 bytes. Επομένως, προκειμένου ένα πακέτο UDP να σταλεί/παραληφθεί με βεβαιότητα πρέπει να έχει συνολικό μήκος μέχρι και (576-60) = 516 bytes μαζί με την επικεφαλίδα του(8 bytes).

4.9 Παρατηρήσατε στην καταγραφή σας να μεταφέρονται με δεδομενογράμματα UDP μηνύματα άλλων πλην του DNS πρωτοκόλλων; Εάν ναι, για ποια πρωτόκολλα πρόκειται;
Παρατήρησα το SSDP πρωτόκολλο.

Εφαρμόστε φίλτρο απεικόνισης ώστε να παραμείνουν μόνο μηνύματα DNS.

4.10 Ποια είναι η σύνταξη του φίλτρου απεικόνισης που χρησιμοποιήσατε;
Display filter : ***dns***

4.11 Ποια είναι η διεύθυνση IPv4 ή IPv6 του εξυπηρετητή DNS που απάντησε στην ερώτηση για τη διεύθυνση του edu-dy.cn.ntua.gr;
Source Address of query response: 2001:4860:4860::8888(IPv6)

4.12 Καταγράψτε τις θύρες (προέλευσης και προορισμού) του πρωτοκόλλου μεταφοράς που χρησιμοποιήθηκαν για ερώτηση (query) στον εξυπηρετητή DNS.
Query

Source Port: 53691

Destination Port: 53

4.13 Καταγράψτε τις θύρες (προέλευσης και προορισμού) του πρωτοκόλλου μεταφοράς που χρησιμοποιήθηκαν στην απόκριση (response) του εξυπηρετητή DNS.

Response

Source Port: 53

Destination Port: 53691

4.14 Ποια από τις παραπάνω θύρες αντιστοιχεί στο πρωτόκολλο εφαρμογής DNS;

Η θύρα 53 (default θύρα DNS)