

Παντελεήμων Πρώιος

ice18390023

ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ II

TCP

Email: ice18390023@uniwa.gr

Εξάμηνο: 4^ο

Ερώτηση 1:

Το πραγματικό sequence number είναι το 232,129,012 αλλά το wireshark μας το δείχνει και ως 0 για ευκολία. Το flag που έχει ενεργοποιηθεί είναι το SYN bit απο τα control bits που υποδικνίει την ανάγκη για εγκαθύδριση επικοινωνίας.

Ερώτηση 2:

Το πραγματικό sequence number είναι το 883,061,785 αλλά το wireshark για να μας διευκολύνει το έχει και ως 0. Η πραγματική τιμή του ACKnowledged πεδίου είναι 232,129,013 αλλά η απλοποιημένη μορφή του είναι 1. Τα flags που ενεργοποιούνται στα control bits είναι το SYN bit και το ACK bit για να δείξει πως το περιεχόμενο σε αυτά τα πεδία είναι έγκυρο και χρειάζεται “ανάγνωση”.

Ερώτηση 3:

Το sequence number που περιέχει την εντολή HTTP POST είναι 232,129,013 ο πραγματικός και 1 ο σχετικός και είναι το τέταρτο πακέτο (No. 4).

Ερώτηση 4:

Ο sequence number των 6 segments που στάλθηκαν μαζί με το POST και ο αριθμός των πακέτων που είναι ACKs αυτών των πακέτων είναι ως εξής:

<i>No. of segments</i>	<i>Sequence number</i>	<i>No. of ACK segments</i>
No.4	1	No.6
No.5	566	No.9
No.7	2,026	No.12
No.8	3,486	No.14
No.10	4,946	No.15
No.11	6,406	No.16

Το πακέτο No.4 στέλνει Sequence number 1 και το μέγεθος του είναι 565 bytes, άρα περιμένουμε απο τον server να μας στείλει κάποιο πακέτο που το Acknowledgment number θα είναι $1+565=566$ και αυτό το πακέτο είναι το No.6. Και έτσι βρίσκουμε και τα υπόλοιπα.

<i>Segment</i>	<i>ACK time – Sent time = RTT(seconds)</i>
Segment 1	$0.053937 - 0.026477 = 0.02746$
Segment 2	$0.077294 - 0.041737 = 0.035557$
Segment 3	$0.124085 - 0.054026 = 0.070059$
Segment 4	$0.169118 - 0.054690 = 0.114428$
Segment 5	$0.217299 - 0.077405 = 0.139894$
Segment 6	$0.267802 - 0.078157 = 0.189645$

Ερώτηση 5:

Το μέγεθος του πρώτου segment (No.4) είναι 565 bytes, ενώ τα άλλα 5 segments (No.5, No.7, No.8, No.10, No.11) είναι 1,460 bytes το οποίο είναι το Maximum Segment Size που συμφωνήθηκε κατά το 3 way handshake και περιέχεται στο δεύτερο πακέτο (No.2) στα Options > TCP Option – Maximum segment size: 1,460 bytes.

Ερώτηση 6:

Το πακέτο No.2 το οποίο έχει σταλθεί από τον server στον client, διαφιμίζει πως η ελάχιστη τιμή του διαθέσιμου buffer για όλη την επικοινωνία είναι 5,840 bytes στο Window size value. Ο αποστολέας δεν στραγγαλίζεται ποτέ.

Ερώτηση 7:

Δεν υπάρχουν retransmitted segments διότι στο διάγραμμα Time Sequence (Steves) ο άξονας Y ή αλλιώς Sequence number αυξάνεται όσο αυξάνεται ο χρόνος και δεν υπάρχει κάποιο σημείο που το sequence number να είναι μικρότερο από το προηγούμενο sequence number όσο αυξάνεται ο χρόνος. Ούτε το πακέτο No.3 με το πακέτο No.4 δεν είναι επαναμετάδοση διότι μπορεί να έχουν το ίδιο Sequence number αλλά το πακέτο No.3 έχει μέγεθος 0, οπότε $1+0=1$ άρα δεν υπάρχει κανένα retransmitted segment.

Ερώτηση 8:

Ο παραλήπτης σε ένα ACK segment επιβεβαιώνει τις περισσότερες φορές 1,460 bytes. Υπάρχουν περιπτώσεις που ο παραλήπτης έστειλε ACK segment μια φορά για 2 segment που έλαβε. Τα πακέτα No.54 και No.55 πείραν και τα δύο μια αθροιστική απάντηση (η οποία υποδικνίζει πως και τα δυο λήφθηκαν σωστά) με το ACK segment No.60 όπως και τα πακέτα No.56 και No.57 πείραν ένα ACK segment το No.61. Κανονικά θα έπρεπε πριν από το πακέτο No.60 ο παραλήπτης να έχει στείλει και άλλο πακέτο με ACK number 36,509 αλλά ενδεχόμενος το ACK segment χάθηκε και δεν έφτασε ποτέ στον client αλλά το επόμενο ACK segment (No.60) που πήγε στον client επιβαιβώνει και το προηγούμενο, διότι είχε ACK number 37,969 που πάει να πει πως μέχρι εκεί ο server τα είχε λάβει όλα χωρίς κάποιο σφάλμα ή κάποιο να χάθηκε στον δρόμο.

Ερώτηση 9:

Η ρυθμαπόδοση της TCP συνδέσης είναι τα συνολικά bytes διά τον συνολικό χρόνο. Τα συνολικά bytes θα τα βρούμε αφαιρώντας από το τελευταίο ACK segment (No.202) τον αριθμό του acknowledgment number (164,091) μείων τον αριθμό του πρώτου TCP segment (No.4) τον sequence number (1) δηλαδή μεταδόθηκαν 164,090 bytes. Ο συνολικός χρόνος είναι η αφαίρεση του χρόνου αυτόν των δύο πακέτων δηλαδή $5.455830 - 0.026477 = 5.429353$. Οπότε η ρυθμαπόδοση είναι $164,090 / 5.429353 = 30,222.754 \text{ byte/second}$.

Ερώτηση 10:

Το slow start αρχίζει απο το πακέτο No.1 και τελειώνει στο πακέτο No.23, δηλαδή την χρονική περίοδο απο 0 έως 0.309553 second. Μετά απο εκεί μέχρι και το τέλος είναι το congestion avoidance για το συγκεκριμένο παράδειγμα.

ROADMAP LAB 5**Εξάσκηση: TCP sequence numbers**

Εφόσον έχουμε Sequence number 156 και το MSS είναι 3 bytes τότε το πρώτο segment θα στείλει 3 bytes με Sequence number 156 και το δεύτερο segment θα στείλει 3 bytes με Sequence number 159.

Εξάσκηση: Σενάριο: Εκκίνηση επικοινωνίας HTTP server, HTTP client

Η εκκίνηση επικοινωνίας για το TCP ονομάζεται 3-way-handshake, όπου ο client θέλει να εκκίνησει μια επικοινωνία με έναν server και επειδή είναι connection-oriented πρέπει να συμφωνήσουν σε κάποια συγκεκριμένα πράγματα (όπως το window scale και το MSS) πριν ξεκινήσουν την επικοινωνία της μεταφοράς δεδομένων της εφαρμογής. Ο client περιμένει μια απάντηση ACK segment απο την πλευρά του server και εκείνος απαντάει με μια αντίστοιχη ανταπόκριση.

SYN segment

Client -----> Server SP = 49152, DP = 80, SEQ no = 50, ACK no = NULL

Flags [SYN = 1, ACK = 0, FIN = 0], Δεδομένα Εφαρμογής = 0

Ο client στέλνει ένα SYN segment για να προσπαθήσει να συγχρονιστεί με τον server. Το SP πρέπει να είναι μια διαθέσιμη τιμή πάνω απο 49151 μέχρι $2^{16} - 1 = 65535$ διότι αυτές είναι οι dynamic ports, κάτω απο το 1023 είναι οι well known ports, και οι ενδιαμέσες χρησιμοποιούνται απο χειριστές, για κάποιο service. Το DP είναι 80 επειδή είναι HTTP server και είναι well known port. Το SEQ no είναι ένας τυχαίος αριθμός μέχρι $2^{32} - 1$ αλλά όχι 0. Το ACK no είναι NULL επειδή δεν έχουμε λάβει κάποιο SEQ no απο τον Server έτσι ώστε να ξέρουμε ποιος θέλει να είναι ο αριθμός της ακολουθίας του. Τα flags είναι μόνο το SYN 1 και όλα τα άλλα 0 γιατί θέλουμε να συγχρονιστούμε με τον server, το ACK είναι 0 επειδή δεν είναι έγκυρο το περιεχόμενο στο ACK no, και το FIN είναι 0 γιατί δεν θέλουμε να λήξουμε την σύνδεση. Τα δεδομένα εφαρμογής είναι 0 διότι δεν μεταφέρονται δεδομένα εφαρμογής επειδή είμαστε στην διαδικασία 3-way-handshake και προσπαθούμε να συγχρονιστούμε με τον server για να μεταφέρουμε τα δεδομένα εφαρμογής.

SYN/ACK segment

Client<----- Server SP = 80, DP = 49152, SEQ no = 20, ACK no = 51

Flags [SYN = 1, ACK = 1, FIN = 0], Δεδομένα Εφαρμογής = 0

Ο server στέλνει ένα SYN/ACK segment στον client (εφόσον θέλει να συνδεθεί). Το SP είναι ο αριθμός πόρτας που βρήκαμε τον server ή αλλιώς είναι ο αριθμός πόρτας του server. Το DP είναι ο αριθμός πόρτας του client τον οποίο γνωρίζουμε από το SYN segment που στάλθηκε. Το SEQ no είναι ένας τυχαίος αριθμός μέχρι $2^{32} - 1$ αλλά όχι 0. Το ACK no είναι ο αριθμός του SEQ no που έστειλε ο client στο SYN segment +1 επειδή το SYN flag ήταν 1. Τα Flags είναι όλα 1 εκτός από το FIN. Το SYN είναι 1 γιατί θέλει και ο server να συγχρονιστεί με τον client. Το ACK είναι 1 γιατί τα περιεχόμενα του ACK no είναι έγκυρα. Το FIN είναι 0 γιατί δεν θέλουμε να τερματίσουμε την επικοινωνία.

ACK segment

Client -----> Server SP = 49152, DP = 80, SEQ no = 51, ACK no = 21

Flags [SYN = 0, ACK = 1, FIN = 0], Δεδομένα Εφαρμογής = 0

Ο client στέλνει ένα segment ενημερώσεις στον server ότι έλαβε το segment. Το SP είναι ο διαθέσιμος αριθμός πόρτας του client. Το DP είναι ο αριθμός πόρτας του HTTP server όπου πλέον τον ξέρουμε και από το προηγούμενο segment. Το SEQ no είναι 51 διότι το προηγούμενο segment που έστειλε ο client στον server ήταν SYN flag = 1, οπότε έχουμε $50+1=51$. Το ACK no είναι 21 γιατί το segment που έστειλε ο server στον client είχε το SYN flag = 1, οπότε έχουμε $20+1=21$. Τα Flags είναι όλα 0 εκτός από το ACK το οποίο περιέχει έγκυρη πληροφορία, το SYN είναι 0 γιατί καταφέραμε να εγκαθιδρύσουμε επικοινωνία μαζί με τον server και το FIN 0 γιατί δεν θέλουμε να τερματίσουμε την σύνδεση.

Εξάσκηση: Σενάριο: Επικοινωνία HTTP client/server

SP =49154, DP = 80 SEQ no=10, ACK no =NULL

----->

Flags [SYN = 1, ACK = 0 , FIN = 0] , Δεδομένα Εφαρμογής = 0 bytes

SP = 80, DP = 49154 SEQ no=0 , ACK no =11

<-----

Flags [SYN = 1, ACK = 1, FIN = 0] , Δεδομένα Εφαρμογής = 0 bytes

SP =49154 , DP = 80 SEQ no= 11, ACK no =1

----->

Flags [SYN = 0, ACK = 1, FIN =0] , Δεδομένα Εφαρμογής = 0 bytes

SP = 49154, DP = 80 SEQ no= 11, ACK no =1

----->

Flags [SYN = 0, ACK = 1, FIN =0] , Δεδομένα Εφαρμογής = 40 bytes

SP = 80, DP = 49154 SEQ no= 1, ACK no = 51
 <-----
 Flags [SYN = 0, ACK =1 , FIN =0] , Δεδομένα Εφαρμογής = 500 bytes

 SP =49154, DP = 80 SEQ no=51 , ACK no =501
 ----->
 Flags [SYN = 0, ACK =1 , FIN =0] , Δεδομένα Εφαρμογής = 0 bytes

 SP = 80, DP = 49154 SEQ no=501 , ACK no =51
 <-----
 Flags [SYN = 0, ACK =1 , FIN =0] , Δεδομένα Εφαρμογής = 500 bytes

 SP = 80, DP = 49154 SEQ no= 1001, ACK no =51
 <-----
 Flags [SYN = 0, ACK =1 , FIN =0] , Δεδομένα Εφαρμογής = 500 bytes

 SP =49154, DP = 80 SEQ no= 51, ACK no =1001
 ----->
 Flags [SYN = 0, ACK =1 , FIN =0] , Δεδομένα Εφαρμογής = 0 bytes

 SP =49154, DP = 80 SEQ no= 51, ACK no =1501
 ----->
 Flags [SYN = 0, ACK =1 , FIN =0] , Δεδομένα Εφαρμογής = 0 bytes

 SP = 80, DP = 49154 SEQ no= 1501, ACK no =51
 <-----
 Flags [SYN = 0, ACK = 1, FIN =1] , Δεδομένα Εφαρμογής = 0 bytes

 SP =49154, DP = 80 SEQ no= 51, ACK no =1501
 ----->
 Flags [SYN = 0, ACK =1 , FIN = 1] , Δεδομένα Εφαρμογής = 0 bytes

 SP = 80, DP = 49154 SEQ no=1501 , ACK no =51
 <-----
 Flags [SYN = 0, ACK =1 , FIN = 0] , Δεδομένα Εφαρμογής = 0 bytes