



**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ ΣΧΟΛΗ  
ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ  
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ Ακ. έτος 2023-  
2024 9ο εξάμηνο, Φοιτητής: Αναστασιάδης  
Βασίλειος ΑΜ:03119954**

**Πρωτόκολλα TCP και UDP**

1 Μετάδοση δεδομένων με TCP

1. Host 10.3.26.66
2. `ip.dst == 1.1.1.1` or `ip.dst == 2.2.2.2` or `ip.dst == 147.102.40.1`
3. 23
4. `tcp.port == 23`
5. .... ..1. = Syn: Set
6. 7 προσπάθειες (linux)
7. Το μόνο που παρατηρώ είναι πως τα πακέτα δεν φεύγουν ανά τακτά χρονικά διαστήματα αλλά η χρονική διαφορά της αποστολής ενός πακέτου από το επόμενο είναι αισθητή
8. Καμία διαφορά
9. Μόνο το πρώτο βήμα παρατήρησα
10. Εγκαταλείπει την προσπάθεια
11. `ip.host == 147.102.40.1` and `tcp.port == 23`
12. 1
13. Παρατηρώ ότι επιστρέφεται απάντηση (RST, ACK) από τον server και δεν γίνονται προσπάθειες για retransmission.
14. .... ..1 .... = Acknowledgment: Set  
..... ..1.. = Reset: Set
15. .... ..1.. = Reset: Set
16. 20 bytes η επικεφαλίδα και δεν έχει data
17. Source Port: 2 bytes  
Destination Port: 2 bytes  
Sequence number: 4 bytes  
Acknowledgment number: 4 bytes  
Header Length: 4 bits  
Flags: 12 bits

Window size value: 2 bytes

Checksum: 2 bytes

Urgent pointer: 2 bytes

18. Σύμφωνα με την ιστοσελίδα λέγεται data offset αλλά στο Wireshark λέγεται header length.
19. Η τιμή του πεδίου είναι 0101(HEX) --> 5(dec), άρα θα έχει 5\*4 bytes μήκος η επικεφαλίδα του TCP πακέτου.
20. Όχι
21. Από το total length του IPv4 - επικεφαλίδα TCP : 40-20=20 bytes
22. 40 bytes
23. Ναι γιατί αυτό το πακέτο έχει και data

## 2 Εγκατάσταση σύνδεσης, μεταφορά δεδομένων και απόλυση σύνδεσης TCP

1. host 46.176.117.147 and tcp
2. 21
3. 20
4. tcp.port == 21
5. 3 πακέτα
6. 2 (SYN,ACK)
7. 40 bytes για τα πρώτα 2 τεμάχια TCP και 32 bytes για το 3ο.
8. 0
9. 0.005370552 seconds
10. Ναι [iRTT: 0.005370552 seconds]
11. Sequence Number: 1 (relative sequence number)
12. Acknowledge number = sequence number + 1
13. Το Acknowledge number παραμένει ίδιο με αυτό της αποδοχής σύνδεσης από τον εξυπηρετητή και το sequence number αυξάνεται κατά 1.
14. 0
15.  $2^{(4*8)}-1 \rightarrow 2^{32}$
16. tcp.port==21 and (tcp.flags.syn==1 or (tcp.flags.ack==1 and tcp.seq==1 ))
17. Window: 65535
18. Window: 65535
19. Window
20. Kind: Window Scale (3)
21. TCP Option - Window scale: 2 (multiply by 4)
22. TCP Option - Maximum segment size: 1460 bytes
23. MSS = MTU- 40=1500-40=1460 bytes
24. TCP Option - Maximum segment size
25. TCP Option - Maximum segment size: 536 bytes
26. MSS = MTU- 40=576-40=536 bytes
27. 536 bytes

28. .... 1 = Fin: Set
29. Η πλευρά του εξυπηρετητή
30. 2
31. 32 bytes
32. 0 bytes
33. Είναι επικεφαλίδα IPv4 + επικεφαλίδα TCP = 20 + 32 = 52 bytes
- Είναι επικεφαλίδα Ethernet + επικεφαλίδα IPv4 + επικεφαλίδα TCP = 14 + 20 + 32 = 66 bytes
34. Είναι επικεφαλίδα IPv4 + επικεφαλίδα TCP = 20 + 32 = 52 bytes
- Είναι επικεφαλίδα Ethernet + επικεφαλίδα IPv4 + επικεφαλίδα TCP = 14 + 20 + 32 = 66 bytes
35. Από τον υπολογιστή: 142 bytes
- Από τον εξυπηρετητή: 781 bytes
36. sequence και acknowledgement numbers
37. tcp.port==20
38. Ο εξυπηρετητής ανακοινώνει MSS=536 bytes και ο υπολογιστής 1460 bytes
39. 1460 bytes
40. [The RTT to ACK the segment was: 0.000187000 seconds]
41. Ναι
42. 27
43. 27
44. [Calculated window size: 65984]
45. Όχι
46. Παραμένει σταθερή
47. Είτε θα απέστειλε ένα ICMP μήνυμα σφάλματος, είτε θα αναμένει να λάβει ένα παράθυρο (window) μεγαλύτερο από το μηδέν, προκειμένου να επιτραπεί η συνέχιση της αποστολής των δεδομένων.
48. Μέγεθος πλαισίου 1445 bytes
- Επικεφαλίδα Ethernet: 14 bytes
- Επικεφαλίδα IP: 20 bytes
- Επικεφαλίδα TCP: 32 bytes
49. Ναι
50. Θα γινόταν θρυμματισμός του πακέτου από κάποιον ενδιάμεσο κόμβο
51. 61441 bytes από τον εξυπηρετητή
- 0 bytes από τον υπολογιστή
52. 61441 bytes / 153 s = 401
53. Όχι

### 3 Αποφυγή συμφόρησης στο TCP

1. tcp.port == 21
2. 46.176.117.147
3. [The RTT to ACK the segment was: 0.000048000 seconds] Είναι πιο γρήγορο
4. Αρχικά, αποστέλλεται μικρή ποσότητα, με τον αριθμό των αποστελλόμενων τεμαχίων να αυξάνεται καθώς περνά ο χρόνος RTT.
5. Έστειλε 4 τεμάχια, που είναι σύμφωνο με ότι προβλέπει το RFC5681
6. Δευτερο:6 Τριτο:10 Τεταρτό:16
7. Πρωτο:1 Δευτερο:2 Τριτο:3 αυξάνεται γραμικά
8. Ναι, είναι αρκετά παρόμοιο

### 4 Μετάδοση δεδομένων με UDP

1. Udp
2. Source Port: 2 bytes  
Destination Port: 2 bytes  
Length: 2 bytes  
Checksum: 2 bytes
3. 8 bytes
4. μέγεθος πακέτου (333 bytes) - μήκος επικεφαλίδας IPv4(20 bytes)  
= 312 bytes
5. Εκφράζει το μήκος δεδομενογράμματος
6. 8 bytes
7. Το ελάχιστο μέγεθος μηνύματος που μπορεί να μεταφερθεί από ένα πακέτο IPv4 χρησιμοποιώντας το πρωτόκολλο UDP είναι 8 bytes. Αυτό το μέγεθος περιλαμβάνει μόνο το κεφάλι του πακέτου UDP, το οποίο έχει μέγεθος 8 bytes. Το μέγιστο μέγεθος μηνύματος που μπορεί να μεταφερθεί από ένα πακέτο IPv4 χρησιμοποιώντας το πρωτόκολλο UDP είναι 65.535 bytes. Αυτό το μέγεθος περιορίζεται από το μέγιστο μέγεθος του πεδίου δεδομένων του πακέτου IPv4, το οποίο έχει μέγεθος 65.535 bytes.
8. 576- μήκος επικεφαλίδας IPv4 (20) - μήκος επικεφαλίδας UDP (8 bytes) = 548 bytes
9. UDP και QUIC
10. Dns
11. 192.168.2.11
12. Source Port: 51053  
Destination Port: 53
13. Source Port: 53  
Destination Port: 51053
14. Port 53

