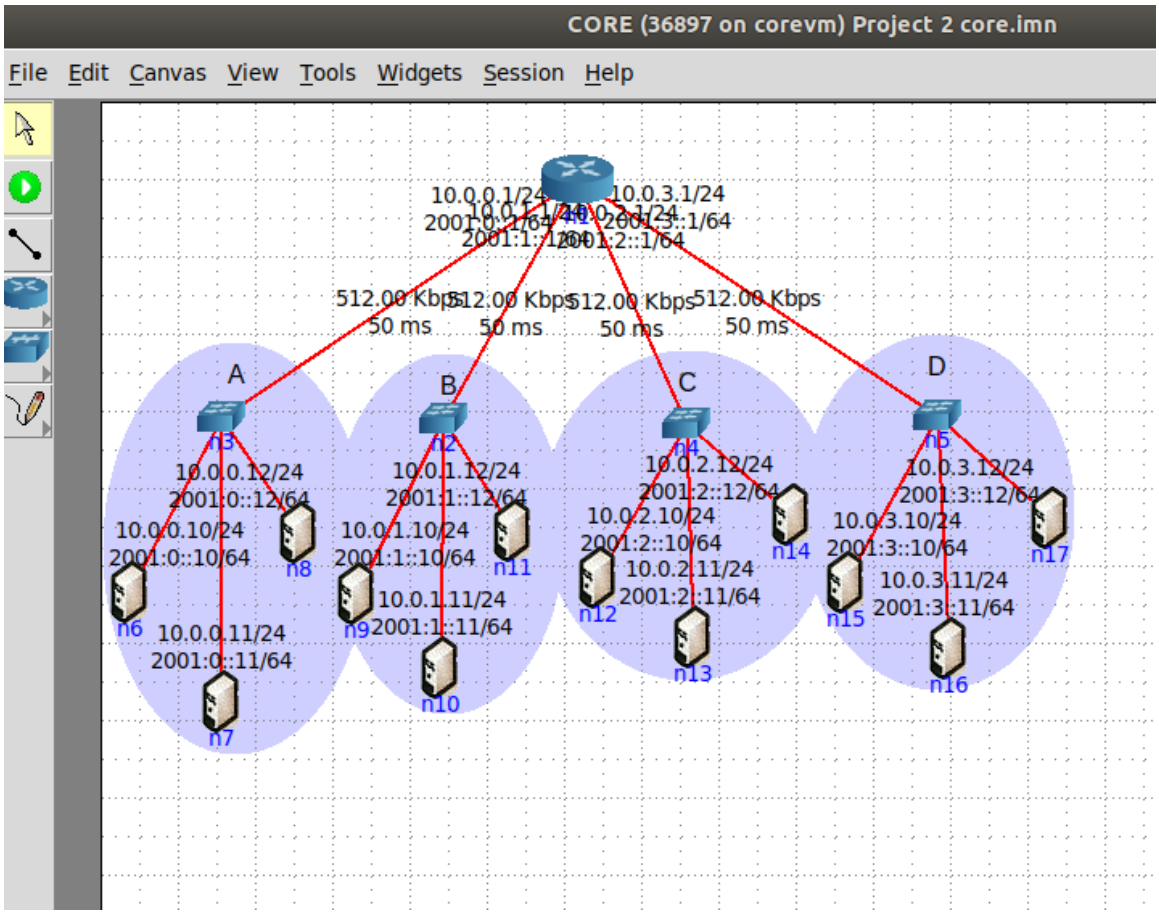
Δίκτυα Επικοινωνιών

2η εργασία - Network Emulation on CORE + Wireshark based Traffic Analysis

Παλαιολόγος Κλεάνθης p3180136

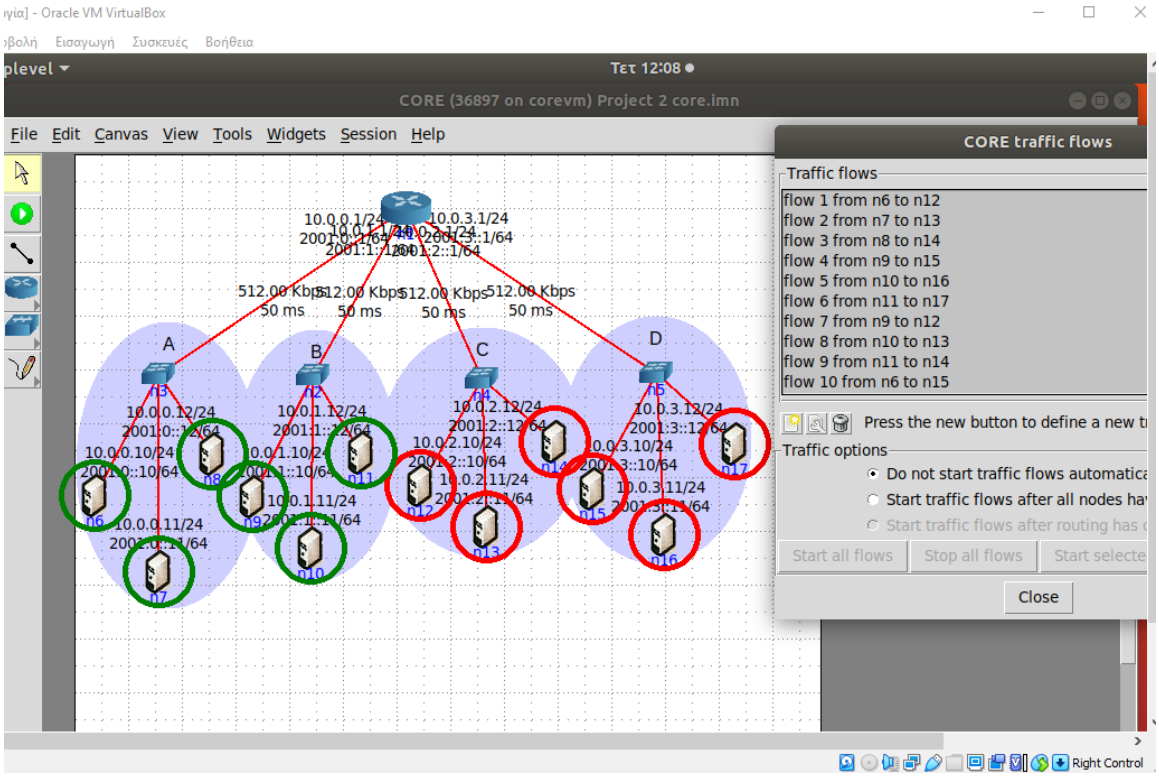
Μαυρίδης Ιωάννης p3180106

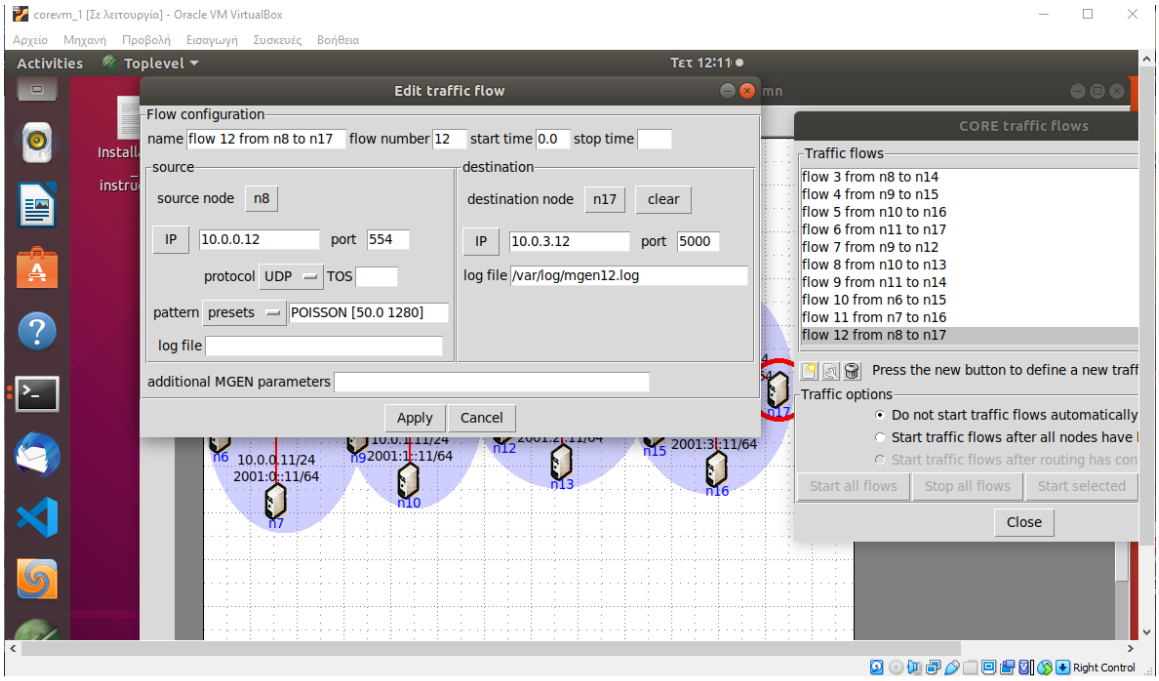
1)Παρακάτω παρουσιάζουμε το ενσύρματο δίκτυο που δημιουργήσαμε στο Core το οποίο αποτελείται από 4 υποδίκτυα (A,B,C,D) και έχει συνολικά 12 hosts. Επίσης συμφωνεί με την προϋπόθεση η οποία αναφέρει πως πρέπει να απλωθεί σε έναν χώρο 1km x 1km.



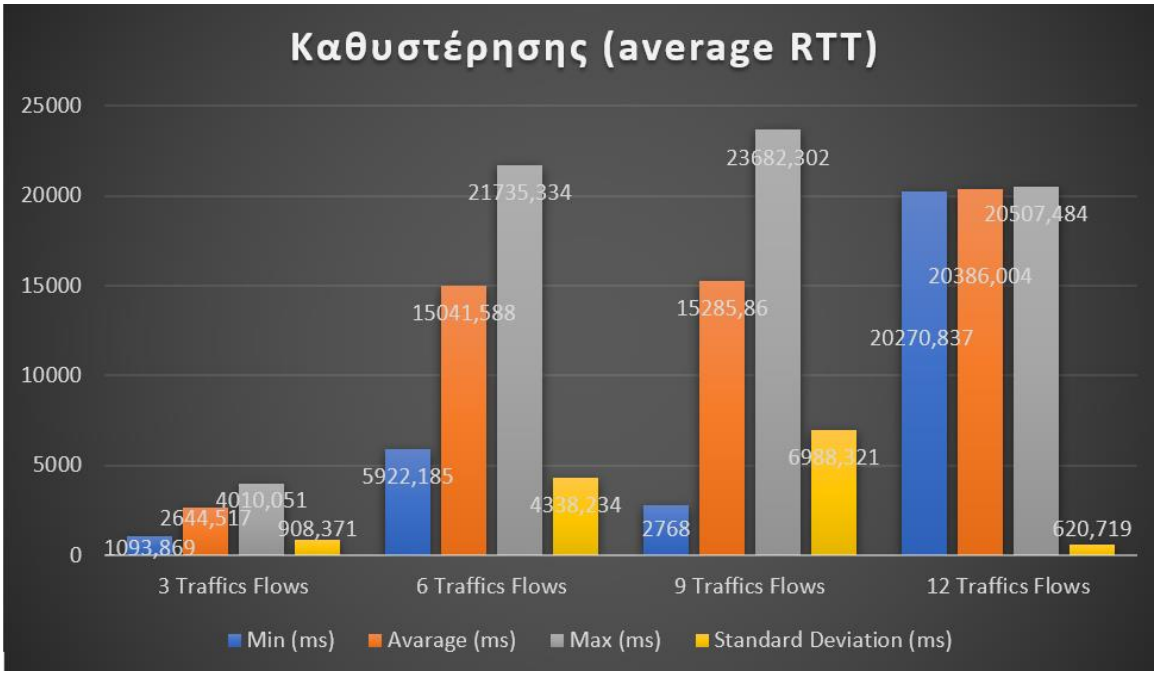
2)Έχουμε δημιουργήσει 12 traffic flows από τα 2 υποδίκτυα A,B προς τα υπόλοιπα C,D. Πιο συγκεκριμένα από όλα τα host του A προς όλα τα host του C και D και ομοίως από όλα τα host του B προς όλα τα host του C και D. Τα traffic flows να είναι κατάλληλα σχεδιασμένα και υπολογισμένα ώστε να εξομοιώνουν: αποστολή αρχείων, push-to-talk voice communications και video streaming μεταξύ κόμβων. Αυτό το πετύχαμε βάζοντας τα κατάλληλα πρωτόκολλα UDP και TCP, επιλέγοντας τα κατάλληλα source ports τα οποία είναι τα 5060, 21 και 554 που

αντιστοιχούν σε voice communications, αποστολή αρχείων και video streaming κατά σειρά. Destination port έχουμε αφήσει το default 5000. Τα patterns που έχουμε τοποθετήσει είναι ποικιλόμορφα και συγκεκριμένα έχουμε Burst από 75 kbps ως 256 kbps και POISSON από 100 kbps ως 512 kbps.

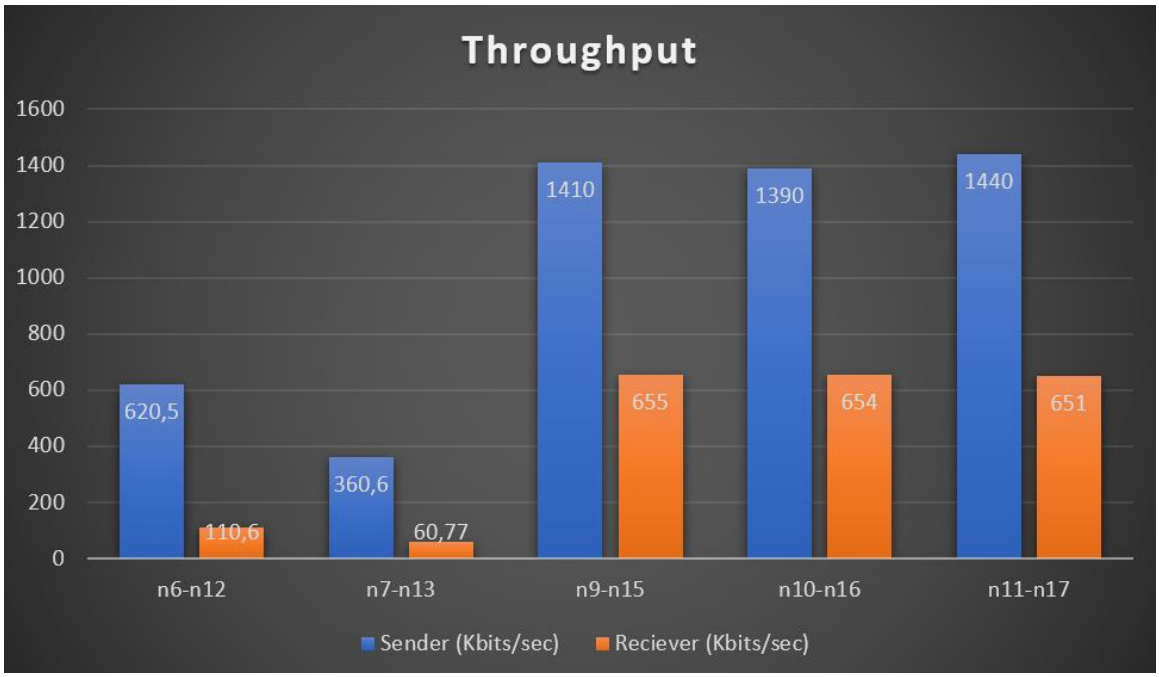




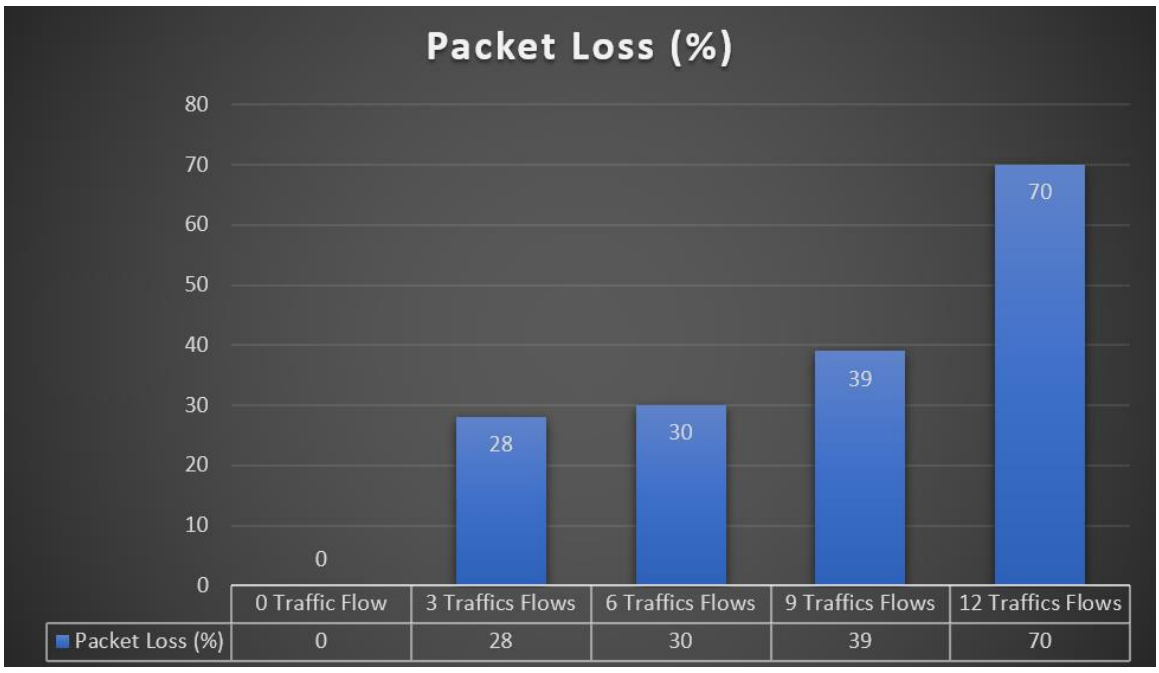
3) i) Μετρήσεις καθυστέρησης (average RTT) ανά traffic flow.



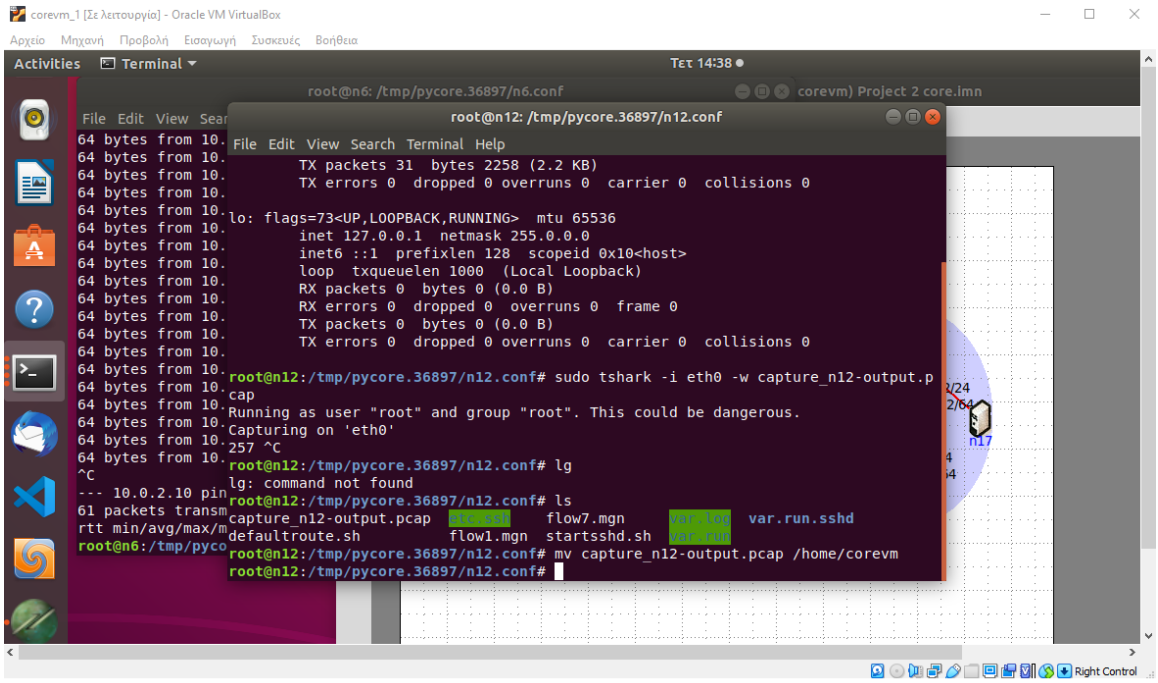
ii) Μετρήσεις throughput ανά source-destination ζευγάρι κόμβων.



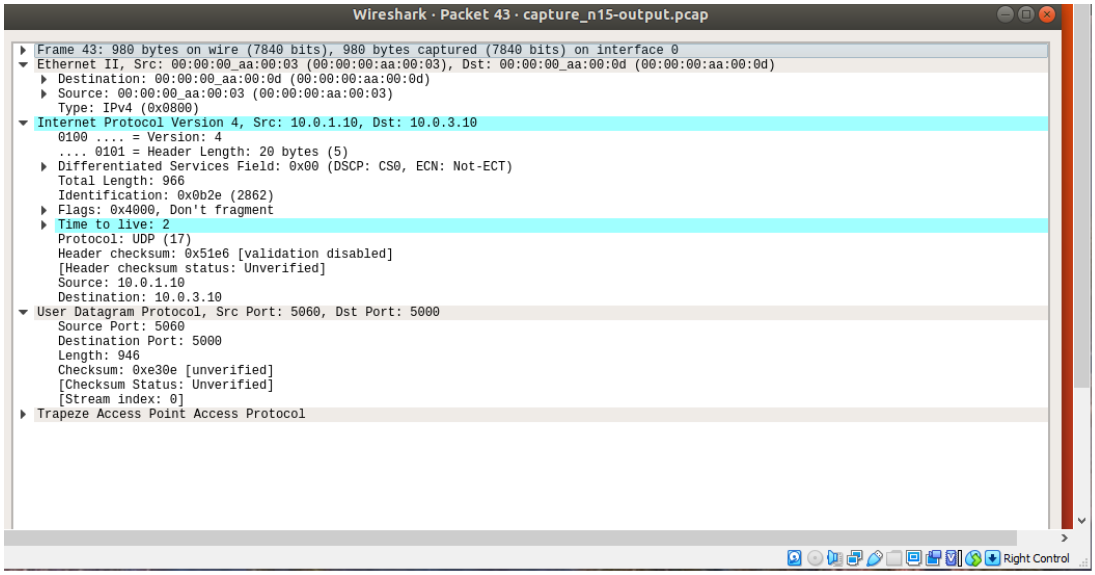
iii) Μετρήσεις packet loss ανά traffic flow.

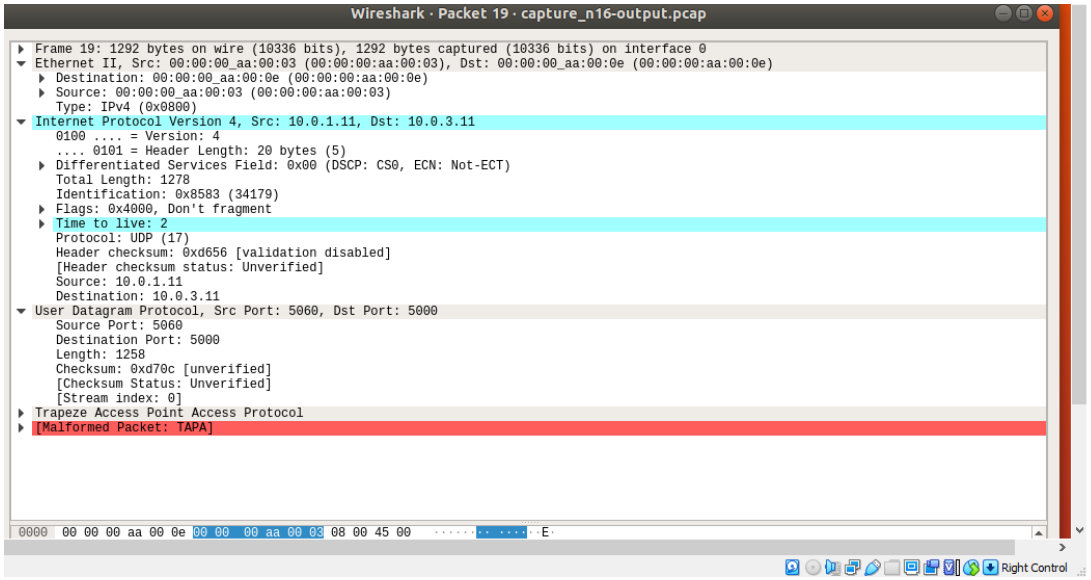


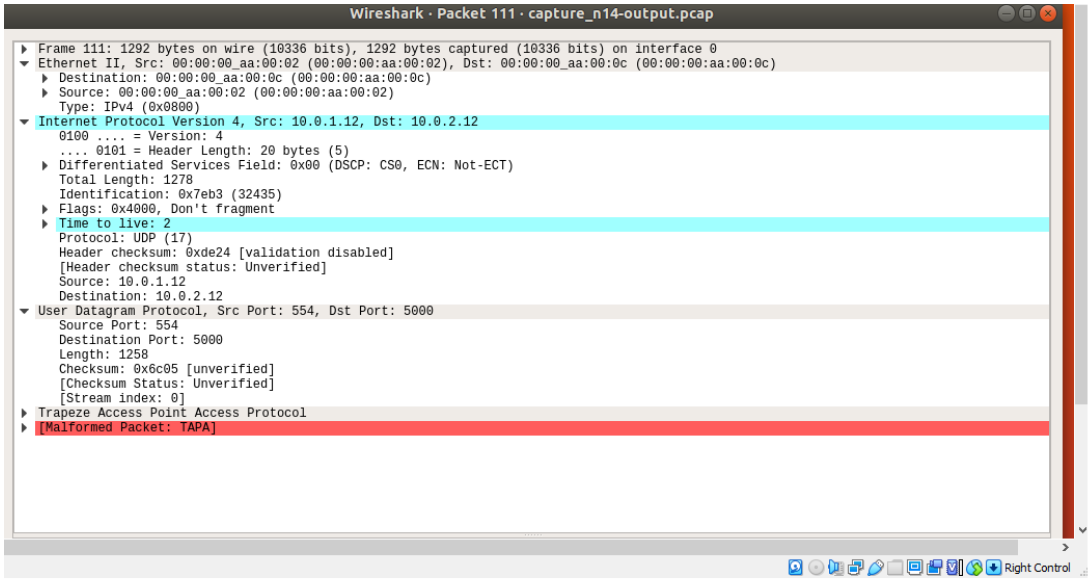
4) Στο παρόν ερώτημα έχουν γίνει καταγραφές περίπου ενός λεπτού για 6 traffic flows τα οποία έχουν κατάληξη στα node 12-n17 τα οποία ανήκουν στα υποδίκτυα C και D. Ακολουθώντας τις οδηγίες του φροντιστηρίου έγιναν οι καταγραφές με την εξής διαδικασία, αρχικά τρέχαμε την εντολή ifconfig στο cmd του destination node, έπειτα χρησιμοποιήσαμε ping για να δημιουργήσουμε traffic flow στους 2 hosts και tshark για να κάνουμε capture στο interface network του εκάστοτε destination node. Στην συνέχεια, τρέχουμε τις 2 εντολές και μετά από λίγο τις διακόπτουμε. Τέλος, αποθηκεύουμε το output file σε έναν άλλον φάκελο στο corevm ως .pcap file.

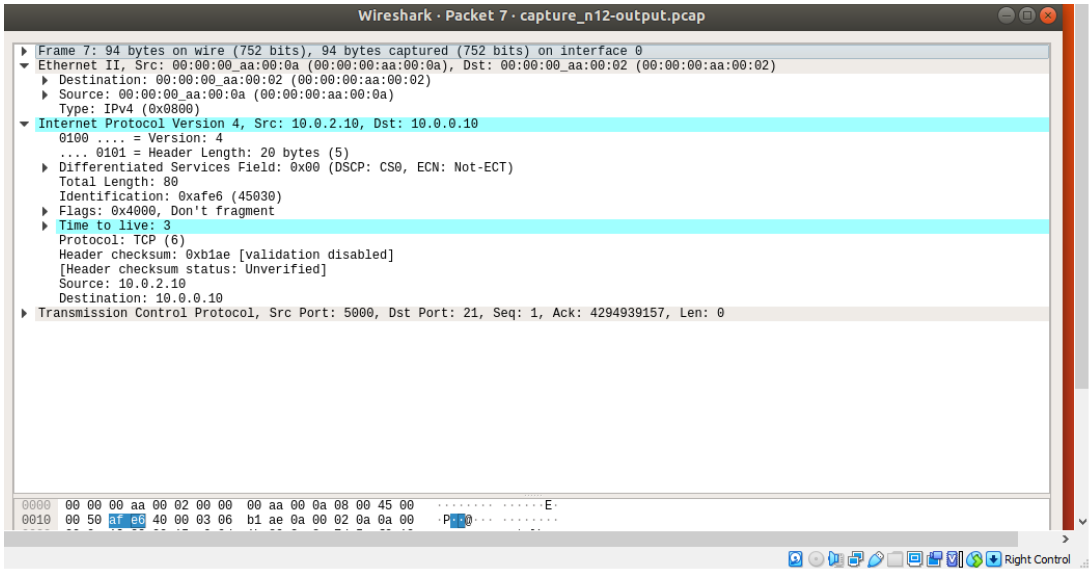


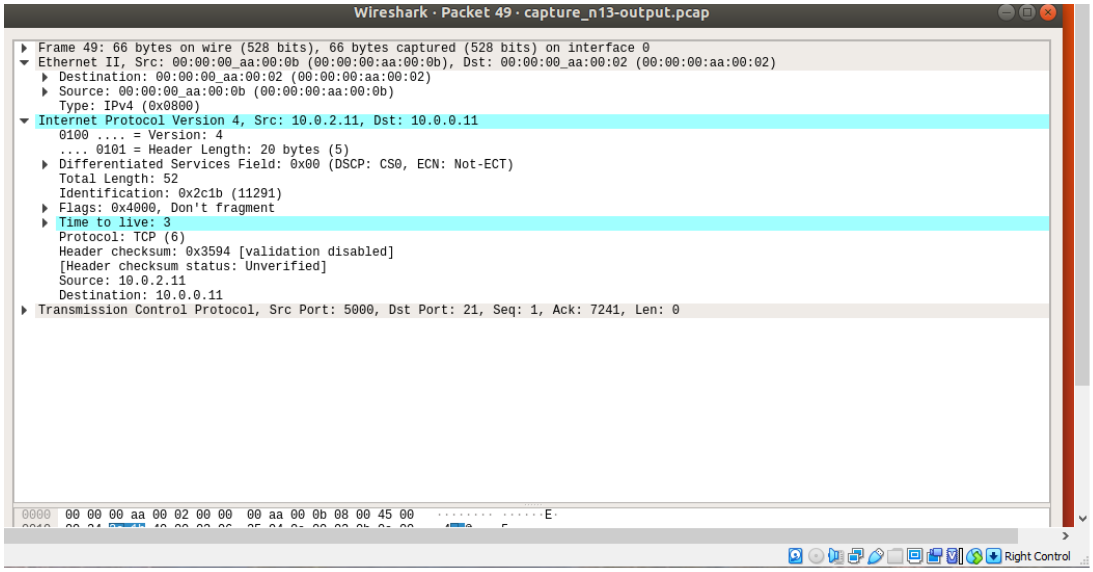
5)Αναλύσαμε τις παραπάνω καταγραφές με την χρήση του Wireshark και παρακάτω παρουσιάζονται τα αποτελέσματα.


Wireshark · Packet 43 · capture_n15-output.pcap
Frame 43: 980 bytes on wire (7840 bits), 980 bytes captured (7840 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:03 (00:00:00:aa:00:03), Dst: 00:00:00_aa:00:0d (00:00:00:aa:00:0d)
Destination: 00:00:00_aa:00:0d (00:00:00:aa:00:0d)
Source: 00:00:00_aa:00:03 (00:00:00:aa:00:03)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.0.1.10, Dst: 10.0.3.10
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 966
Identification: 0x0b2e (2862)
Flags: 0x4000, Don't fragment
Time to live: 2
Protocol: UDP (17)
Header checksum: 0x51e6 [validation disabled]
[Header checksum status: Unverified]
Source: 10.0.1.10
Destination: 10.0.3.10
User Datagram Protocol, Src Port: 5060, Dst Port: 5000
Source Port: 5060
Destination Port: 5000
Length: 946
Checksum: 0xe30e [unverified]
[Checksum Status: Unverified]
[Stream index: 0]
Trapeze Access Point Access Protocol
Right Control


Wireshark · Packet 19 · capture_n16-output.pcap
Frame 19: 1292 bytes on wire (10336 bits), 1292 bytes captured (10336 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:03 (00:00:00:aa:00:03), Dst: 00:00:00_aa:00:0e (00:00:00:aa:00:0e)
Destination: 00:00:00_aa:00:0e (00:00:00:aa:00:0e)
Source: 00:00:00_aa:00:03 (00:00:00:aa:00:03)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.0.1.11, Dst: 10.0.3.11
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 1278
Identification: 0x8583 (34179)
Flags: 0x4000, Don't fragment
Time to live: 2
Protocol: UDP (17)
Header checksum: 0xd656 [validation disabled]
[Header checksum status: Unverified]
Source: 10.0.1.11
Destination: 10.0.3.11
User Datagram Protocol, Src Port: 5060, Dst Port: 5000
Source Port: 5060
Destination Port: 5000
Length: 1258
Checksum: 0xd70c [unverified]
[Checksum Status: Unverified]
[Stream index: 0]
Trapeze Access Point Access Protocol
[Malformed Packet: TAPA]
0000 00 00 00 aa 00 0e 00 00 00 aa 00 03 08 00 45 00
Right Control


Wireshark · Packet 111 · capture_n14-output.pcap
Frame 111: 1292 bytes on wire (10336 bits), 1292 bytes captured (10336 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:02 (00:00:00:aa:00:02), Dst: 00:00:00_aa:00:0c (00:00:00:aa:00:0c)
Destination: 00:00:00_aa:00:0c (00:00:00:aa:00:0c)
Source: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.0.1.12, Dst: 10.0.2.12
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 1278
Identification: 0x7eb3 (32435)
Flags: 0x4000, Don't fragment
Time to live: 2
Protocol: UDP (17)
Header checksum: 0xde24 [validation disabled]
[Header checksum status: Unverified]
Source: 10.0.1.12
Destination: 10.0.2.12
User Datagram Protocol, Src Port: 554, Dst Port: 5000
Source Port: 554
Destination Port: 5000
Length: 1258
Checksum: 0x6c05 [unverified]
[Checksum Status: Unverified]
[Stream index: 0]
Trapeze Access Point Access Protocol
[Malformed Packet: TAPA]
Right Control


Wireshark · Packet 7 · capture_n12-output.pcap
Frame 7: 94 bytes on wire (752 bits), 94 bytes captured (752 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:0a (00:00:00:aa:00:0a), Dst: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
Destination: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
Source: 00:00:00_aa:00:0a (00:00:00:aa:00:0a)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.0.2.10, Dst: 10.0.0.10
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 80
Identification: 0xafe6 (45030)
Flags: 0x4000, Don't fragment
Time to live: 3
Protocol: TCP (6)
Header checksum: 0xb1ae [validation disabled]
[Header checksum status: Unverified]
Source: 10.0.2.10
Destination: 10.0.0.10
Transmission Control Protocol, Src Port: 5000, Dst Port: 21, Seq: 1, Ack: 4294939157, Len: 0
0000 00 00 00 aa 00 02 00 00 00 aa 00 0a 08 00 45 00
0010 00 50 af e6 40 00 03 06 b1 ae 0a 00 02 0a 0a 00
Right Control


Wireshark · Packet 49 · capture_n13-output.pcap
Frame 49: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:0b (00:00:00:aa:00:0b), Dst: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
Destination: 00:00:00_aa:00:02 (00:00:00:aa:00:02)
Source: 00:00:00_aa:00:0b (00:00:00:aa:00:0b)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.0.2.11, Dst: 10.0.0.11
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 52
Identification: 0x2c1b (11291)
Flags: 0x4000, Don't fragment
Time to live: 3
Protocol: TCP (6)
Header checksum: 0x3594 [validation disabled]
[Header checksum status: Unverified]
Source: 10.0.2.11
Destination: 10.0.0.11
Transmission Control Protocol, Src Port: 5000, Dst Port: 21, Seq: 1, Ack: 7241, Len: 0
0000 00 00 00 aa 00 02 00 00 00 aa 00 0b 08 00 45 00 ....E.
Right Control