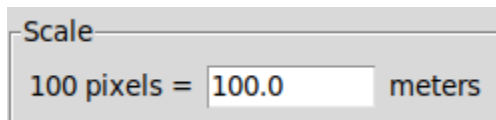


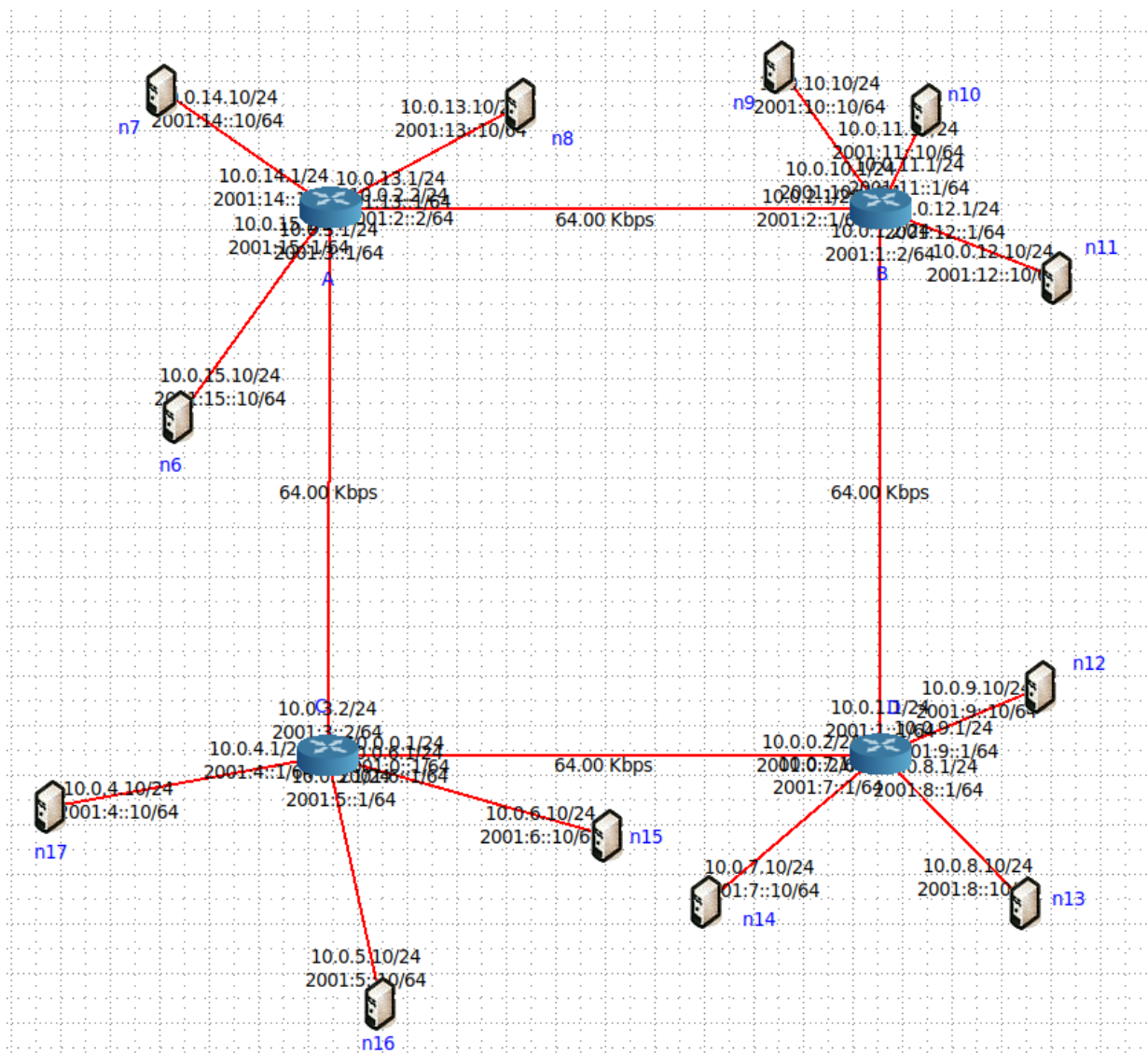
2ή εργασία - Network Emulation on CORE + Wireshark based Traffic Analysis

1)



Για την σχεδίαση του δικτύου μας έχουμε βάλει 4 routers A, B, C, D σε απόσταση 1000 pixels καθώς έχουμε θέσει ότι 100 pixels αντιστοιχούν σε 100 μέτρα.

Έπειτα, τοποθετήσαμε 3 κόμβους σε κάθε router αριθμημένους από το n6 έως το n17 όπως φαίνεται παρακάτω:



2)

Για το emulation του video streaming θα χρειαστεί να προσομοιώσουμε τα εξής:

- HTTPS που χρησιμοποιείται για παράδειγμα απο το Youtube και αντίστοιχες πλατφόρμες στο port 443
- QUIC που χρησιμοποιείται στα video streams το οποίο βασίζεται σε UDP και χρησιμοποιεί dynamic ports τόσο για τον client όσο και για τον server

Οπότε δημιουργούμε τα παρακάτω traffic flows:

http/s for vid streaming n6-n15
http/s for vid streaming n6-n16
http/s for vid streaming n6-n12
QUIC(video stream) emulation n6-n12
QUIC(video stream) emulation n6-n16
QUIC(video stream) emulation n6-n15

- http/s for vid streaming
n6-n12
(subnet A -> subnet D)

name http/s for vid streaming r flow number 5 start time 0.0 stop time	
source	
source node n6	destination
IP 10.0.15.10 port 443	destination node n12 clear
protocol TCP TOS	IP 10.0.9.10 port 7654
pattern presets PERIODIC [10.0 125]	log file /var/log/mgen5.log

- http/s for vid streaming
n12-n6
(subnet D -> subnet A)

name for vid streaming n12-n6 flow number 12 start time 0.0 stop time	
source	
source node n12	destination
IP 10.0.9.10 port 7654	destination node n6 clear
protocol TCP TOS	IP 10.0.15.10 port 443
pattern presets PERIODIC [10.0 1250]	log file /var/log/mgen12.log

- QUIC(video stream)
emulation n6-n12
(subnet A -> subnet D)

name QUIC(video stream) emu flow number 6 start time 0.0 stop time	
source	
source node n6	destination
IP 10.0.15.10 port 2345	destination node n12 clear
protocol UDP TOS	IP 10.0.9.10 port 23444
pattern presets PERIODIC [50.0 1280]	log file /var/log/mgen6.log

- http/s vid streaming n6-n15
(subnet A -> subnet C)

name http/s for vid streaming r flow number 1		start time 0.0	stop time
source		destination	
source node	n6	destination node	n15 clear
IP	10.0.15.10	port	443
protocol	TCP	TOS	
pattern	presets	PERIODIC [10.0 125]	
		IP	10.0.6.10
		port	420
		log file /var/log/mgen1.log	

- http/s vid streaming n15-n6
(subnet C -> subnet A)

name tp/s vid streaming n15-n flow number 13		start time 0.0	stop time
source		destination	
source node	n15	destination node	n6 clear
IP	10.0.6.10	port	420
protocol	TCP	TOS	
pattern	presets	PERIODIC [10.0 1250]	
		IP	10.0.15.10
		port	443
		log file /var/log/mgen13.log	

- QUIC (video stream) emulation n6-n15
(subnet A -> subnet C)

name QUIC(video stream) emu flow number 3		start time 0.0	stop time
source		destination	
source node	n6	destination node	n15 clear
IP	10.0.15.10	port	1754
protocol	UDP	TOS	
pattern	presets	PERIODIC [50.0 1280]	
		IP	10.0.6.10
		port	5032
		log file /var/log/mgen3.log	

- http/s vid streaming n6-n16
(subnet A -> subnet C)

name http/s for vid streaming r flow number 2		start time 0.0	stop time
source		destination	
source node	n6	destination node	n16 clear
IP	10.0.15.10	port	80
protocol	TCP	TOS	
pattern	presets	PERIODIC [1.0 125]	
		IP	10.0.5.10
		port	32000
		log file /var/log/mgen2.log	

- http/s vid streaming n16-n6
(subnet C -> subnet A)

name tp/s vid streaming n16-n flow number 14		start time 0.0	stop time
source		destination	
source node	n16	destination node	n6 clear
IP	10.0.5.10	port	32000
protocol	TCP	TOS	
pattern	presets	PERIODIC [10.0 1250]	
		IP	10.0.15.10
		port	80
		log file /var/log/mgen14.log	

- QUIC (video stream)
emulation n6-n16
(subnet A -> subnet C)

name		QUIC(video stream) emu		flow number	4	start time	0.0	stop time	
source									
source node		n6							
IP	10.0.15.10	port	1936						
protocol		UDP							
TOS									
pattern	presets	PERIODIC [50.0 1280]							
destination									
destination node		n16							
clear									
IP	10.0.5.10	port	4848						
log file		/var/log/mgen4.log							

Για το emulation του file transfer θα προσομοιώσουμε τα εξής:

- FTP (File Transfer Protocol) που χρησιμοποιεί το port 21 (by default) για τον server και οποιοδήποτε διαθέσιμο port μετά το 1024 για τον client. Επίσης θα υποθέσουμε ότι χρησιμοποιεί το TCP

Οπότε δημιουργούμε τα εξής traffic flows:

FTP n8-n14
FTP n9-n12
FTP n8-n15

- FTP n8-n14
(subnet A -> subnet D)

name		FTP n8-n14		flow number	8	start time	0.0	stop time	
source									
source node		n8							
IP	10.0.13.10	port	21						
protocol		TCP							
TOS									
pattern	presets	PERIODIC [50.0 1280]							
destination									
destination node		n14							
clear									
IP	10.0.7.10	port	1028						
log file		/var/log/mgen8.log							

- FTP n14-n8
(subnet D -> subnet A)

name		FTP n14-n8		flow number	15	start time	0.0	stop time	
source									
source node		n14							
IP	10.0.7.10	port	1028						
protocol		TCP							
TOS									
pattern	presets	PERIODIC [1.0 125]							
destination									
destination node		n8							
clear									
IP	10.0.13.10	port	21						
log file		/var/log/mgen15.log							

- FTP n9-n12
(subnet B -> subnet D)

name		SIP ptt n10-n13		flow number	18	start time	0.0	stop time	
source					destination				
source node		n10			destination node		n13		
IP		10.0.11.10			port		5060		
protocol		UDP			TOS				
pattern		presets			BURST [RANDOM 5.0 PE				
					clear		var		
IP		10.0.8.10			port		5060		
					:		1025		
log file					/var/log/mgen18.log				

- FTP n12-n9
(subnet D -> subnet B)

name		FTP n12-n9		flow number	16	start time	0.0	stop time	
source					destination				
source node		n12			destination node		n9		
IP		10.0.9.10			port		1025		
protocol		TCP			TOS				
pattern		presets			PERIODIC [1.0 125]				
					clear				
IP		10.0.10.10			port		21		
log file					/var/log/mgen16.log				

- FTP n8-n15
(subnet A -> C)

name		FTP n8-n15		flow number	9	start time	0.0	stop time	
source					destination				
source node		n8			destination node		n15		
IP		10.0.13.10			port		21		
protocol		TCP			TOS				
pattern		presets			PERIODIC [50.0 1280]				
					clear				
IP		10.0.6.10			port		1420		
log file					/var/log/mgen9.log				

- FTP n15-n8
(subnet C -> subnet A)

Για το emulation του push to talk voice communications θα προσομοιώσουμε τα εξής:

- SIP που χρησιμοποιεί UDP και χρησιμοποιεί ως source και destination port την 5060

Οπότε δημιουργούμε τα εξής traffic flows (βάζουμε burst random 5s, για να προσομοιώσουμε το push to talk που δεν είναι συνεχόμενη επικοινωνία):

SIP ptt n10-n13
SIP ptt n13-n10
SIP ptt n6-n17
SIP ptt n17-n6

- SIP ptt n10-n13
(subnet B -> subnet D)

name SIP ptt n10-n13		flow number 18	start time 0.0	stop time
source		destination		
source node	n10	destination node	n13	clear
IP	10.0.11.10	port	5060	
protocol	UDP	TOS		
pattern	presets	BURST [RANDOM 5.0 PE		
		log file /var/log/mgen18.log		

- SIP ptt n13-n10
(subnet D -> subnet B)

name SIP ptt n13-n10		flow number 19	start time 0.0	stop time
source		destination		
source node	n13	destination node	n10	clear
IP	10.0.8.10	port	5060	
protocol	UDP	TOS		
pattern	presets	BURST [RANDOM 5.0 PE		
		log file /var/log/mgen19.log		

- SIP ptt n6-n17
(subnet A -> subnet C)

name SIP ptt n6-n17		flow number 20	start time 0.0	stop time
source		destination		
source node	n6	destination node	n17	clear
IP	10.0.15.10	port	5060	
protocol	UDP	TOS		
pattern	presets	BURST [RANDOM 5.0 PE		
		log file /var/log/mgen20.log		

- SIP ptt n17-n6
(subnet C -> subnet A)

name SIP ptt n17-n6		flow number 21	start time 0.0	stop time
source		destination		
source node	n17	destination node	n6	clear
IP	10.0.4.10	port	5060	
protocol	UDP	TOS		
pattern	presets	BURST [RANDOM 5.0 PE		
		log file /var/log/mgen21.log		

- SIP ppt n9-n17
(subnet A -> C)

name SIP ppt n9-n17		flow number 10	start time 0.0	stop time
source		destination		
source node	n9	destination node	n17	clear
IP	10.0.10.10	port	5060	
protocol	UDP	TOS		
pattern	presets	BURST [REGULAR 15.0 PI		
		log file /var/log/mgen10.log		

- SIP ppt n17-n9
(subnet C->A)

name SIP ppt n17-n9		flow number 11	start time 0.0	stop time
source			destination	
source node n17			destination node n9 clear	
IP 10.0.4.10	port 5060		IP 10.0.10.10	port 5060
protocol UDP TOS			log file /var/log/mgen11.log	
pattern presets BURST [REGULAR 15.0 P]				

3)

α) Για την μέτρηση του RTT χρησιμοποιούμε την εντολή ping των Linux με παράμετρο -c 10 για να τρέξει για δέκα πακέτα μόνο.

Για να δούμε πως επηρεάζει το traffic flow τις επιδόσεις της σύνδεσης A-C τρέχουμε το ping από έναν κόμβο του υποδικτύου A, στην συγκεκριμένη περίπτωση τον n7, για να φανεί η κίνηση στο δίκτυο. Αν χρησιμοποιούσαμε κάποιον κόμβο από άλλο υποδίκτυο όπως το C, δεν θα υπήρχε κίνηση για να επηρεάσει την απόδοση.

Ping n7-n15 :

Χωρίς traffic flow(Κατάσταση 0):

```
root@n7:/tmp/pycore.41731/n7.conf# ping -c 10 10.0.6.10
PING 10.0.6.10 (10.0.6.10) 56(84) bytes of data:
64 bytes from 10.0.6.10: icmp_seq=1 ttl=62 time=0.054 ms
64 bytes from 10.0.6.10: icmp_seq=2 ttl=62 time=0.079 ms
64 bytes from 10.0.6.10: icmp_seq=3 ttl=62 time=0.069 ms
64 bytes from 10.0.6.10: icmp_seq=4 ttl=62 time=0.073 ms
64 bytes from 10.0.6.10: icmp_seq=5 ttl=62 time=0.076 ms
64 bytes from 10.0.6.10: icmp_seq=6 ttl=62 time=0.068 ms
64 bytes from 10.0.6.10: icmp_seq=7 ttl=62 time=0.069 ms
64 bytes from 10.0.6.10: icmp_seq=8 ttl=62 time=0.063 ms
64 bytes from 10.0.6.10: icmp_seq=9 ttl=62 time=0.067 ms
64 bytes from 10.0.6.10: icmp_seq=10 ttl=62 time=0.070 ms

--- 10.0.6.10 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9223ms
rtt min/avg/max/mdev = 0.054/0.068/0.079/0.012 ms
```

Με http/s for vid streaming n6-n15 && http/s for vid streaming n6-n15 && http/s for vid streaming n15-n6 (Video streaming emulation)(Κατάσταση 1):

```
root@n7:/tmp/pycore.41731/n7.conf# ping -c 10 10.0.6.10
PING 10.0.6.10 (10.0.6.10) 56(84) bytes of data.
64 bytes from 10.0.6.10: icmp_seq=1 ttl=62 time=1977 ms
64 bytes from 10.0.6.10: icmp_seq=2 ttl=62 time=1993 ms
64 bytes from 10.0.6.10: icmp_seq=3 ttl=62 time=1993 ms
64 bytes from 10.0.6.10: icmp_seq=4 ttl=62 time=2026 ms
64 bytes from 10.0.6.10: icmp_seq=5 ttl=62 time=2030 ms
64 bytes from 10.0.6.10: icmp_seq=6 ttl=62 time=1999 ms
64 bytes from 10.0.6.10: icmp_seq=7 ttl=62 time=2000 ms
64 bytes from 10.0.6.10: icmp_seq=8 ttl=62 time=2033 ms
64 bytes from 10.0.6.10: icmp_seq=9 ttl=62 time=2014 ms
64 bytes from 10.0.6.10: icmp_seq=10 ttl=62 time=1998 ms

--- 10.0.6.10 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9077ms
rtt min/avg/max/mdev = 1977.658/2006.899/2033.716/17.635 ms, pipe 3
```

Με τα προηγούμενα && FTP n8-n15 && FTP n15-n8(Κατάσταση 2):

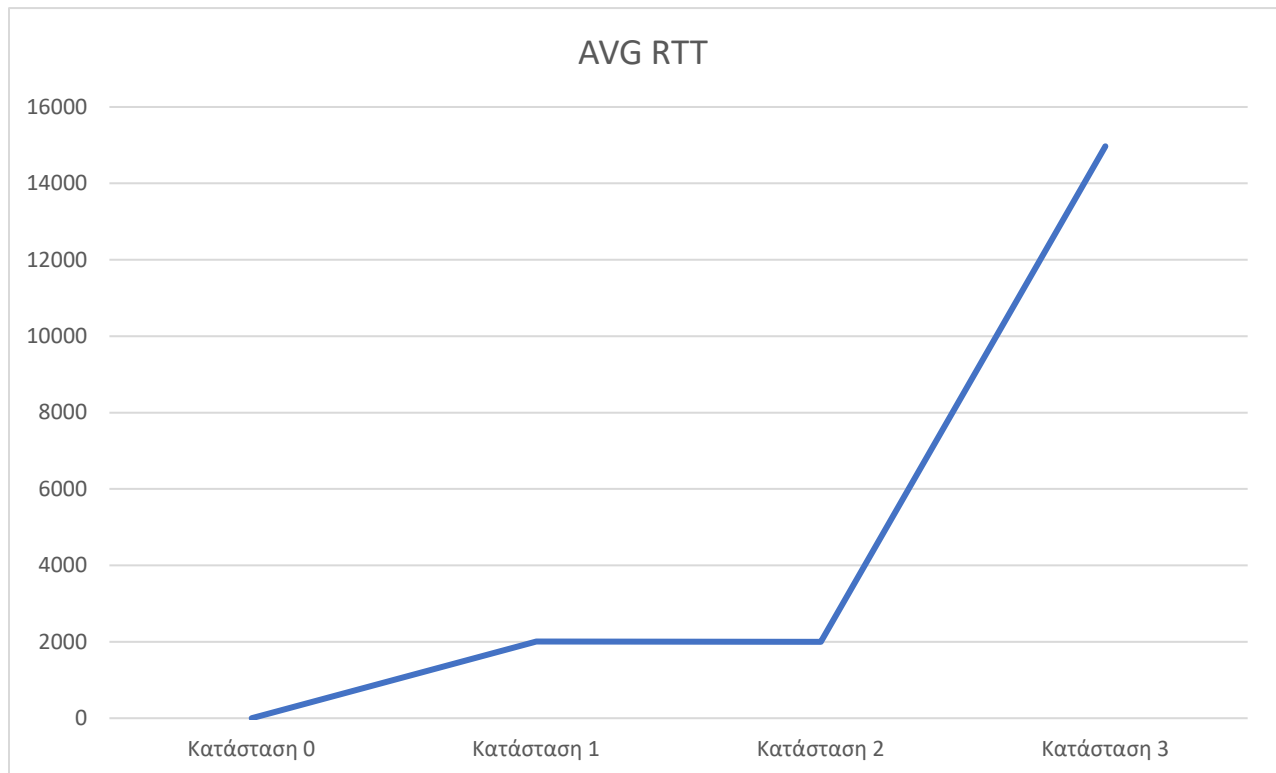
```
root@n7:/tmp/pycore.41731/n7.conf# ping -c 10 10.0.6.10
PING 10.0.6.10 (10.0.6.10) 56(84) bytes of data.
64 bytes from 10.0.6.10: icmp_seq=1 ttl=62 time=1980 ms
64 bytes from 10.0.6.10: icmp_seq=2 ttl=62 time=1997 ms
64 bytes from 10.0.6.10: icmp_seq=3 ttl=62 time=1991 ms
64 bytes from 10.0.6.10: icmp_seq=4 ttl=62 time=2022 ms
64 bytes from 10.0.6.10: icmp_seq=5 ttl=62 time=2012 ms
64 bytes from 10.0.6.10: icmp_seq=6 ttl=62 time=1995 ms
64 bytes from 10.0.6.10: icmp_seq=7 ttl=62 time=1991 ms
64 bytes from 10.0.6.10: icmp_seq=8 ttl=62 time=2027 ms
64 bytes from 10.0.6.10: icmp_seq=9 ttl=62 time=2017 ms
64 bytes from 10.0.6.10: icmp_seq=10 ttl=62 time=1994 ms

--- 10.0.6.10 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9088ms
rtt min/avg/max/mdev = 1980.815/2003.143/2027.845/15.070 ms, pipe 3
```


Με όλα τα traffic flows(Κατάσταση 3):

```
root@n7:/tmp/pycore.41731/n7.conf# ping -c 10 10.0.6.10
PING 10.0.6.10 (10.0.6.10) 56(84) bytes of data.
64 bytes from 10.0.6.10: icmp_seq=1 ttl=62 time=12692 ms
64 bytes from 10.0.6.10: icmp_seq=2 ttl=62 time=15836 ms
64 bytes from 10.0.6.10: icmp_seq=3 ttl=62 time=16053 ms
64 bytes from 10.0.6.10: icmp_seq=4 ttl=62 time=15296 ms

--- 10.0.6.10 ping statistics ---
10 packets transmitted, 4 received, 60% packet loss, time 9215ms
rtt min/avg/max/mdev = 12692.351/14969.789/16053.912/1343.515 ms, pipe 10
```



b) Για την μέτρηση του throughput θα χρησιμοποιήσουμε το tool iperf στους κόμβους n6-n15

Χωρίς traffic flow(Κατάσταση 0):

```
root@n15:/tmp/pycore.41731/n15.conf# iperf3 -c 10.0.15.10 -p 7575
Connecting to host 10.0.15.10, port 7575
[ 4] local 10.0.6.10 port 45666 connected to 10.0.15.10 port 7575
[ ID] Interval           Transfer     Bandwidth       Retr   Cwnd
[ 4]  0.00-1.00   sec    107 KBytes    880 Kbits/sec    0    33.9 KBytes
[ 4]  1.00-2.00   sec    140 KBytes   1.15 Mbits/sec    0    35.4 KBytes
[ 4]  2.00-3.00   sec    29.7 KBytes   244 Kbits/sec    0    36.8 KBytes
[ 4]  3.00-4.00   sec    65.0 KBytes   532 Kbits/sec    0    43.8 KBytes
[ 4]  4.00-5.00   sec    127 KBytes   1.04 Mbits/sec    0    58.0 KBytes
[ 4]  5.00-6.00   sec    127 KBytes   1.04 Mbits/sec    0    72.1 KBytes
[ 4]  6.00-7.00   sec    63.6 KBytes   520 Kbits/sec    0    87.7 KBytes
[ 4]  7.00-8.00   sec     0.00 Bytes    0.00 bits/sec    7    76.4 KBytes
[ 4]  8.00-9.00   sec     0.00 Bytes    0.00 bits/sec   11    55.1 KBytes
[ 4]  9.00-10.00  sec    63.6 KBytes   520 Kbits/sec    4    63.6 KBytes
- - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[ 4]  0.00-10.00  sec    724 KBytes   593 Kbits/sec    22
[ 4]  0.00-10.00  sec    342 KBytes   280 Kbits/sec
sender
receiver
iperf Done.
```

Με http/s for vid streaming n6-n15 && http/s for vid streaming n6-n15 && http/s for vid streaming n15-n6 (Video streaming emulation)(Κατάσταση 1):

```
root@n15:/tmp/pycore.41731/n15.conf# iperf3 -c 10.0.15.10 -p 7575
Connecting to host 10.0.15.10, port 7575
[ 4] local 10.0.6.10 port 45676 connected to 10.0.15.10 port 7575
[ ID] Interval           Transfer     Bandwidth       Retr   Cwnd
[ 4]  0.00-1.97   sec    76.4 KBytes   317 Kbits/sec    1    14.1 KBytes
[ 4]  1.97-2.00   sec     0.00 Bytes    0.00 bits/sec    0    14.1 KBytes
[ 4]  2.00-3.00   sec     0.00 Bytes    0.00 bits/sec    0    14.1 KBytes
[ 4]  3.00-4.01   sec    31.1 KBytes   254 Kbits/sec    0    17.0 KBytes
[ 4]  4.01-5.00   sec     0.00 Bytes    0.00 bits/sec    0    28.3 KBytes
[ 4]  5.00-6.00   sec     0.00 Bytes    0.00 bits/sec    0    31.1 KBytes
[ 4]  6.00-7.00   sec    129 KBytes   1.06 Mbits/sec    0    38.2 KBytes
[ 4]  7.00-8.00   sec     2.83 KBytes  23.2 Kbits/sec    0    41.0 KBytes
[ 4]  8.00-9.00   sec    21.2 KBytes   174 Kbits/sec    0    50.9 KBytes
[ 4]  9.00-10.00  sec    132 KBytes   1.08 Mbits/sec    0    59.4 KBytes
- - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[ 4]  0.00-10.00  sec    392 KBytes   321 Kbits/sec    1
[ 4]  0.00-10.00  sec    141 KBytes   116 Kbits/sec
sender
receiver
iperf Done.
```

Με τα προηγούμενα && FTP n8-n15 && FTP n15-n8(Κατάσταση 2):

```
root@n15:/tmp/pycore.41731/n15.conf# iperf3 -c 10.0.15.10 -p 7575
Connecting to host 10.0.15.10, port 7575
[  4] local 10.0.6.10 port 45680 connected to 10.0.15.10 port 7575
[ ID] Interval           Transfer     Bandwidth       Retr   Cwnd
[  4]  0.00-1.98   sec    76.4 KBytes    317 Kbits/sec     1   14.1 KBytes
[  4]  1.98-2.00   sec     0.00 Bytes     0.00 bits/sec     0   14.1 KBytes
[  4]  2.00-3.00   sec     0.00 Bytes     0.00 bits/sec     0   14.1 KBytes
[  4]  3.00-4.00   sec    31.1 KBytes    255 Kbits/sec     0   18.4 KBytes
[  4]  4.00-5.00   sec     0.00 Bytes     0.00 bits/sec     0   28.3 KBytes
[  4]  5.00-6.00   sec     0.00 Bytes     0.00 bits/sec     0   29.7 KBytes
[  4]  6.00-7.00   sec    18.4 KBytes    151 Kbits/sec     0   53.7 KBytes
[  4]  7.00-8.00   sec    17.0 KBytes    139 Kbits/sec     0   59.4 KBytes
[  4]  8.00-9.00   sec    407 KBytes    3.34 Mbits/sec     0   79.2 KBytes
[  4]  9.00-10.00  sec    69.3 KBytes    567 Kbits/sec     0   103 KBytes
-----
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]  0.00-10.00  sec    619 KBytes    507 Kbits/sec     1
[  4]  0.00-10.00  sec    160 KBytes    131 Kbits/sec
iperf Done.
```

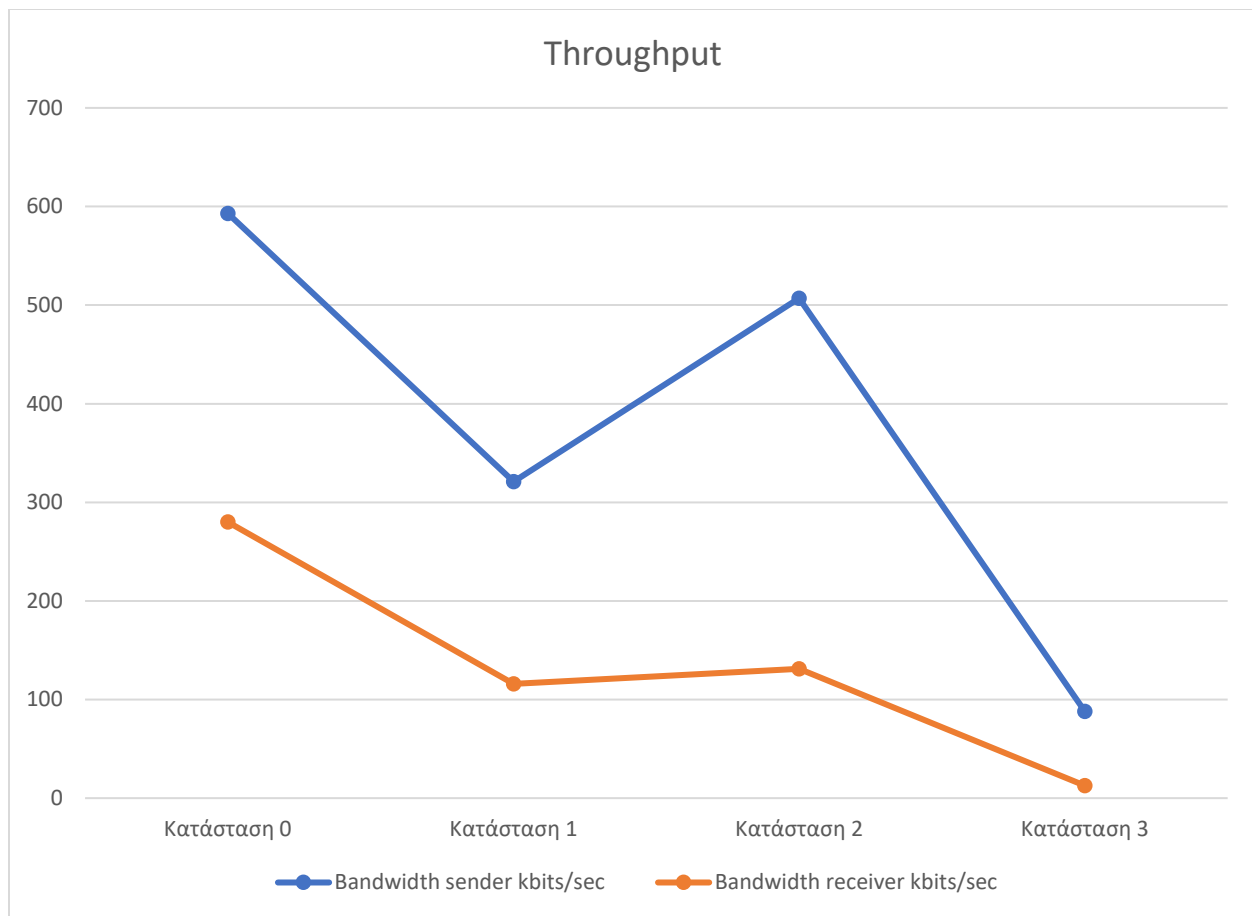
Με όλα τα traffic flows(Κατάσταση 3):

```

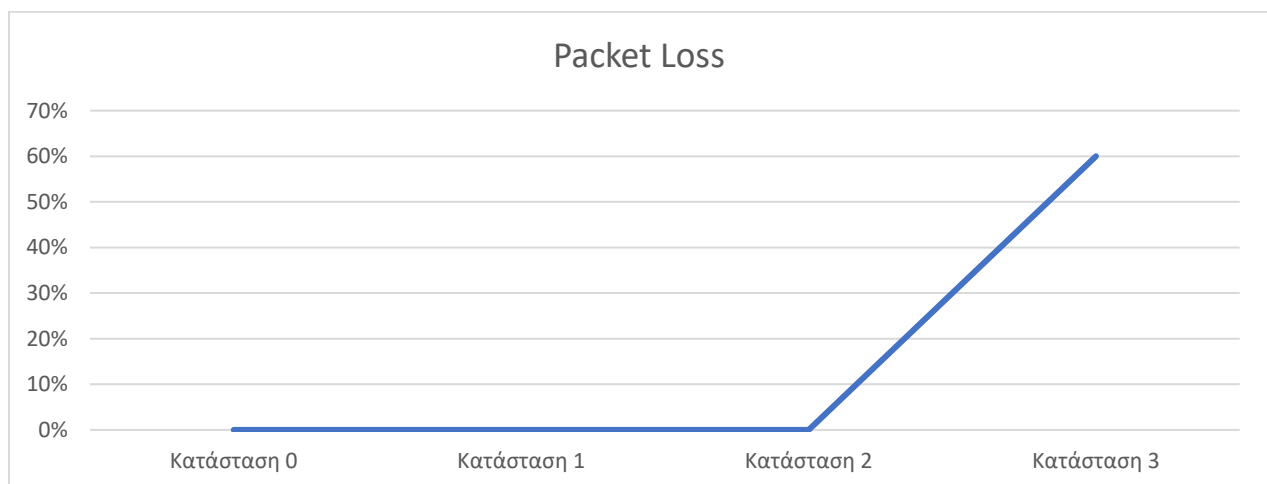
root@n15:/tmp/pycore.41731/n15.conf# iperf3 -c 10.0.15.10 -p 7575
Connecting to host 10.0.15.10, port 7575
[  4] local 10.0.6.10 port 45684 connected to 10.0.15.10 port 7575
[ ID] Interval           Transfer     Bandwidth       Retr   Cwnd
[  4]  0.00-1.00        sec   76.4 KBytes    624 Kbits/sec     3   14.1 KBytes
[  4]  1.00-2.00        sec    0.00 Bytes    0.00 bits/sec     0   14.1 KBytes
[  4]  2.00-3.00        sec   31.1 KBytes   254 Kbits/sec     0   28.3 KBytes
[  4]  3.00-4.00        sec    0.00 Bytes    0.00 bits/sec     0   28.3 KBytes
[  4]  4.00-5.00        sec    0.00 Bytes    0.00 bits/sec     0   28.3 KBytes
[  4]  5.00-6.00        sec    0.00 Bytes    0.00 bits/sec     0   28.3 KBytes
[  4]  6.00-7.00        sec    0.00 Bytes    0.00 bits/sec     0   29.7 KBytes
[  4]  7.00-8.00        sec    0.00 Bytes    0.00 bits/sec     1   12.7 KBytes
[  4]  8.00-9.00        sec    0.00 Bytes    0.00 bits/sec     0   12.7 KBytes
[  4]  9.00-10.00       sec    0.00 Bytes    0.00 bits/sec     0   12.7 KBytes
- - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]  0.00-10.00       sec   107 KBytes   88.0 Kbits/sec     4
[  4]  0.00-10.00       sec   15.6 KBytes  12.7 Kbits/sec
sender
receiver

iperf Done.

```



c) Για το packet loss θα χρησιμοποιήσουμε το ring όπως και στο ερώτημα α, όπου φαίνεται και στα screenshots του. Εξαιτίας της παραμέτρου -c 10 δεν προλαβαίνει να υπάρξει μεγάλο packet loss



4) Θα τρέξουμε το wireshark στον κόμβο n15 και θα κάνουμε capture κίνηση με την εντολή ως root

- `sudo tshark -i eth0 -w capture_n15-out_katN.pcap`

Όπου eth0 το όνομα του network interface και -w το όνομα του εξαγόμενου αρχείου και N η κάθε διαφορετική κατάσταση

5)

1^η Καταγραφή:

1	0.000000000	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
2	2.001641799	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
3	4.002100446	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
4	6.002896249	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
6	8.007547036	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
7	10.009037841	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
8	12.010438177	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
10	14.013891658	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
11	16.014173758	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet
13	18.016261211	10.0.6.1	224.0.0.5	OSPF	78 Hello Packet

Παρατηρούμε ότι δεν υπάρχει καθόλου κίνηση το οποίο όντως συμβαίνει διότι βρισκόμαστε στην κατάσταση 0 (κανένα traffic flow)

2^η Καταγραφή:

1	0.000000000	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
2	0.042508919	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
3	0.083428754	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
4	0.123889827	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
5	0.165231384	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
6	0.209261449	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
7	0.250389101	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
8	0.289227105	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
9	0.332670188	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
10	0.372506552	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
11	0.414771659	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
12	0.457174769	10.0.15.10	10.0.6.10	UDP	1322 1754 → 5032 Len=1280
705	27.938464031	10.0.15.10	10.0.6.10	TCP	1514 443 → 420 [ACK] Seq=17606 Ack=1 Win=502 Len=1448 TSval
706	27.982436297	10.0.6.10	10.0.15.10	TCP	66 420 → 443 [ACK] Seq=1 Ack=19054 Win=501 Len=0 TSval
757	30.010470403	10.0.15.10	10.0.6.10	TCP	1514 [TCP Previous segment not captured] 443 → 420 [ACK]
758	30.010489338	10.0.6.10	10.0.15.10	TCP	78 [TCP Dup ACK 706#1] 420 → 443 [ACK] Seq=1 Ack=19054
808	32.043082214	10.0.15.10	10.0.6.10	TCP	1514 [TCP Retransmission] 443 → 420 [ACK] Seq=19054 Ack=
809	32.043167063	10.0.6.10	10.0.15.10	TCP	66 420 → 443 [ACK] Seq=1 Ack=21950 Win=496 Len=0 TSval
859	34.075562697	10.0.15.10	10.0.6.10	TCP	1514 443 → 420 [ACK] Seq=21950 Ack=1 Win=502 Len=1448 TS
861	34.120766854	10.0.6.10	10.0.15.10	TCP	66 420 → 443 [ACK] Seq=1 Ack=23398 Win=501 Len=0 TSval
912	36.151457016	10.0.15.10	10.0.6.10	TCP	1514 [TCP Previous segment not captured] 443 → 420 [ACK]
913	36.151473828	10.0.6.10	10.0.15.10	TCP	78 [TCP Dup ACK 861#1] 420 → 443 [ACK] Seq=1 Ack=23398
963	38.189102651	10.0.15.10	10.0.6.10	TCP	1514 [TCP Retransmission] 443 → 420 [ACK] Seq=23398 Ack=
964	38.189201046	10.0.6.10	10.0.15.10	TCP	66 420 → 443 [ACK] Seq=1 Ack=26294 Win=496 Len=0 TSval
1014	40.217673758	10.0.15.10	10.0.6.10	TCP	1514 443 → 420 [ACK] Seq=26294 Ack=1 Win=502 Len=1448 TS
1015	40.259226678	10.0.6.10	10.0.15.10	TCP	66 420 → 443 [ACK] Seq=1 Ack=27742 Win=501 Len=0 TSval

Παρατηρούμε ότι υπάρχει το emulation του πρωτοκόλλου QUIC που χρησιμοποιεί το UDP και έχει ως source IP την IP του κόμβου n6 το οποίο μπορούμε να το επαληθεύσουμε και από τα screenshots στο ερώτημα 2(QUIC (video stream) emulation n6-n15).

Επιπλέον παρατηρούμε τα TCP πακέτα που είναι του πρωτοκόλλου HTTP που προέρχονται από τον ίδιο κόμβο τον n6, και επαληθεύεται από τα screenshots του ερωτήματος 2(http/s vid streaming n6-n15).

3^η Καταγραφή:

Υπάρχει όλη η προηγούμενη κίνηση δηλαδή η προσομοίωση HTTP και QUIC που χρησιμοποιούνται για το video streaming, αλλά παρατηρούμε ότι υπάρχει πλέον και η κίνηση από την προσομοίωση μεταφοράς αρχείων (FTP) και έχει ως source IP την IP του κόμβου n8 όπως φαίνεται και στο screenshot του ερωτήματος 2(FTP n8-n15).

96	3.591112290	10.0.13.10	10.0.6.10	TCP	66	21 → 1420 [ACK] Seq=14481 Ack=501 Win=501 Len=0 TSval=4101469401 TSecr=3274974704
113	4.216684332	10.0.13.10	10.0.6.10	TCP	1514	[TCP Retransmission] 21 → 1420 [ACK] Seq=1449 Ack=501 Win=501 Len=1448 TSval=4101469983 TSecr=3274975286
151	5.631585019	10.0.13.10	10.0.6.10	TCP	66	21 → 1420 [ACK] Seq=14481 Ack=751 Win=501 Len=0 TSval=4101471398 TSecr=3274976701
168	6.257410535	10.0.13.10	10.0.6.10	TCP	1514	[TCP Retransmission] 21 → 1420 [ACK] Seq=4345 Ack=751 Win=501 Len=1448 TSval=4101472024 TSecr=3274977327
205	7.674131457	10.0.13.10	10.0.6.10	TCP	66	21 → 1420 [ACK] Seq=14481 Ack=1001 Win=501 Len=0 TSval=4101473439 TSecr=3274978742
222	8.301258510	10.0.13.10	10.0.6.10	TCP	1514	[TCP Retransmission] 21 → 1420 [ACK] Seq=5793 Ack=1001 Win=501 Len=1448 TSval=4101474065 TSecr=3274979368
258	9.676631288	10.0.13.10	10.0.6.10	TCP	66	21 → 1420 [ACK] Seq=14481 Ack=1251 Win=501 Len=0 TSval=4101475482 TSecr=3274980784
276	10.351034741	10.0.13.10	10.0.6.10	TCP	1514	[TCP Retransmission] 21 → 1420 [ACK] Seq=8689 Ack=1251 Win=501 Len=1448 TSval=4101476108 TSecr=3274981411
312	11.683341270	10.0.13.10	10.0.6.10	TCP	66	21 → 1420 [ACK] Seq=14481 Ack=1501 Win=501 Len=0 TSval=4101477484 TSecr=3274982787
331	12.393152270	10.0.13.10	10.0.6.10	TCP	1514	[TCP Retransmission] 21 → 1420 [ACK] Seq=7241 Ack=1501 Win=501 Len=1448 TSval=4101478158 TSecr=3274983461
366	13.690026773	10.0.13.10	10.0.6.10	TCP	66	21 → 1420 [ACK] Seq=14481 Ack=1751 Win=501 Len=0 TSval=4101479491 TSecr=3274984794
386	14.436159572	10.0.13.10	10.0.6.10	TCP	1514	[TCP Retransmission] 21 → 1420 [ACK] Seq=10137 Ack=1751 Win=501 Len=1448 TSval=4101480200 TSecr=3274985503
420	15.683439848	10.0.13.10	10.0.6.10	TCP	66	21 → 1420 [ACK] Seq=14481 Ack=2001 Win=501 Len=0 TSval=4101481498 TSecr=3274986800