

ΕΡΓΑΣΙΑ ΔΙΑΧΕΤΡΙΣΗ ΔΙΚΤΥΩΝ

ΟΜΑΔΑ:

Ον/μο: KOMINH ΣΥΜΕΛΛΑ ΦΩΤΕΙΝΗ
ΑΜ: 1115201400072

Ον/μο: ΜΑΣΤΟΡΑΚΗ ΑΙΚΑΤΕΡΙΝΑ
ΑΜ: 1115201400100

1ο ΠΕΙΡΑΜΑ

Σε κάθε φάση προσθέσαμε τις κατάλληλες εντολές για τα flows του switch και τα routing tables. Στο τέλος της κάθε φάσης διαγράφουμε τις ρυθμίσεις της προηγούμενης.

1η φάση

```
os.system('ovs-ofctl mod-flows switch in_port=1,actions=output:4')
os.system('ovs-ofctl mod-flows switch in_port=4,actions=output:1')
os.system('ovs-ofctl mod-flows switch in_port=2,actions=drop')
os.system('ovs-ofctl mod-flows switch in_port=3,actions=drop')
os.system('ovs-ofctl del-flows eNodeB1')
os.system('ovs-ofctl del-flows eNodeB2')
os.system('ovs-ofctl del-flows rsu1')
```

```
car[0].cmd('ip route add 200.0.10.2 via 200.0.10.50')
client.cmd('ip route add 200.0.10.100 via 200.0.10.150')
```

2η φάση

```
os.system('ovs-ofctl mod-flows switch in_port=1,actions=drop')
os.system('ovs-ofctl mod-flows switch in_port=2,actions=output:4')
os.system('ovs-ofctl mod-flows switch in_port=4,actions=output:2,3')
os.system('ovs-ofctl mod-flows switch in_port=3,actions=output:4')
os.system('ovs-ofctl del-flows eNodeB1')
os.system('ovs-ofctl del-flows eNodeB2')
os.system('ovs-ofctl del-flows rsu1')
```

```
car[0].cmd('ip route del 200.0.10.2 via 200.0.10.50')
client.cmd('ip route del 200.0.10.100 via 200.0.10.150')
```

3η φάση

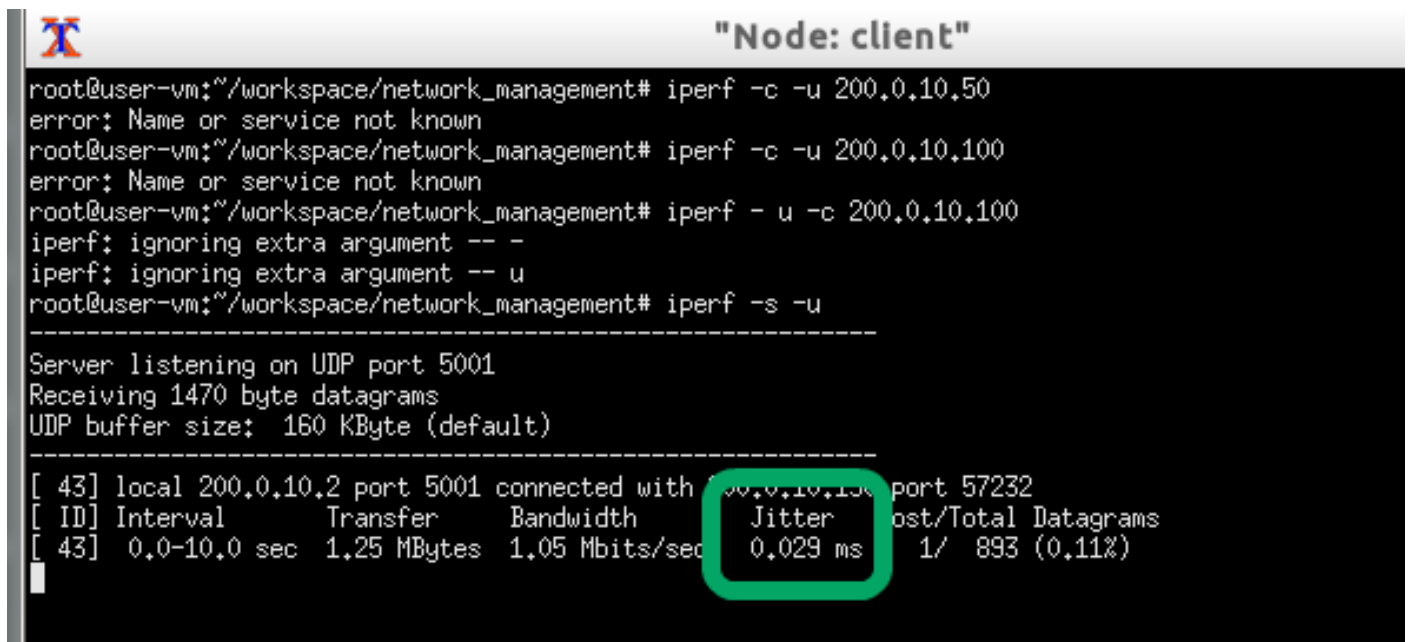
```
os.system('ovs-ofctl mod-flows switch in_port=1,actions=drop')
os.system('ovs-ofctl mod-flows switch in_port=3,actions=drop')
os.system('ovs-ofctl mod-flows switch in_port=2,actions=output:4')
os.system('ovs-ofctl mod-flows switch in_port=4,actions=output:2')
os.system('ovs-ofctl del-flows eNodeB1')
os.system('ovs-ofctl del-flows eNodeB2')
os.system('ovs-ofctl del-flows rsu1')
```

Σε κάθε φάση παίρνουμε μετρήσεις για:

- ✓ Throughput
- ✓ Jitter
- ✓ Packet loss
- ✓ Latency

Οι μετρήσεις των jitter και packet loss, σε κάθε φάση, φαίνεται στα παρακάτω screenshots:

1η φάση



```
root@user-vm:~/workspace/network_management# iperf -c -u 200.0.10.50
error: Name or service not known
root@user-vm:~/workspace/network_management# iperf -c -u 200.0.10.100
error: Name or service not known
root@user-vm:~/workspace/network_management# iperf -u -c 200.0.10.100
iperf: ignoring extra argument -- -
iperf: ignoring extra argument -- u
root@user-vm:~/workspace/network_management# iperf -s -u
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)
-----
[ 43] local 200.0.10.2 port 5001 connected with 200.0.10.100 port 57232
[ ID] Interval      Transfer    Bandwidth      Jitter          Post/Total Datagrams
[ 43] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  0.029 ms        1/ 893 (0.11%)
```

```
"Node: car0"
root@user-vm:~/workspace/network_management# iperf -c -u 200.0.10.2
error: Name or service not known
root@user-vm:~/workspace/network_management# iperf -c -u
error: Name or service not known
root@user-vm:~/workspace/network_management# iperf -s -u
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)
-----
[[A^Croot@user-vm:~/workspace/network_management# iperf -c 10.0.0.4 -u
-----
Client connecting to 10.0.0.4, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size: 160 KByte (default)
-----
[ 43] local 200.0.10.100 port 37685 connected with 10.0.0.4 port 5001
^C[ ID] Interval      Transfer      Bandwidth
[ 43] 0.0- 1.9 sec  28.0 GBytes  129 Gbits/sec
[ 43] Sent 166 datagrams
read failed: Connection refused
[ 43] WARNING: did not receive ack of last datagram after 2 tries.
root@user-vm:~/workspace/network_management# iperf -c 200.0.10.50
connect failed: Connection refused
root@user-vm:~/workspace/network_management# iperf -c 200.0.10.150
connect failed: No route to host
root@user-vm:~/workspace/network_management# iperf -s -u
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)
-----
[ 43] local 200.0.10.100 port 5001 connected with 192.168.1.1 port 59065
[ ID] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[ 43] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  0.053 ms    0/ 893 (0%)
^C
```

```
"Node: car0"
root@user-vm:~/workspace/network_management# ping 10.0.0.4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=63 time=9.17 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=63 time=7.02 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=63 time=7.06 ms
64 bytes from 10.0.0.4: icmp_seq=4 ttl=63 time=6.95 ms
64 bytes from 10.0.0.4: icmp_seq=5 ttl=63 time=7.15 ms
^C
--- 10.0.0.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 6.955/7.473/9.179/0.866 ms
root@user-vm:~/workspace/network_management#
```

```
"Node: car3"
root@user-vm:~/workspace/network_management# ping 200.0.10.2
PING 200.0.10.2 (200.0.10.2) 56(84) bytes of data.
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=6.67 ms
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=5.87 ms
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=5.40 ms
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=5.22 ms
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=5.12 ms
^C
--- 200.0.10.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 5.122/5.666/6.670/0.579 ms
root@user-vm:~/workspace/network_management#
```

2η φάση

```
"Node: client"
root@user-vm:~/workspace/network_management# iperf -s -u
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)

[ 43] local 200.0.10.2 port 5001 connected with 200.0.10.100 port 58463
[ ID] Interval      Transfer    Bandwidth      Jitter    Lost/Total Datagrams
[ 43] 0.0-10.0 sec  2.50 MBytes  2.10 Mbits/sec  7.391 ms   0/ 893 (0%)
[ 43] 0.0-10.0 sec  892 datagrams received out of order
read failed: Connection refused

mininet@user-vm: ~/workspace/network_management
[0x41f000] xcb_glx vout display debug: display is visible
[0x41f000] xcb_glx vout display debug: display is visible
[0x41f000] xcb_glx vout display debug: display is visible
PING 200.0.10.2 (200.0.10.2) 56(84) bytes of data.
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=5.78 ms
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=6.31 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=6.90 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=6.91 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=4.30 ms
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=4.32 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=5.13 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=5.14 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=3.97 ms
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=3.97 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=5.21 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=5.21 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=3.97 ms
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=3.97 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=5.19 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=5.19 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=4.00 ms
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=4.00 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=5.28 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=5.28 ms (DUP!)
^C
[0x936d1a8] --- 200.0.10.2 ping statistics ---
5 packets transmitted, 5 received, +15 duplicates, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 3.972/5.006/6.913/0.925 ms
main libvlc debug: removing all interfaces
[0x936d1a8] main libvlc debug: exiting
[0x9412f98] main interface debug: removing module "qt4"
[0x937c4a0] mininet-wifi> []
```

Εδώ παρατηρούμε ότι εμφανίζεται η ένδειξη DUP δίπλα από κάθε πακέτο, γιατί τα πακέτα μεταδίδονται multicasting και ο client τα λαμβάνει διπλά.

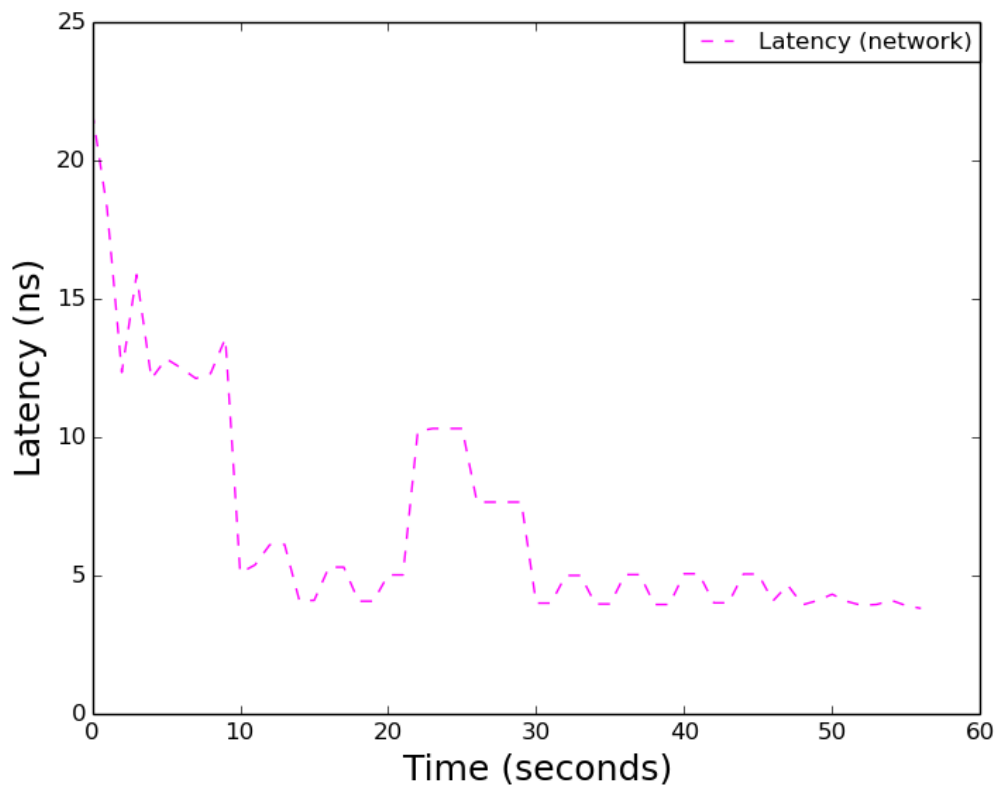
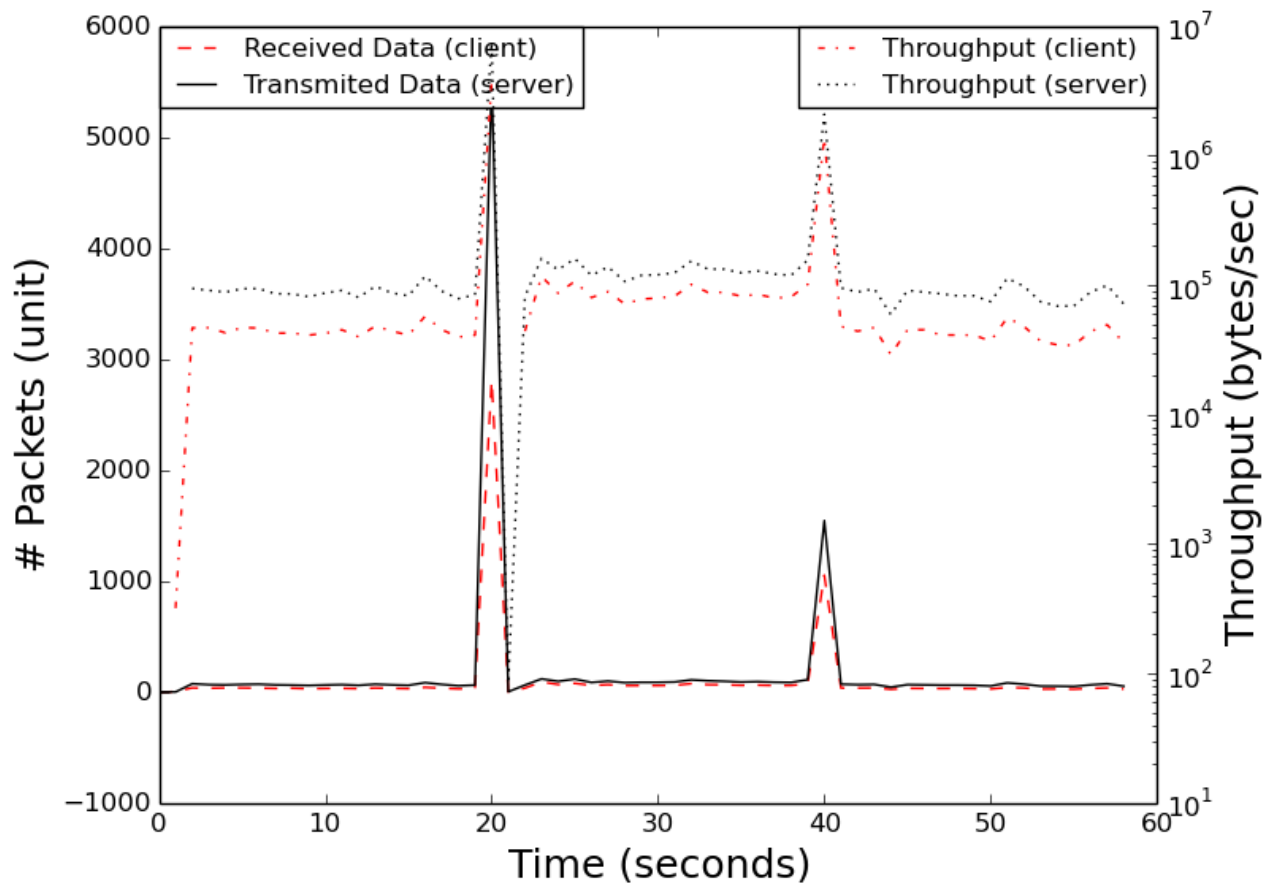
3η φάση

```
"Node: client"
root@user-vm:~/workspace/network_management# iperf -s -u
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)

[ 43] local 200.0.10.2 port 5001 connected with 200.0.10.100 port 44948
[ ID] Interval      Transfer    Bandwidth      Jitter    Lost/Total Datagrams
[ 43] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  0.227 ms   1/ 893 (0.11%)

mininet@user-vm: ~/workspace/network_management
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=5.29 ms (DUP!)
^C
[0x936d1a8] --- 200.0.10.2 ping statistics ---
5 packets transmitted, 5 received, +15 duplicates, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 3.972/5.006/6.913/0.925 ms
main libvlc debug: removing all interfaces
[0x936d1a8] main libvlc debug: exiting
[0x9412f98] main interface debug: removing module "qt4"
[0x937c4a0] mininet-wifi> exit
Moving nodes
Applying third phase
*** Starting CLI:
mininet-wifi> xterm car0 client
mininet-wifi> car0 ping client
PING 200.0.10.2 (200.0.10.2) 56(84) bytes of data.
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=5.94 ms
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=3.95 ms
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=3.83 ms
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=3.84 ms
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=3.85 ms
^C
--- 200.0.10.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 3.836/4.200/6.913/0.925 ms
mininet-wifi> []
```

Οι μετρήσεις για το throughput και το latency φαίνονται στα παρακάτω διαγράμματα, τα οποία εμφανίζονται μετά από την 3η φάση.



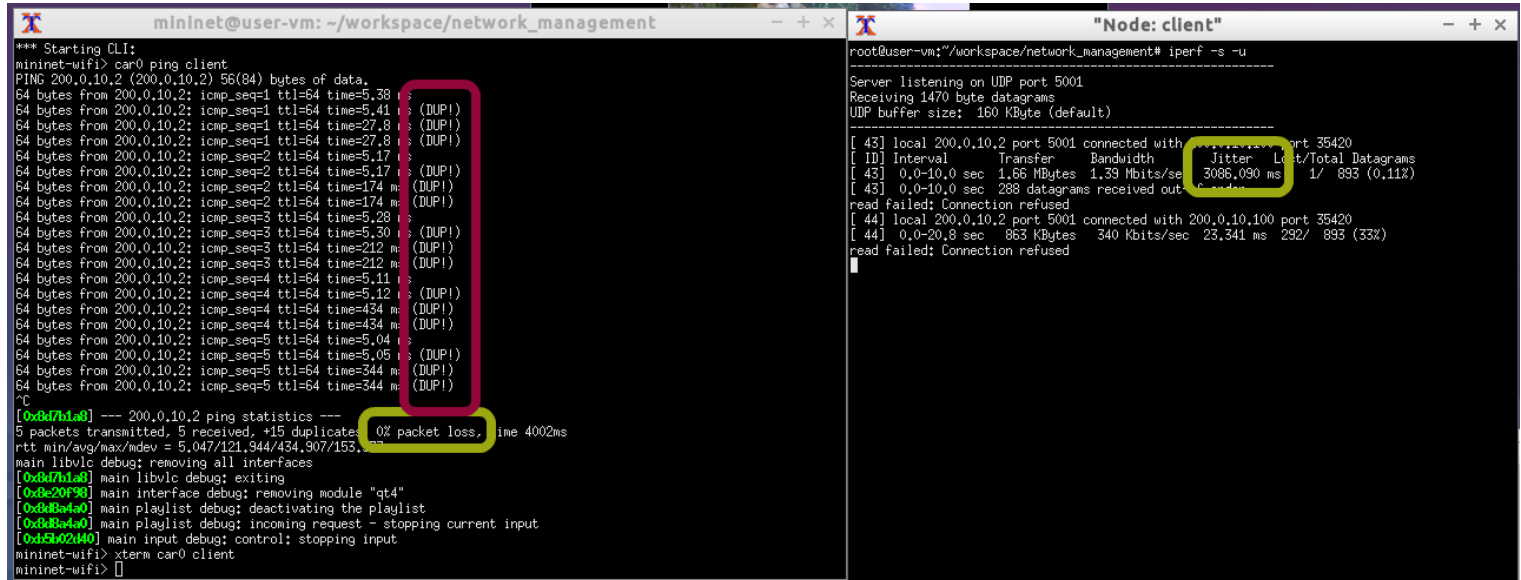
2ο ΠΕΙΡΑΜΑ – bicasting

Έχουμε ρυθμίσει τα flows του switch, με αντίστοιχο τρόπο με το 1ο πείραμα για κάθε φάση.

Οι μετρήσεις των jitter και packet loss για κάθε φάση είναι οι παρακάτω:

1η φάση

eNodeB1 – rsu1

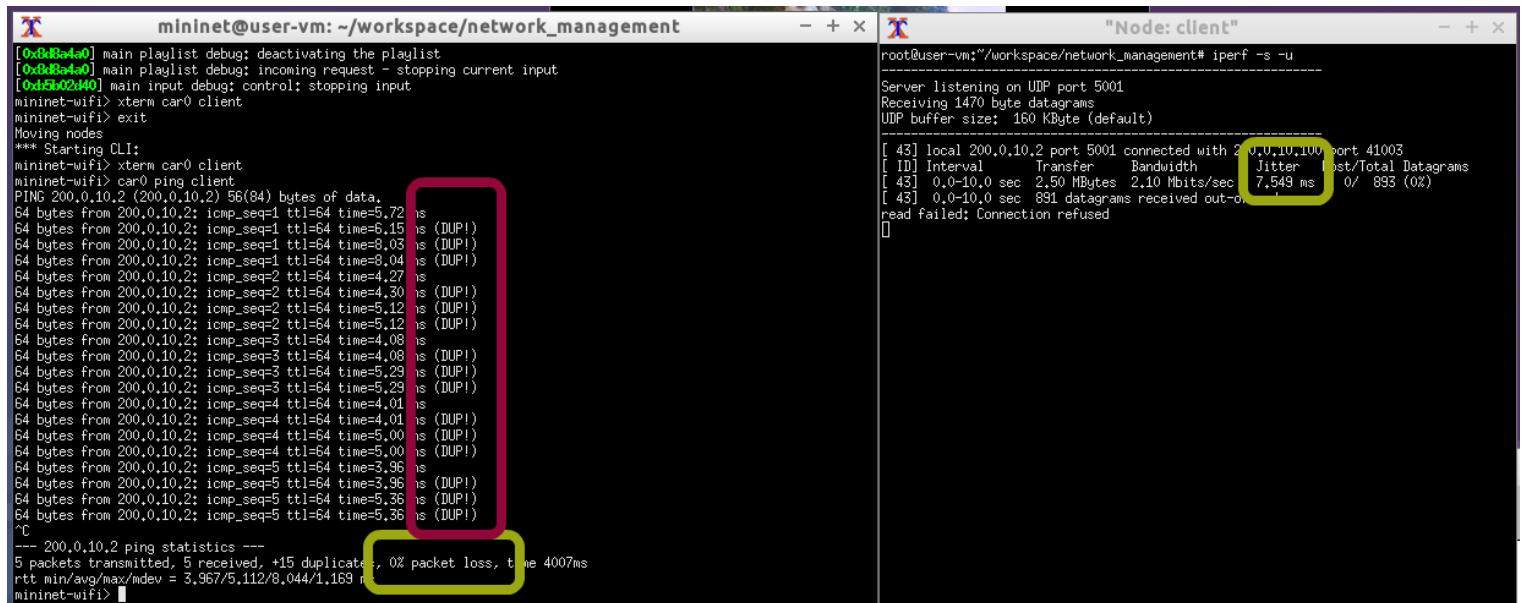


```
mininet@user-vm: ~/workspace/network_management
*** Starting CLI:
mininet-wifi> car0 ping client
PING 200.0.10.2 (200.0.10.2) 56(84) bytes of data:
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=5.38 ms
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=5.41 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=27.8 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=27.8 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=5.17 ms
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=5.17 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=174 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=174 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=5.28 ms
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=5.30 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=212 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=212 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=5.11 ms
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=5.12 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=434 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=434 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=5.04 ms
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=5.05 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=344 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=344 ms (DUP!)
^C
[0x8d7b1a0] --- 200.0.10.2 ping statistics ---
5 packets transmitted, 5 received, +15 duplicates, 0% packet loss, time 4002ms
rtt min/avg/max/mdev = 5.047/121.944/434.907/153.133 ms
main libvlc debug: removing all interfaces
[0x8d7b1a0] main libvlc debug: exiting
[0x8e20f98] main interface debug: removing module "qt4"
[0x8b3a4a0] main playlist debug: deactivating the playlist
[0x8b3a4a0] main playlist debug: incoming request - stopping current input
[0x8b3a4a0] main input debug: control: stopping input
mininet-wifi> xterm car0 client
mininet-wifi>

root@user-vm: ~/workspace/network_management# iperf -s -u
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)

[ 43] local 200.0.10.2 port 5001 connected with 200.0.10.100 port 35420
[ ID] Interval      Transfer    Bandwidth      Jitter    Lost/Total Datagrams
[ 43] 0.0-10.0 sec  1.66 MBytes  1.39 Mbits/sec  3086.090 ms  1/ 893 (0.11%)
[ 43] 0.0-10.0 sec  288 datagrams received out-order
read failed: Connection refused
[ 44] local 200.0.10.2 port 5001 connected with 200.0.10.100 port 35420
[ 44] 0.0-20.8 sec  863 KBytes   340 Kbits/sec  23.341 ms  292/ 893 (33%)
read failed: Connection refused
```

eNodeB2 – rsu1



```
mininet@user-vm: ~/workspace/network_management
[0x8b3a4a0] main playlist debug: deactivating the playlist
[0x8b3a4a0] main playlist debug: incoming request - stopping current input
[0x8b3a4a0] main input debug: control: stopping input
mininet-wifi> xterm car0 client
mininet-wifi> exit
Moving nodes
*** Starting CLI:
mininet-wifi> xterm car0 client
mininet-wifi> car0 ping client
PING 200.0.10.2 (200.0.10.2) 56(84) bytes of data:
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=5.72 ms
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=5.15 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=8.03 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=8.04 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=4.27 ms
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=4.30 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=5.12 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=5.12 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=4.08 ms
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=4.08 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=5.29 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=5.29 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=4.01 ms
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=4.01 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=5.00 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=5.00 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=3.96 ms
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=3.96 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=5.36 ms (DUP!)
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=5.36 ms (DUP!)
^C
--- 200.0.10.2 ping statistics ---
5 packets transmitted, 5 received, +15 duplicates, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 3.967/5.112/8.044/1.163 ms
mininet-wifi>

root@user-vm: ~/workspace/network_management# iperf -s -u
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)

[ 43] local 200.0.10.2 port 5001 connected with 200.0.10.100 port 41003
[ ID] Interval      Transfer    Bandwidth      Jitter    Lost/Total Datagrams
[ 43] 0.0-10.0 sec  2.50 MBytes  2.10 Mbits/sec  7.549 ms   0/ 893 (0%)
[ 43] 0.0-10.0 sec  891 datagrams received out-order
read failed: Connection refused
```

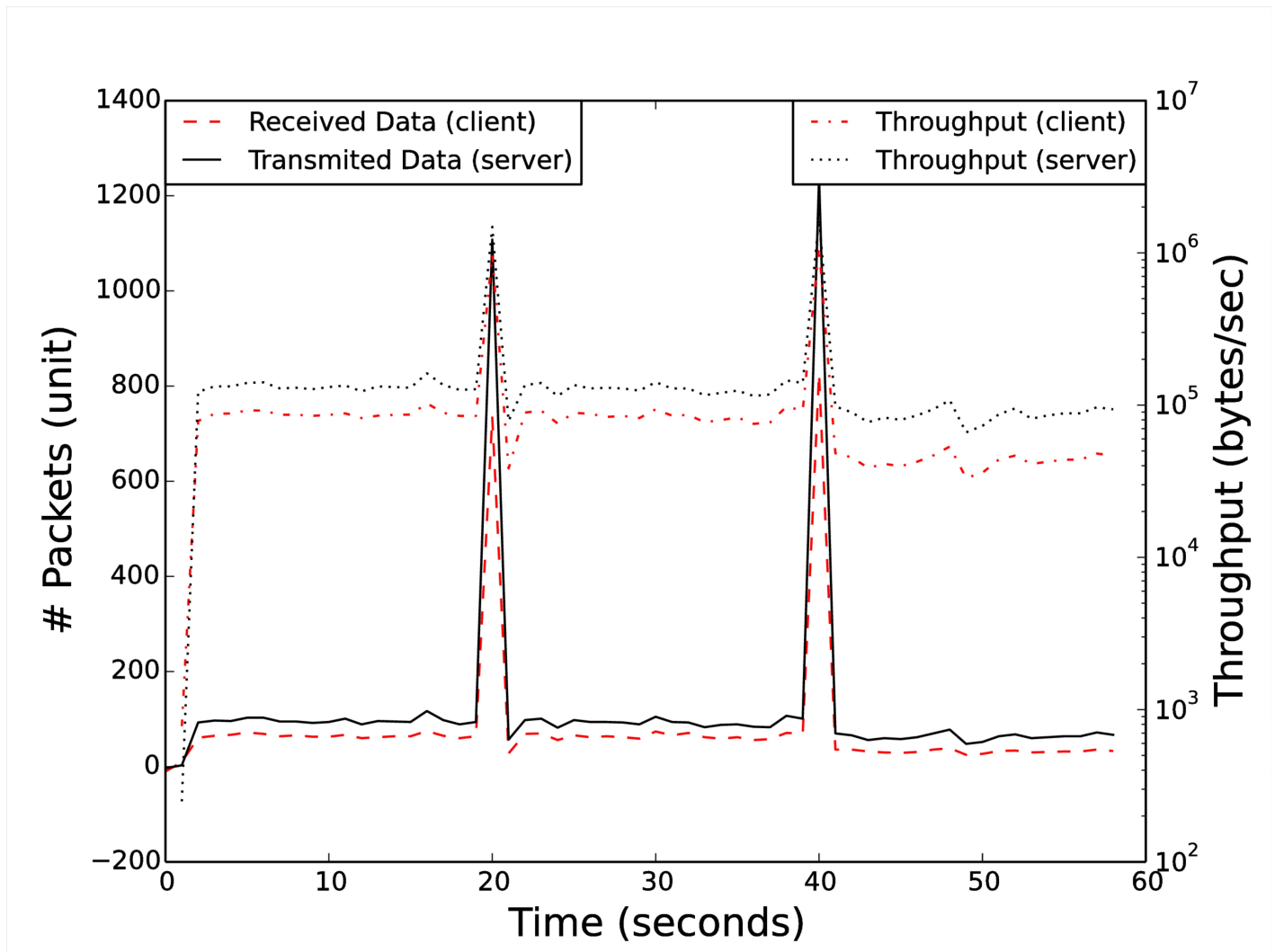
2η φάση

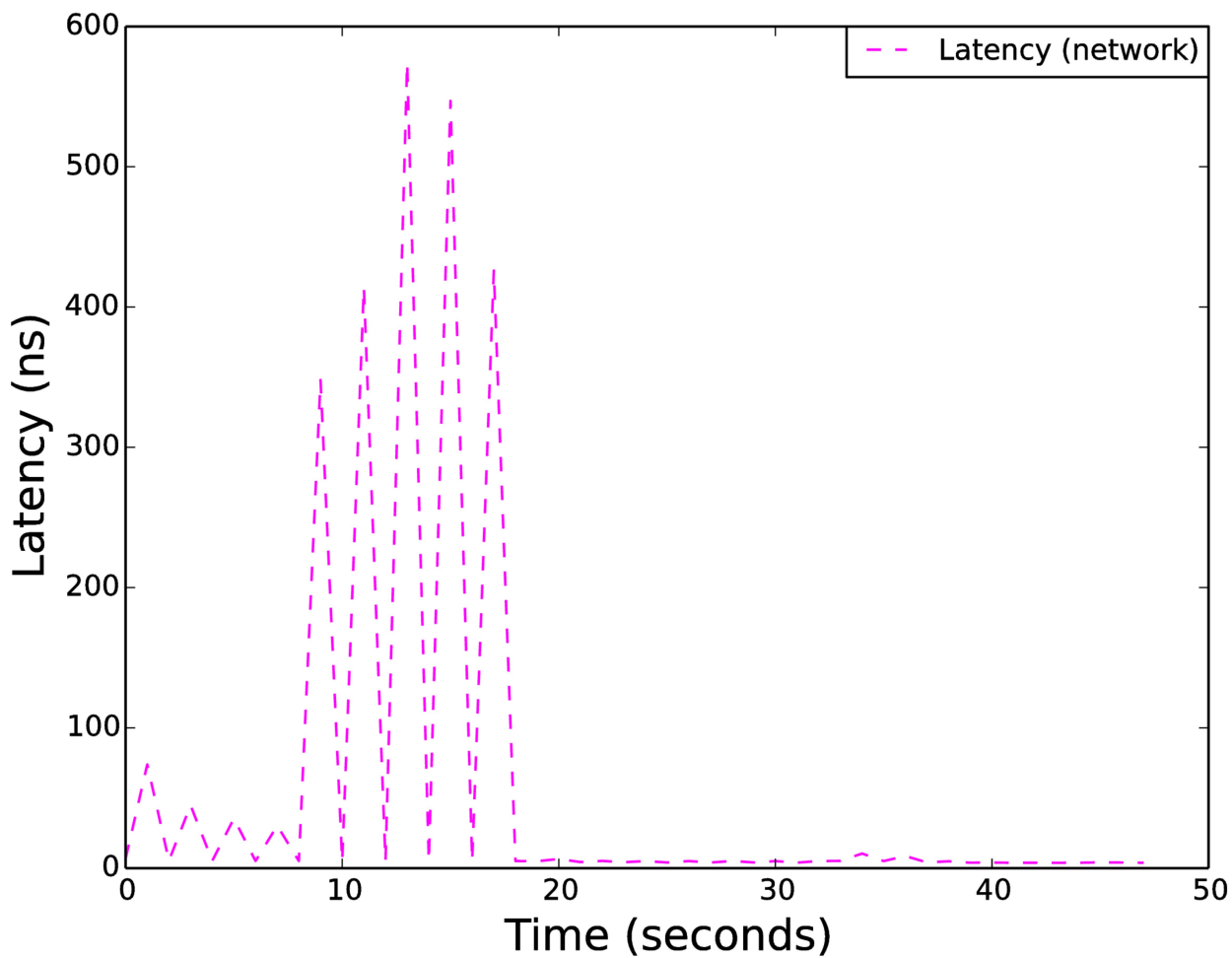
```
mininet@user-vm: ~/...etwork_management - + x
Applying second phase
*** Starting CLI:
mininet-wifi> xterm car0 client
mininet-wifi> car0 ping client
PING 200.0.10.2 (200.0.10.2) 56(84) bytes of data:
64 bytes from 200.0.10.2: icmp_seq=1 ttl=64 time=5.59 ms
64 bytes from 200.0.10.2: icmp_seq=2 ttl=64 time=3.89 ms
64 bytes from 200.0.10.2: icmp_seq=3 ttl=64 time=3.81 ms
64 bytes from 200.0.10.2: icmp_seq=4 ttl=64 time=3.80 ms
64 bytes from 200.0.10.2: icmp_seq=5 ttl=64 time=3.79 ms
64 bytes from 200.0.10.2: icmp_seq=6 ttl=64 time=4.01 ms
64 bytes from 200.0.10.2: icmp_seq=7 ttl=64 time=4.12 ms
^C
--- 200.0.10.2 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6007ms
rtt min/avg/max/mdev = 3.793/4.117/5.600/10.605 ms
mininet-wifi>

root@user-vm:~/workspace/network_management# iperf -s -u
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)

[ 43] local 200.0.10.2 port 5001 connected with 200.0.10.100 port 55692
[ 10] Interval      Transfer      Bandwidth      Jitter      Lost/Total Datagrams
[ 43] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  0.033 ms    1/ 893 (0.11%)
```

Ομοίως με το 1ο πείραμα, οι μετρήσεις του throughput και του latency εμφανίζονται με γραφική παράσταση μετά τη 2η φάση.





Σύγκριση αποτελεσμάτων

Από τα αποτελέσματα των παραπάνω πειραμάτων παρατηρούμε ότι:

- ◆ Το latency είναι μικρότερο όταν το car0 επικοινωνεί κατευθείαν με τον client μέσω μίας μόνο κεραίας, ενώ αυξάνεται όταν επικοινωνεί είτε μέσω τρίτου είτε με bicastng.
- ◆ Το throughput, παρατηρούμε ότι, αυξάνεται κατά ένα μεγάλο ποσοστό στις περιπτώσεις που έχουμε bicastng, γεγονός που είναι λογικό καθώς ο συνολικός αριθμός πακέτων που μεταδίδονται είναι ο διπλάσιος.
- ◆ Όσον αφορά το packet loss, στην πλειονότητα των περιπτώσεων είναι 0 - καθώς κανένα πακέτο δεν χάνεται – ωστόσο, στα παραπάνω screenshots παρατηρείτε την περίπτωση όπου ένα πακέτο έχει χαθεί λόγω χρήσης UDP πρωτοκόλλου.