

Διαχείριση Δικτύων – Ευφυή Δίκτυα

ΟΝΟΜΑ: ΠΑΠΑΔΟΠΟΥΛΛΟΣ ΜΙΧΑΛΗΣ

A.M: 03114702

ΟΝΟΜΑ: ΚΑΡΔΑΡΗΣ ΧΑΡΑΛΑΜΠΟΣ

A.M: 03114074

ΟΜΑΔΑ: netmg034

6η Ομάδα Ασκήσεων

Στόχοι εργαστηριακής άσκησης:

A) Εξοικείωση με τις ερευνητικές υποδομές του GENI (Global Environment for Network Innovations) και ορισμένα διαθέσιμα εργαλεία που παρέχονται.

B) Σχεδίαση και δημιουργία ενός απλού δικτυακού πειράματος χρησιμοποιώντας πόρους του GENI.

A.1. Συνδεθείτε στο <https://portal.geni.net/> χρησιμοποιώντας το προσωπικό όνομα χρήστη και τον κωδικό πρόσβασης του πολυτεχνείου (υποστηρίζεται academic login μέσω shibboleth) και ως όνομα οργανισμού: "National Technical University of Athens". Επιλέξτε το project: *Netman2019* και δημιουργήστε ένα δικό σας slice με το όνομα της εργαστηριακής σας ομάδας (π.χ. netmg000).

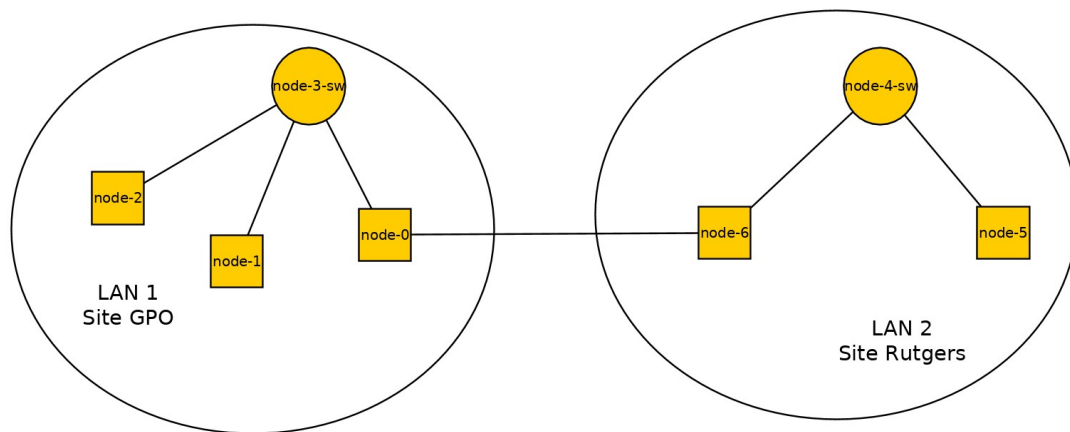
A.2. Χρησιμοποιώντας κάποιο από τα παρεχόμενα εργαλεία (π.χ. Slice Jacks, Flack, κλπ.) ένας χρήστης δημιουργεί/διαχειρίζεται δικτυακά πειράματα ζητώντας με ασφαλή τρόπο (με τη βοήθεια του πρωτοκόλλου SSL) υπολογιστικούς/δικτυακούς πόρους από έναν ή περισσότερους GENI Aggregates (περισσότερες πληροφορίες για τη σχετική ορολογία: <http://groups.geni.net/geni/wiki/GENIGlossary>). Στο κεντρικό μενού επιλέξτε την καρτέλα Profile->SSL και δημιουργήστε το SSL πιστοποιητικό σας.

A.3. Προκειμένου να συνδέεστε στους πόρους (π.χ. εικονικές μηχανές - VMs) που δεσμεύετε, θα χρησιμοποιείτε το ασφαλές ssh (secure shell) πρωτόκολλο. Από το μηχάνημα maria.netmode.ntua.gr, δημιουργήστε RSA SSH κλειδιά (ιδιωτικό και δημόσιο) και ανεβάστε το SSH δημόσιο κλειδί σας στο GENI Experimenter Portal (Profile->SSH Keys).

A.4. Χρησιμοποιήστε το εργαλείο Slice Jacks (beta) για το «slice» που έχετε δημιουργήσει (επιλογή "Add Resources") και σχεδιάστε την τοπολογία που απεικονίζεται στην Εικόνα 1 (για τη δημιουργία των κόμβων-hosts επιλέξτε Node Type: default-vm και Disk Image: any, ενώ για τους OF OVS κόμβους επιλέξτε Node Type: emulab-xen και Disk Image: Ubuntu 14 with OVS, by Niky). Αφού ρυθμίσετε κατάλληλα τους κόμβους που σχεδιάσατε με βάση τις πληροφορίες που σας δίνονται στο παρακάτω σχήμα (π.χ. οι 3 πρώτες εικονικές μηχανές να ανήκουν στο τοπικό υποδίκτυο 10.10.1.0/24, κτλ.) δημιουργήστε την σχεδιασμένη τοπολογία και δεσμεύστε τους αντίστοιχους πόρους πατώντας 'Δέσμευση Πόρων' ('Reserve

Resources'). Στο link μεταξύ των 2 sites προτιμήστε (Stitched Ethernet ή EGRE Tunnel).

Η τοπολογία που φτιαξαμε μέσω του εργαλείου Jacks φαίνεται πιο κάτω:



Μεταξύ των nodes node-0 (GPO) και node-6 (Rutgers) υλοποιείται stitch-ethernet tunnelling.

B.1. Συνδεθείτε με ssh στους κόμβους-εικονικές μηχανές που έχετε δημιουργήσει (κατά προτίμηση από το μηχάνημα maria.netmode.ntua.gr). Για να βρείτε τις πληροφορίες σύνδεσης για κάθε δεσμευμένο κόμβο, πατήστε στο «slice» σας και στη συνέχεια επιλέξτε Details. Πόσα interfaces έχει κάθε κόμβος και ποιές IP διευθύνσεις έχουν εκχωρηθεί στο καθένα;

Site GPO

node-0 →

mechatr0@node-0:~\$ **ip a**

1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default qlen 1

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid_lft forever preferred_lft forever

inet6 ::1/128 scope host

valid_lft forever preferred_lft forever

2: **eth0**: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000

link/ether 02:bd:73:d7:1f:65 brd ff:ff:ff:ff:ff:ff

inet **172.17.5.7**/12 brd 172.31.255.255 scope global eth0

valid_lft forever preferred_lft forever

inet6 fe80::bd:73ff:fed7:1f65/64 scope link

valid_lft forever preferred_lft forever

3: **eth1**: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000

link/ether 02:3d:6f:55:cf:ac brd ff:ff:ff:ff:ff:ff

inet **10.10.1.1**/24 brd 10.10.1.255 scope global eth1

valid_lft forever preferred_lft forever

inet6 fe80::3d:6fff:fe55:cfac/64 scope link

valid_lft forever preferred_lft forever

4: **eth2**: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000

link/ether 02:05:0c:41:50:01 brd ff:ff:ff:ff:ff:ff

inet **192.168.1.1**/24 brd 192.168.1.255 scope global eth2

valid_lft forever preferred_lft forever

inet6 fe80::5:cff:fe41:5001/64 scope link

valid_lft forever preferred_lft forever

node-1 →

mechatr0@node-1:~\$ ip a

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000
    link/ether 02:98:8f:ff:27:03 brd ff:ff:ff:ff:ff:ff
    inet 172.17.2.12/12 brd 172.31.255.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::98:8fff:feff:2703/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000
    link/ether 02:59:45:23:e4:e8 brd ff:ff:ff:ff:ff:ff
    inet 10.10.1.3/24 brd 10.10.1.255 scope global eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::59:45ff:fe23:e4e8/64 scope link
        valid_lft forever preferred_lft forever
```

node-2 →

mechatr0@node-2:~\$ ip a

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000
    link/ether 02:5c:37:38:f9:3c brd ff:ff:ff:ff:ff:ff
    inet 172.17.4.12/12 brd 172.31.255.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::5c:37ff:fe38:f93c/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000
    link/ether 02:28:d7:50:11:76 brd ff:ff:ff:ff:ff:ff
    inet 10.10.1.5/24 brd 10.10.1.255 scope global eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::28:d7ff:fe50:1176/64 scope link
        valid_lft forever preferred_lft forever
```

node-3-sw →

mechatr0@node-3-sw:~\$ **ip a**

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000
    link/ether 02:ad:d7:9e:3d:b8 brd ff:ff:ff:ff:ff:ff
    inet 172.17.2.17/12 brd 172.31.255.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::ad:d7ff:fe9e:3db8/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master ovs-system state UP group default qlen 1000
    link/ether 02:12:f1:10:1b:49 brd ff:ff:ff:ff:ff:ff
    inet 10.10.1.2/24 brd 10.10.1.255 scope global eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::12:f1ff:fe10:1b49/64 scope link
        valid_lft forever preferred_lft forever
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master ovs-system state UP group default qlen 1000
    link/ether 02:41:d6:40:4b:91 brd ff:ff:ff:ff:ff:ff
    inet 10.10.1.4/24 brd 10.10.1.255 scope global eth2
        valid_lft forever preferred_lft forever
    inet6 fe80::41:d6ff:fe40:4b91/64 scope link
        valid_lft forever preferred_lft forever
5: eth3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master ovs-system state UP group default qlen 1000
    link/ether 02:21:8a:e0:01:04 brd ff:ff:ff:ff:ff:ff
    inet 10.10.1.6/24 brd 10.10.1.255 scope global eth3
        valid_lft forever preferred_lft forever
    inet6 fe80::21:8aff:fee0:104/64 scope link
        valid_lft forever preferred_lft forever
6: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN
group default
    link/ether 12:b6:63:6d:72:81 brd ff:ff:ff:ff:ff:ff
7: br0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group
default
    link/ether 2a:8a:32:cd:de:40 brd ff:ff:ff:ff:ff:ff
```

node-4-sw →

mechatr0@node-4-sw:~\$ **ip a**

1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid_lft forever preferred_lft forever

inet6 ::1/128 scope host

valid_lft forever preferred_lft forever

2: **eth0**: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000

link/ether 02:5e:c9:49:54:b5 brd ff:ff:ff:ff:ff:ff

inet **172.17.2.9**/12 brd 172.31.255.255 scope global eth0

valid_lft forever preferred_lft forever

inet6 fe80::5e:c9ff:fe49:54b5/64 scope link

valid_lft forever preferred_lft forever

3: **eth1**: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master ovs-system state UP group default qlen 1000

link/ether 02:bd:79:64:33:a2 brd ff:ff:ff:ff:ff:ff

inet **10.10.2.4**/24 brd 10.10.2.255 scope global eth1

valid_lft forever preferred_lft forever

inet6 fe80::bd:79ff:fe64:33a2/64 scope link

valid_lft forever preferred_lft forever

4: **eth2**: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
master ovs-system state UP group default qlen 1000

link/ether 02:3a:fa:33:81:66 brd ff:ff:ff:ff:ff:ff

inet **10.10.2.2**/24 brd 10.10.2.255 scope global eth2

valid_lft forever preferred_lft forever

inet6 fe80::3a:faff:fe33:8166/64 scope link

valid_lft forever preferred_lft forever

5: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN
group default

link/ether a2:4c:10:cd:8e:73 brd ff:ff:ff:ff:ff:ff

6: br0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group
default

link/ether ba:18:52:d3:e6:4d brd ff:ff:ff:ff:ff:ff

mechatr0@node-4-sw:~\$ ping 10.10.1.1

PING 10.10.1.1 (10.10.1.1) 56(84) bytes of data.

Node-5 →

mechatr0@node-5:~\$ **ip a**

1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default qlen 1

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid_lft forever preferred_lft forever

inet6 ::1/128 scope host

valid_lft forever preferred_lft forever

2: **eth0**: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000

link/ether 02:3d:2a:e2:4d:f7 brd ff:ff:ff:ff:ff:ff

inet **172.17.3.7**/12 brd 172.31.255.255 scope global eth0

valid_lft forever preferred_lft forever

inet6 fe80::3d:2aff:fee2:4df7/64 scope link

valid_lft forever preferred_lft forever

3: **eth1**: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000

link/ether 02:44:7e:91:89:ec brd ff:ff:ff:ff:ff:ff

inet **10.10.2.3**/24 brd 10.10.2.255 scope global eth1

valid_lft forever preferred_lft forever

inet6 fe80::44:7eff:fe91:89ec/64 scope link

valid_lft forever preferred_lft forever

node-6 →

mechatr0@node-6:~\$ **ip a**

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000
    link/ether 02:8d:46:d8:75:06 brd ff:ff:ff:ff:ff:ff
    inet 172.17.1.10/12 brd 172.31.255.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::8d:46ff:fed8:7506/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000
    link/ether 02:f4:eb:c1:f8:e3 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.2/24 brd 192.168.1.255 scope global eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::f4:ebff:fec1:f8e3/64 scope link
        valid_lft forever preferred_lft forever
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP group default qlen 1000
    link/ether 02:21:b7:e2:42:45 brd ff:ff:ff:ff:ff:ff
    inet 10.10.2.1/24 brd 10.10.2.255 scope global eth2
        valid_lft forever preferred_lft forever
    inet6 fe80::21:b7ff:fee2:4245/64 scope link
        valid_lft forever preferred_lft forever
```

Η αντιστοιχία των κόμβων μας με την εκφώνηση είναι

```
node-0 → LAN 1 node 2 (tunnel)
node-1 → LAN 1 node 1
node-2 → LAN 1 node 3
node-3-sw → LAN 1 node of ovs
node-4-sw → LAN 2 node of ovs
node-5 → LAN 2 node 2
node-6 → LAN 2 node 1 (tunnel)
```

B.2. Δοκιμάστε να τρέξετε την εντολή ping από τον πάνω αριστερά κόμβο (LAN 1 – node 1) προς τον πάνω δεξιά κόμβο (LAN 1 – node 3). Τι παρατηρείτε και για ποιο λόγο;

```
mechatr0@node-1:~$ ping -c3 10.10.1.5
PING 10.10.1.5 (10.10.1.5) 56(84) bytes of data.
From 10.10.1.5 icmp_seq=1 Destination Host Unreachable
From 10.10.1.5 icmp_seq=2 Destination Host Unreachable
From 10.10.1.5 icmp_seq=3 Destination Host Unreachable
```

```
--- 10.10.1.5 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2015ms
pipe 3
```

Παρατηρούμε ότι το ping αποτυγχάνει. Ο λόγος είναι η εικονική φύση του switch την οποία πρέπει να ενημερώσουμε για την ύπαρξη των συνδεσεων στα interfaces του.

Στη συνέχεια προγραμματίστε το OVS του κόμβου OF OVS του LAN 1 ώστε να λειτουργεί ως Layer 2 switch και προσθέστε τα interfaces του κόμβου που τον συνδέουν με τα nodes 1, 2 και 3. Επαναλάβετε την δοκιμή με την εντολή ping. Τι παρατηρείτε; Επαναλάβετε την ίδια διαδικασία στο OF OVS του LAN 2 και ελέγξτε πάλι με ping. Δικαιολογήστε τις απαντήσεις σας.

Συνδεόμαστε στο switch του LAN1 και κάνουμε τις απαραίτητες ρυθμίσεις ώστε τα nodes να μπορούν να επικοινωνούν μεταξύ τους.

```
mechatr0@node-3-sw:~$ sudo ovs-vsctl add-br br0
mechatr0@node-3-sw:~$ sudo ovs-vsctl add-port br0 eth1
mechatr0@node-3-sw:~$ sudo ovs-vsctl add-port br0 eth2
mechatr0@node-3-sw:~$ sudo ovs-vsctl add-port br0 eth3
```

```
mechatr0@node-3-sw:~$ sudo ovs-vsctl show
89362c3a-1d64-451b-977d-7b2718a1ba20
```

```
Bridge "br0"
  Port "br0"
    Interface "br0"
      type: internal
  Port "eth1"
    Interface "eth1"
  Port "eth2"
    Interface "eth2"
  Port "eth3"
    Interface "eth3"
ovs_version: "2.3.1"
```

```
mechatr0@node-1:~$ ping -c 3 10.10.1.5
```

```
PING 10.10.1.5 (10.10.1.5) 56(84) bytes of data.
64 bytes from 10.10.1.5: icmp_seq=1 ttl=64 time=1.44 ms
64 bytes from 10.10.1.5: icmp_seq=2 ttl=64 time=1.00 ms
64 bytes from 10.10.1.5: icmp_seq=3 ttl=64 time=1.02 ms
```

```
--- 10.10.1.1 ping statistics ---
```

```
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.001/1.158/1.447/0.207 ms
```

Η επικοινωνία, μετά την παραμετροποίηση του OVS switch είναι εφικτή όπως φαίνεται και πιο πάνω.

Αντίστοιχα και για το switch του LAN 2

```
mechatr0@node-4-sw:~$ sudo ovs-vsctl add-br br0
mechatr0@node-4-sw:~$ sudo ovs-vsctl add-port br0 eth1
mechatr0@node-4-sw:~$ sudo ovs-vsctl add-port br0 eth2
mechatr0@node-4-sw:~$ sudo ovs-vsctl show
89362c3a-1d64-451b-977d-7b2718a1ba20
Bridge "br0"
  Port "eth2"
    Interface "eth2"
  Port "eth1"
    Interface "eth1"
  Port "br0"
    Interface "br0"
      type: internal
ovs_version: "2.3.1"
```

```
mechatr0@node-5:~$ ping -c 3 10.10.2.1  
PING 10.10.2.1 (10.10.2.1) 56(84) bytes of data.  
64 bytes from 10.10.2.1: icmp_seq=1 ttl=64 time=1.95 ms  
64 bytes from 10.10.2.1: icmp_seq=2 ttl=64 time=0.679 ms  
64 bytes from 10.10.2.1: icmp_seq=3 ttl=64 time=0.657 ms  
  
--- 10.10.2.2 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2001ms  
rtt min/avg/max/mdev = 0.657/1.095/1.951/0.606 ms
```

B.3. Με τη χρήση του προγράμματος ping ελέγξτε την συνδεσιμότητα μεταξύ του πάνω αριστερά κόμβου (LAN 1 – node 1) και του κάτω δεξιά κόμβου (LAN 2 – node 2) της Εικόνας 1. Τι παρατηρείτε και για ποιο λόγο; Δικαιολογήστε την απάντησή σας.

```
mechatr0@node-1:~$ ping -c 3 10.10.2.3
PING 10.10.2.3 (10.10.2.3) 56(84) bytes of data.
From 10.10.1.3 icmp_seq=1 Destination Host Unreachable
From 10.10.1.3 icmp_seq=2 Destination Host Unreachable
From 10.10.1.3 icmp_seq=3 Destination Host Unreachable
```

```
--- 10.10.2.3 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2015ms
pipe 3
```

Όπως και πριν, η επικοινωνία μεταξύ των δύο μηχανημάτων αποτυγχάνει. Ωστόσο, ο λόγος σ' αυτήν την περίπτωση είναι ότι τα δυο μηχανήματα βρίσκονται σε ξεχωριστά LAN

```
mechatr0@node-1:~$ route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
0.0.0.0 172.16.0.1 0.0.0.0 UG 0 0 0 eth0
10.0.0.0 10.10.1.3 255.0.0.0 UG 0 0 0 eth1
10.10.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1
172.16.0.0 0.0.0.0 255.240.0.0 U 0 0 0 eth0
192.168.0.0 10.10.1.3 255.255.0.0 UG 0 0 0 eth1
```

Παρατηρούμε ότι δεν υπάρχει κανόνας δρομολόγησης προς το LAN 2 που να χρησιμοποιεί την tunneling ζεύξη.

B.4. Καταγράψτε τον πίνακα δρομολόγησης για ένα κόμβο με Tunneling Interface. Προβείτε στις απαραίτητες ενέργειες προκειμένου να δρομολογούνται επιτυχώς πακέτα μεταξύ του πάνω αριστερά κόμβου (LAN 1) και του κάτω δεξιά κόμβου (LAN 2) της Εικόνας 1. Καταγράψτε τη διαδρομή μεταξύ των δυο ανωτέρω κόμβων με τη χρήση του εργαλείου traceroute.

Ο κόμβος με tunneling interface στο LAN 1 είναι ο node-0

```
mechatr0@node-0:~$ route -n
```

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	172.16.0.1	0.0.0.0	UG 0	0	0	0	eth0
10.0.0.0	10.10.1.2	255.0.0.0	UG 0	0	0	0	eth1
10.10.1.0	0.0.0.0	255.255.255.0	U 0	0	0	0	eth1
172.16.0.0	0.0.0.0	255.240.0.0	U 0	0	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U 0	0	0	0	eth2

Θέλουμε να επιτύχουμε δρομολόγηση μεταξύ των κόμβων node-1 (10.10.1.3) και node-5 (10.10.2.3)

Καταρχάς πρέπει να εισάγουμε τον κανόνα για προώθηση των πακέτων στο LAN 2 από τον κόμβο του LAN με tunneling interface

```
mechatr0@node-0:~$ sudo ip route add 10.10.2.0/24 via 192.168.1.2
```

```
mechatr0@node-0:~$ route -n
```

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	172.16.0.1	0.0.0.0	UG 0	0	0	0	eth0
10.0.0.0	10.10.1.2	255.0.0.0	UG 0	0	0	0	eth1
10.10.1.0	0.0.0.0	255.255.255.0	U 0	0	0	0	eth1
10.10.2.0	192.168.1.2	255.255.255.0	UG 0	0	0	0	eth2
172.16.0.0	0.0.0.0	255.240.0.0	U 0	0	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U 0	0	0	0	eth2

Ακολούθως πρέπει να εισάγουμε στο routetable του node-1 κανόνα για να προωθεί πακέτα με προορισμό το LAN 2 στο node-0 (10.10.1.1), που θα λειτουργεί σαν gateway προς το LAN 2.

```
mechatr0@node-1:~$ sudo ip route add 10.10.2.0/24 via 10.10.1.1
```

```
mechatr0@node-1:~$ route -n
```

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	172.16.0.1	0.0.0.0	UG 0	0	0	0	eth0
10.0.0.0	10.10.1.4	255.0.0.0	UG 0	0	0	0	eth1
10.10.1.0	0.0.0.0	255.255.255.0	U 0	0	0	0	eth1
10.10.2.0	10.10.1.1	255.255.255.0	UG 0	0	0	0	eth1
172.16.0.0	0.0.0.0	255.240.0.0	U 0	0	0	0	eth0
192.168.0.0	10.10.1.4	255.255.0.0	UG 0	0	0	0	eth1
192.168.1.0	10.10.1.1	255.255.255.0	UG 0	0	0	0	eth1

Εντελώς ανάλογα και τους κόμβους του LAN 2 ώστε να είναι δυνατές οι απαντήσεις.

```
mechatr0@node-6:~$ sudo ip route add 10.10.1.0/24 via 192.168.1.1
```

```
mechatr0@node-6:~$ route -n
```

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	172.16.0.1	0.0.0.0	UG 0	0	0	0	eth0
10.0.0.0	10.10.2.2	255.0.0.0	UG 0	0	0	0	eth2
10.10.1.0	192.168.1.1	255.255.255.0	UG 0	0	0	0	eth1
10.10.2.0	0.0.0.0	255.255.255.0	U 0	0	0	0	eth2
172.16.0.0	0.0.0.0	255.240.0.0	U 0	0	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U 0	0	0	0	eth1

```
mechatr0@node-5:~$ sudo ip route add 10.10.1.0/24 via 10.10.2.1
```

```
mechatr0@node-5:~$ route -n
```

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	172.16.0.1	0.0.0.0	UG 0	0	0	0	eth0
10.0.0.0	10.10.2.4	255.0.0.0	UG 0	0	0	0	eth1
10.10.1.0	10.10.2.1	255.255.255.0	UG 0	0	0	0	eth1
10.10.2.0	0.0.0.0	255.255.255.0	U 0	0	0	0	eth1
172.16.0.0	0.0.0.0	255.240.0.0	U 0	0	0	0	eth0
192.168.0.0	10.10.2.4	255.255.0.0	UG 0	0	0	0	eth1
192.168.1.0	10.10.2.1	255.255.255.0	UG 0	0	0	0	eth1

Πλέον είναι δυνατή η επικοινωνία μεταξύ των 2 κόμβων.

```
mechatr0@node-1:~$ ping -c 3 10.10.2.3
```

PING 10.10.2.3 (10.10.2.3) 56(84) bytes of data.

64 bytes from 10.10.2.3: icmp_seq=1 ttl=62 time=12.4 ms

64 bytes from 10.10.2.3: icmp_seq=2 ttl=62 time=11.7 ms

64 bytes from 10.10.2.3: icmp_seq=3 ttl=62 time=11.6 ms

--- 10.10.2.3 ping statistics ---

3 packets transmitted, 3 received, 0% packet loss, time 2002ms

rtt min/avg/max/mdev = 11.679/11.978/12.478/0.377 ms

B.5. Τι παρατηρείτε με τους χρόνους καθυστέρησης; Σε ποιο τμήμα της διαδρομής μέχρι τον προορισμό εισάγεται η μεγαλύτερη καθυστέρηση; Πως δικαιολογείται; Εξηγήστε αναλυτικά.

```
mechatr0@node-1:~$ traceroute 10.10.2.3
traceroute to 10.10.2.3 (10.10.2.3), 30 hops max, 60 byte packets
 1 node-0-link-0 (10.10.1.1)  2.196 ms  2.160 ms  2.133 ms
 2 node-6-link-5 (192.168.1.2) 12.194 ms 12.176 ms 12.152 ms
 3 10.10.2.3 (10.10.2.3) 15.384 ms 15.369 ms 15.335 ms
```

Παρατηρούμε μεγαλύτερη καθυστέρηση στην tunneling σύνδεση, κάτι που είναι λογικό, αφού συνδέει υποδίκτυα που βρίσκονται σε διαφορετικές φυσικές τοποθεσίες και άρα έχουμε επιπλέον καθυστερήσεις διάδοσης των μηνυμάτων.

B.6. Θα μπορούσαμε να σχεδιάσουμε το σκαρίφημα του GR-IX με χρήση των εργαλείων ring και traceroute από κάποιον κόμβο της υποδομής του GENI; Επιβεβαιώστε στην πράξη την απάντηση που δώσατε στο 4^ο ερώτημα της δεύτερης εργαστηριακής άσκησης.

Όχι δεν είναι δυνατό καθώς κάθε μήνυμα από τον αντίστοιχο κόμβο του δικτύου 1, θα δρομολογηθεί μέσω του αντίστοιχου Tier 1 παρόχου, που παρέχει internet στον τελικό κόμβο. Αυτό σημαίνει ότι δεν θα περάσει από κανένα switch του GR-IX, που εξυπηρετεί μόνο τοπικά στον ελληνικό χώρο υποδίκτυα και άρα δεν είμαστε σε θέση να σχεδιάσουμε το σκαρίφημα του GR-IX.

```
mechatr0@node-0:~$ traceroute www.forthnet.gr
traceroute to www.forthnet.gr (62.1.46.47), 30 hops max, 60 byte packets
 1 pc5.instagenei.gpolab.bbn.com (192.1.242.144)  0.566 ms  0.545 ms  0.523 ms
 2 192.1.242.129 (192.1.242.129)  1.127 ms  1.131 ms  1.113 ms
 3 192.1.101.5 (192.1.101.5)  1.039 ms  1.021 ms  0.998 ms
 4 te0-0-1-1.214.nr11.b002250-1.bos01.atlas.cogentco.com (38.140.156.97) 1.974 ms 1.951 ms 2.008 ms
 5 te0-3-0-0.agr22.bos01.atlas.cogentco.com (154.24.14.205) 1.904 ms te0-3-0-0.agr21.bos01.atlas.cogentco.com (154.24.14.201) 1.867 ms
 6 be3663.ccr32.bos01.atlas.cogentco.com (154.54.87.101) 1.805 ms be3662.ccr31.bos01.atlas.cogentco.com (154.54.87.97) 1.758 ms be3663
.ccr32.bos01.atlas.cogentco.com (154.54.87.101) 1.456 ms
 7 be3471.ccr41.jfk02.atlas.cogentco.com (154.54.40.154) 7.447 ms be3472.ccr42.jfk02.atlas.cogentco.com (154.54.46.34) 7.130 ms be3471
.ccr41.jfk02.atlas.cogentco.com (154.54.40.154) 7.001 ms
 8 be3496.ccr31.jfk10.atlas.cogentco.com (154.54.0.142) 7.241 ms be3295.ccr31.jfk05.atlas.cogentco.com (154.54.80.2) 6.997 ms 6.975 m
s
 9 telecomitalia.jfk05.atlas.cogentco.com (154.54.11.142) 6.995 ms telecomitalia.jfk10.atlas.cogentco.com (154.54.10.218) 7.146 ms 7.1
22 ms
10 93.186.129.151 (93.186.129.151) 129.760 ms 213.144.178.151 (213.144.178.151) 138.688 ms 93.186.129.149 (93.186.129.149) 138.467 ms
11 213.144.178.43 (213.144.178.43) 134.589 ms 141.705 ms 134.530 ms
12 distr-klm-01be3.forthnet.gr (213.16.247.22) 140.435 ms 139.101 ms 139.251 ms
13 gil-24.serv-klm-01.forthnet.gr (213.16.247.125) 133.495 ms 136.265 ms 133.593 ms
14 * * *
15 * * *
16 * * *
17 * * *
18 * * *
19 * * *
20 * * *
21 * * *
22 * * *
23 * * *
24 * * *
25 * * *
26 * * *
27 * * *
28 * * *
29 * * *
30 * * *
```

Όπως βλέπουμε στην traceroute το μήνυμα περνάει από κόμβους που διαχειρίζονται από τη Telecom Italia που είναι ένας Tier 1 πάροχος.

Εργαλεία που θα χρειαστείτε:

ssh-keygen, chmod, ssh, ip, ifconfig, netstat, route, ping, traceroute