

Samantha Neri, Khushveen Kaur Umra
CprE 489, Section 2
5th April 2023

Lab 6 Report

Contribution: Samantha Neri (50%), Khushveen Kaur Umra (50%)

1) Summarize what you learned in a few paragraphs. (20 points)

With this lab, we gained experience with using static routing protocol and manually updating routing tables. We used GENI, to create a slice for this lab and read our nodes. Then we set up routing by running an ssh command for all three of our nodes (A, B, and C) then got information by loading the slice from the portal.

After that we used the “route” command for all of the nodes in order to show their respective routing tables. Then we tried pinging the two IP addresses from node A to nodes B and C; however we noticed that two IP addresses (one from B and C) were not able to be pinged. We realized that these nodes could not be pinged since they were not directly connected to node A. We compensated for this by setting up a static route. Once this was established, we cleaned up the slice by using delete in the manage resources panel of the slice page; leaving us with an empty canvas.

2g) Take a screenshot of your slice when all the nodes are ready and include it in your report. (5 points)

Status	Aggregate
READY	Illinois InstaGENI

Aggregate Illinois InstaGENI's Resources:

Node #1:

Status	Client ID	Component ID	Expiration	Type	Hostname
READY	node-a	pc3	2023-04-12T16:30:20.000Z	default-vm	node-a.Umra-lab6.ch-geni-net.instageni.illinois.edu
Login	ssh vsc@pc3.instageni.illinois.edu -p 29210 ssh ruminski@pc3.instageni.illinois.edu -p 29210 ssh daji@pc3.instageni.illinois.edu -p 29210 ssh kumra@pc3.instageni.illinois.edu -p 29210				
Interfaces		MAC		Layer 3	
interface-0	pc3:lo0	026ea63b6e8c		ipv4: 192.168.3.10	
interface-2	pc3:lo0	02320bc937f4		ipv4: 192.168.1.10	

Node #2:

Status	Client ID	Component ID	Expiration	Type	Hostname
READY	node-b	pc3	2023-04-12T16:30:20.000Z	default-vm	node-b.Umra-lab6.ch-geni-net.instageni.illinois.edu
Login	ssh vsc@pc3.instageni.illinois.edu -p 29211 ssh ruminski@pc3.instageni.illinois.edu -p 29211 ssh daji@pc3.instageni.illinois.edu -p 29211 ssh kumra@pc3.instageni.illinois.edu -p 29211				
Interfaces		MAC		Layer 3	
interface-3	pc3:lo0	024bcc643e6b		ipv4: 192.168.1.11	
interface-4	pc3:lo0	020073acd21		ipv4: 192.168.2.11	

Node #3:

Status	Client ID	Component ID	Expiration	Type	Hostname
READY	node-c	pc3	2023-04-12T16:30:20.000Z	default-vm	node-c.Umra-lab6.ch-geni-net.instageni.illinois.edu
Login	ssh vsc@pc3.instageni.illinois.edu -p 29212 ssh ruminski@pc3.instageni.illinois.edu -p 29212 ssh daji@pc3.instageni.illinois.edu -p 29212 ssh kumra@pc3.instageni.illinois.edu -p 29212				
Interfaces		MAC		Layer 3	
interface-1	pc3:lo0	027471025d6d		ipv4: 192.168.3.12	
interface-5	pc3:lo0	022006032b60		ipv4: 192.168.2.12	

4) Execute the route command in each of the three nodes to show their respective routing tables. Include a screenshot of the tables in your lab report. (10 points)

- Node A

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

```
kumra@node-a:~$ 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      1024  0      0 eth0
172.16.0.0       0.0.0.0         255.240.0.0      U        0    0      0 eth0
172.16.0.1       0.0.0.0         255.255.255.255 UH      1024  0      0 eth0
192.168.1.0      0.0.0.0         255.255.255.0    U        0    0      0 eth1
192.168.3.0      0.0.0.0         255.255.255.0    U        0    0      0 eth2
kumra@node-a:~$
```

- Node B

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

```
kumra@node-b:~$ 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      1024  0      0 eth0
172.16.0.0       0.0.0.0         255.240.0.0      U        0    0      0 eth0
172.16.0.1       0.0.0.0         255.255.255.255 UH      1024  0      0 eth0
192.168.0.0      192.168.2.12    255.255.0.0      UG        0    0      0 eth1
192.168.1.0      0.0.0.0         255.255.255.0    U        0    0      0 eth2
192.168.2.0      0.0.0.0         255.255.255.0    U        0    0      0 eth1
192.168.3.8      192.168.1.10    255.255.255.252 UG        0    0      0 eth2
kumra@node-b:~$
```

- Node C

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

```
kumra@node-c:~$ 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      1024  0      0 eth0
172.16.0.0       0.0.0.0         255.240.0.0      U        0    0      0 eth0
172.16.0.1       0.0.0.0         255.255.255.255 UH      1024  0      0 eth0
192.168.2.0      0.0.0.0         255.255.255.0    U        0    0      0 eth1
192.168.3.0      0.0.0.0         255.255.255.0    U        0    0      0 eth2
kumra@node-c:~$
```

5) From node A, try to ping the other addresses of nodes B and C (two IP addresses for each node). Include a screenshot of the ping outputs and explain the results. (10 points)

→Node B
(192.168.1.11)

```
kumra@node-a:~$ ping 192.168.1.11
PING 192.168.1.11 (192.168.1.11) 56(84) bytes of data.
64 bytes from 192.168.1.11: icmp_seq=1 ttl=64 time=1.04 ms
64 bytes from 192.168.1.11: icmp_seq=2 ttl=64 time=0.591 ms
64 bytes from 192.168.1.11: icmp_seq=3 ttl=64 time=0.563 ms
64 bytes from 192.168.1.11: icmp_seq=4 ttl=64 time=0.520 ms
64 bytes from 192.168.1.11: icmp_seq=5 ttl=64 time=0.628 ms
64 bytes from 192.168.1.11: icmp_seq=6 ttl=64 time=0.611 ms
64 bytes from 192.168.1.11: icmp_seq=7 ttl=64 time=0.641 ms
64 bytes from 192.168.1.11: icmp_seq=8 ttl=64 time=0.635 ms
64 bytes from 192.168.1.11: icmp_seq=9 ttl=64 time=0.497 ms
64 bytes from 192.168.1.11: icmp_seq=10 ttl=64 time=0.535 ms
^C
--- 192.168.1.11 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9013ms
rtt min/avg/max/mdev = 0.497/0.626/1.040/0.146 ms
```

To (192.168.2.11)

```
kumra@node-a:~$ ping 192.168.2.11
PING 192.168.2.11 (192.168.2.11) 56(84) bytes of data.
^C
--- 192.168.2.11 ping statistics ---
16 packets transmitted, 0 received, 100% packet loss, time 15357ms
```

Results:

192.168.1.11: We were able to successfully ping this IP address from Node A, as shown in the screenshot above. There was no packet loss as well. This is because, Node A had a direct route set up to this IP address, which is in B, and hence we were able to ping it.

192.168.2.11: We are unable to ping this IP address because Node A is not connected to this IP address, which is in Node B. Which is why there was a 100% packet loss seen. In order to ping this IP address successfully, we need to set up a static route.

→Node C
(192.168.3.12)

```
kumra@node-a:~$ ping 192.168.3.12
PING 192.168.3.12 (192.168.3.12) 56(84) bytes of data.
64 bytes from 192.168.3.12: icmp_seq=1 ttl=64 time=0.960 ms
64 bytes from 192.168.3.12: icmp_seq=2 ttl=64 time=0.543 ms
64 bytes from 192.168.3.12: icmp_seq=3 ttl=64 time=0.603 ms
64 bytes from 192.168.3.12: icmp_seq=4 ttl=64 time=0.535 ms
64 bytes from 192.168.3.12: icmp_seq=5 ttl=64 time=0.604 ms
64 bytes from 192.168.3.12: icmp_seq=6 ttl=64 time=0.639 ms
64 bytes from 192.168.3.12: icmp_seq=7 ttl=64 time=0.544 ms
64 bytes from 192.168.3.12: icmp_seq=8 ttl=64 time=0.607 ms
64 bytes from 192.168.3.12: icmp_seq=9 ttl=64 time=0.537 ms
64 bytes from 192.168.3.12: icmp_seq=10 ttl=64 time=0.551 ms
^C
--- 192.168.3.12 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9013ms
rtt min/avg/max/mdev = 0.535/0.612/0.960/0.122 ms
```

To 192.168.2.12

```
kumra@node-a:~$ ping 192.168.2.12
PING 192.168.2.12 (192.168.2.12) 56(84) bytes of data.
^C
--- 192.168.2.12 ping statistics ---
92 packets transmitted, 0 received, 100% packet loss, time 93174ms
```

Results:

192.168.3.12: We were able to successfully ping this IP address from Node A, as shown in the screenshot above. There was no packet loss as well. This is because, Node A had a direct route set up to this IP address, which is in C, and hence we were able to ping it.

192.168.2.12: We are unable to ping this IP address because Node A is not connected to this IP address, which is in Node C. Which is why there was a 100% packet loss. In order to ping this IP address successfully, we need to set up a static route.

6) What happens when you traceroute from A to IP address 192.168.2.12 before you set up the static routes? Why? Include a screenshot of the traceroute output in your lab report. (10 points)

→ When we try to traceroute from A to the IP address 192.168.2.12 before setting up the static routes, we were not able to reach that IP address because it is on Node C and there is not path set up to that node and no static route for tracing that route. Which is why, the majority of the packets are skipped.

```
kumra@node-a:~$ traceroute 192.168.2.12
traceroute to 192.168.2.12 (192.168.2.12), 30 hops max, 60 byte packets
 1  pc3.instageni.illinois.edu (72.36.65.22)  0.359 ms  0.245 ms  0.315 ms
 2  dyn-72-36-65-254.instageni.illinois.edu (72.36.65.254)  0.540 ms dyn-72-36-65-253.instageni.illinois.edu (72.36.65.253)
    0.401 ms dyn-72-36-65-254.instageni.illinois.edu (72.36.65.254)  0.630 ms
 3  t-core9-2.gw.uiuc.edu (72.36.80.93)  0.791 ms  0.949 ms t-core1-1.gw.uiuc.edu (72.36.80.89)  0.808 ms
 4  172.20.20.194 (172.20.20.194)  1.194 ms t-ur1carne.gw.uiuc.edu (72.36.80.6)  0.873 ms  0.732 ms
 5  t-ur1carne.gw.uiuc.edu (72.36.80.2)  0.808 ms  0.670 ms  0.810 ms
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
25  * * *
26  * * *
27  * * *
28  * * *
29  * * *
30  * * *
```

Setting Static Routes

- Now, modify the routing tables to allow for node A to reach the IP addresses that you could not reach in step 6. Include a screenshot of the routing table of node A (10 points).

```
kumra@node-a:~$ sudo route add -net 192.168.2.0 netmask 255.255.255.0 gw 192.168.3.12 eth2
```

```
kumra@node-a:~$ 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	1024	0	0	eth0
172.16.0.0	0.0.0.0	255.240.0.0	U	0	0	0	eth0
172.16.0.1	0.0.0.0	255.255.255.255	UH	1024	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
192.168.2.0	192.168.3.12	255.255.255.0	UG	0	0	0	eth2
192.168.3.0	0.0.0.0	255.255.255.0	U	0	0	0	eth2

• Take a screenshot of node A successfully pinging and using traceroute on 192.168.2.12 (10 points).

```
kumra@node-a:~$ sudo route add -net 192.168.2.0 netmask 255.255.255.0 gw 192.168.3.12 eth2
```

```
kumra@node-a:~$ ping 192.168.2.12
```

PING 192.168.2.12 (192.168.2.12) 56(84) bytes of data:

64 bytes from 192.168.2.12: icmp_seq=1 ttl=64 time=0.533 ms

64 bytes from 192.168.2.12: icmp_seq=2 ttl=64 time=0.581 ms

64 bytes from 192.168.2.12: icmp_seq=3 ttl=64 time=0.612 ms

64 bytes from 192.168.2.12: icmp_seq=4 ttl=64 time=0.572 ms

64 bytes from 192.168.2.12: icmp_seq=5 ttl=64 time=0.568 ms

^C

--- 192.168.2.12 ping statistics ---

5 packets transmitted, 5 received, 0% packet loss, time 4005ms

rtt min/avg/max/mdev = 0.533/0.573/0.612/0.029 ms

```
kumra@node-a:~$ traceroute 192.168.2.12
```

traceroute to 192.168.2.12 (192.168.2.12), 30 hops max, 60 byte packets

1 node-c-link-2 (192.168.2.12) 0.685 ms 0.569 ms 0.445 ms

- Setup up more static route(s) so that every node can access every interface in the system. Take screenshots of routing tables of nodes B and C as well (10 points).

After adding all the possible static routes, such that every node can access every interface in the system, the following are the routing tables for A, B, and C.

– Node A

```
kumra@node-a:~$ 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      1024  0      0 eth0
172.16.0.0       0.0.0.0         255.240.0.0      U        0    0      0 eth0
172.16.0.1       0.0.0.0         255.255.255.255 UH      1024  0      0 eth0
192.168.1.0      0.0.0.0         255.255.255.0    U        0    0      0 eth1
192.168.2.0      192.168.1.11    255.255.255.0    UG       0    0      0 eth1
192.168.2.0      192.168.3.12    255.255.255.0    UG       0    0      0 eth2
192.168.3.0      0.0.0.0         255.255.255.0    U        0    0      0 eth2
kumra@node-a:~$
```

– Node B

```
kumra@node-b:~$ 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      1024  0      0 eth0
172.16.0.0       0.0.0.0         255.240.0.0      U        0    0      0 eth0
172.16.0.1       0.0.0.0         255.255.255.255 UH      1024  0      0 eth0
192.168.0.0      192.168.2.12    255.255.0.0      UG       0    0      0 eth1
192.168.1.0      0.0.0.0         255.255.255.0    U        0    0      0 eth2
192.168.2.0      0.0.0.0         255.255.255.0    U        0    0      0 eth1
192.168.3.0      192.168.2.12    255.255.255.0    UG       0    0      0 eth1
192.168.3.0      192.168.1.10    255.255.255.0    UG       0    0      0 eth2
192.168.3.8      192.168.1.10    255.255.255.252 UG       0    0      0 eth2
kumra@node-b:~$
```

– Node C

```
kumra@node-c:~$ 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      1024  0      0 eth0
172.16.0.0       0.0.0.0         255.240.0.0      U        0    0      0 eth0
172.16.0.1       0.0.0.0         255.255.255.255 UH      1024  0      0 eth0
192.168.1.0      192.168.3.10    255.255.255.0    UG       0    0      0 eth2
192.168.2.0      0.0.0.0         255.255.255.0    U        0    0      0 eth1
192.168.3.0      0.0.0.0         255.255.255.0    U        0    0      0 eth2
kumra@node-c:~$
```


- Show a traceroute from node B using traceroute on the four interfaces it does not own. Take a screenshot of each traceroute output. (10 points).

```
kumra@node-b:~$ traceroute 192.168.1.10
traceroute to 192.168.1.10 (192.168.1.10), 30 hops max, 60 byte packets
 1 node-a-link-1 (192.168.1.10)  0.690 ms  0.564 ms  0.718 ms
kumra@node-b:~$ traceroute 192.168.3.10
traceroute to 192.168.3.10 (192.168.3.10), 30 hops max, 60 byte packets
 1 node-a-link-0 (192.168.3.10)  0.709 ms  0.588 ms  0.460 ms
kumra@node-b:~$ traceroute 192.168.3.12
traceroute to 192.168.3.12 (192.168.3.12), 30 hops max, 60 byte packets
 1 node-c-link-0 (192.168.3.12)  0.779 ms  0.644 ms  0.723 ms
kumra@node-b:~$ traceroute 192.168.2.12
traceroute to 192.168.2.12 (192.168.2.12), 30 hops max, 60 byte packets
 1 node-c-link-2 (192.168.2.12)  0.618 ms  0.490 ms  0.691 ms
kumra@node-b:~$ █
```

Cleanup

Include a screenshot of your slice with no resources on the Manage Resources panel (see example below). (5 points)

Slice: Umra-lab6
Project: CPRE_489_S23

Slice expires in 6 days ✓
Project has no expiration ✓

Add ResourcesRenewUpdate SSH KeysTools

Manage Resources

acks initialized: no resources

View Respec

RenewRenew DateDeleteSSHRestartSnapshotDetailsAdd ResourcesExpand