Rajat Sethi

ECE 4380 – Computer Communications

Machine Problem 3

7/20/2021

Summary:

The goal of this lab was to create a network on GENI using OSPF routing. Once created, I was to explore the properties of OSPF routing through several commands, including but not limited to:

- vtysh
- show ip ospf neighbor
- show ip ospf route
- traceroute
- tcpdump
- configure terminal

I was also able to explore a little bit of Quagga, Zebra, and OSPFD and their configurations.

Implementation:

The implementation for this project was quite similar to "Machine Problem 2." Using GENI, I created a network on the Rutgers server with six bridges and six hosts. Through scp, I moved the routeconfig.sh file into each of the bridges and installed Quagga with its dependencies.

For the first question, I used grep with a RegEx to split the output into separate lines. Using grep and echo, I organized and split the lines even further until I had arrays for the ethernet interfaces, IP numbers, and IP subnets.

For the second question, I did the same thing as "Machine Problem 2." I logged into each bridge and used ifconfig to determine where each ethernet interface went.

For the third question, I took the same link up and down while testing the network using "neighbor" and "ping." I examined the dead time for the down link, and used ping to see how the routing changed almost instantly when the link fell.

For the fourth question, I used "configure terminal", "interface", and "ip ospf cost" commands to change the cost of a link. I then used traceroute and tcpdump to see what changes would happen to the network (especially the hosts that were using that route for the shortest path).

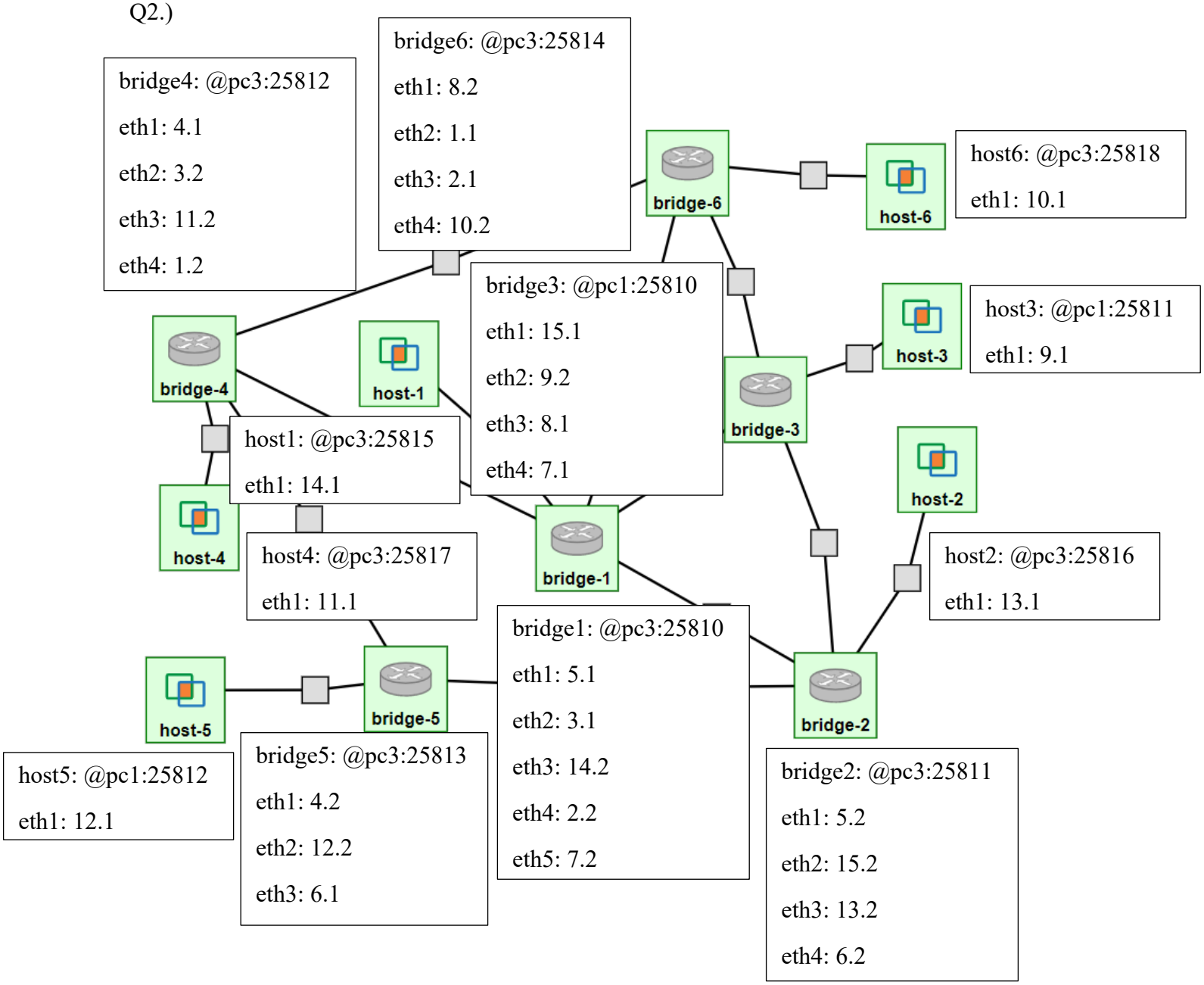
Questions:

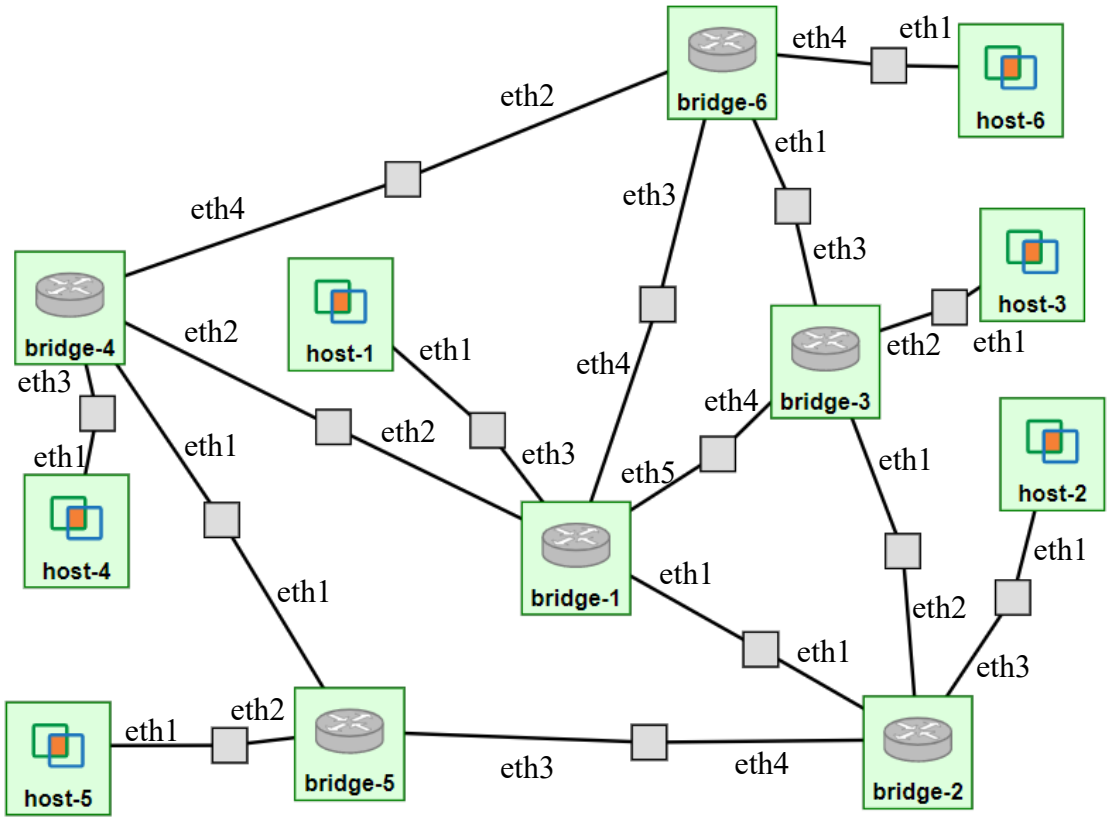
Q1.)

```
#!/bin/bash

ETHLINES=$(ip addr show | grep -E "eth[1-9][0-9]*$")

ETHNUMS=($(echo "${ETHLINES[*]}" | grep -Eo "eth[1-9][0-9]*"))

IPNUMS=($(echo "${ETHLINES[*]}" | grep -Eo "inet [0-9][0-9]*\.[0-9][0-9]*\.[0-9][0-9]*\.[0-9][0-9]*" | cut -c6-))

IPSUBS=($(echo "${IPNUMS[*]}" | grep -Eo "[0-9][0-9]*\.[0-9][0-9]*\.[0-9][0-9]*"))


for i in ${!ETHNUMS[@]} ; do

        echo "${ETHNUMS[$i]} has IP address ${IPNUMS[$i]} and connects to subnet ${IPSUBS[$i]}.0/24"

done
```

```
eth1 has IP address 10.10.5.1 and connects to subnet 10.10.5.0/24
eth2 has IP address 10.10.3.1 and connects to subnet 10.10.3.0/24
eth3 has IP address 10.10.14.2 and connects to subnet 10.10.14.0/24
eth4 has IP address 10.10.2.2 and connects to subnet 10.10.2.0/24
eth5 has IP address 10.10.7.2 and connects to subnet 10.10.7.0/24
```

Q2.)


bridge6: @pc3:25814
eth1: 8.2
eth2: 1.1
eth3: 2.1
eth4: 10.2
bridge4: @pc3:25812
eth1: 4.1
eth2: 3.2
eth3: 11.2
eth4: 1.2
bridge-6
host-6
host6: @pc3:25818
eth1: 10.1
bridge3: @pc1:25810
eth1: 15.1
eth2: 9.2
eth3: 8.1
eth4: 7.1
host-3
host3: @pc1:25811
eth1: 9.1
bridge-4
host-1
bridge-3
host1: @pc3:25815
eth1: 14.1
host-2
host-4
host4: @pc3:25817
eth1: 11.1
bridge-1
host2: @pc3:25816
eth1: 13.1
bridge1: @pc3:25810
eth1: 5.1
eth2: 3.1
eth3: 14.2
eth4: 2.2
eth5: 7.2
host-5
bridge-5
bridge-2
host5: @pc1:25812
eth1: 12.1
bridge5: @pc3:25813
eth1: 4.2
eth2: 12.2
eth3: 6.1
bridge2: @pc3:25811
eth1: 5.2
eth2: 15.2
eth3: 13.2
eth4: 6.2



Q3.)

Taking down the link between bridge-1 and bridge-4 (Both are eth2)

```
Neighbor ID     Pri State           Dead Time Address         Interface            RXmtL RqstL DBsmL
172.17.3.12       1 Full/DR           33.936s 10.10.4.2       eth1:10.10.4.1           0     0     0
172.17.3.9        1 Full/Backup        1.753s 10.10.3.1       eth2:10.10.3.2           1     0     0
172.17.3.13       1 Full/DR           30.266s 10.10.1.1       eth4:10.10.1.2           0     0     0
bridge-4.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu# show ip ospf neighbor

Neighbor ID     Pri State           Dead Time Address         Interface            RXmtL RqstL DBsmL
172.17.3.12       1 Full/DR           33.316s 10.10.4.2       eth1:10.10.4.1           0     0     0
172.17.3.9        1 Full/Backup        1.133s 10.10.3.1       eth2:10.10.3.2           1     0     0
172.17.3.13       1 Full/DR           39.646s 10.10.1.1       eth4:10.10.1.2           0     0     0
bridge-4.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu# show ip ospf neighbor

Neighbor ID     Pri State           Dead Time Address         Interface            RXmtL RqstL DBsmL
172.17.3.12       1 Full/DR           32.686s 10.10.4.2       eth1:10.10.4.1           0     0     0
172.17.3.9        1 Full/Backup        0.502s 10.10.3.1       eth2:10.10.3.2           1     0     0
172.17.3.13       1 Full/DR           39.016s 10.10.1.1       eth4:10.10.1.2           0     0     0
bridge-4.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu# show ip ospf neighbor

Neighbor ID     Pri State           Dead Time Address         Interface            RXmtL RqstL DBsmL
172.17.3.12       1 Full/DR           32.066s 10.10.4.2       eth1:10.10.4.1           2     0     0
172.17.3.13       1 Full/DR           39.968s 10.10.1.1       eth4:10.10.1.2           0     0     0
```

Sending pings from host-4 to host-1

(Traceroute while the link is up)

```
sethi@host-4:~$ traceroute 10.10.14.1
traceroute to 10.10.14.1 (10.10.14.1), 30 hops max, 60 byte packets
 1  bridge-4-link-10 (10.10.11.2)  0.475 ms  0.394 ms  0.349 ms
 2  bridge-1-link-2 (10.10.3.1)  0.924 ms  0.860 ms  0.813 ms
 3  host-1-link-13 (10.10.14.1)  1.184 ms  1.139 ms  1.029 ms
```

(Traceroute while the link is down)

```
sethi@host-4:~$ traceroute 10.10.14.1
traceroute to 10.10.14.1 (10.10.14.1), 30 hops max, 60 byte packets
 1  bridge-4-link-10 (10.10.11.2)  0.324 ms  0.293 ms  0.353 ms
 2  bridge-6-link-0 (10.10.1.1)  0.845 ms  0.796 ms  0.858 ms
 3  bridge-1-link-1 (10.10.2.2)  1.175 ms  1.131 ms  1.298 ms
 4  host-1-link-13 (10.10.14.1)  1.595 ms  1.510 ms  1.424 ms
```

(Change in pings run at 0.05 seconds)

```
64 bytes from 10.10.14.1: icmp_seq=68 ttl=62 time=1.25 ms
64 bytes from 10.10.14.1: icmp_seq=69 ttl=62 time=1.22 ms
64 bytes from 10.10.14.1: icmp_seq=187 ttl=61 time=1.33 ms
64 bytes from 10.10.14.1: icmp_seq=188 ttl=61 time=1.19 ms
```

Running the experiment several times, I found that the route would be recalculated in a negligible amount of time (<0.1 seconds). However, if a ping got “stuck” by chance (as no acknowledgement was received), then it would take around 5-10 seconds to fully recalculate.

Q4.)

Change in cost to link between bridge-1 and bridge-4 (Applied on both interfaces).

```
bridge-1.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu# configure terminal
bridge-1.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu(config)# interface eth2
bridge-1.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu(config-if)# ip ospf cost 50
bridge-1.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu(config-if)# exit
bridge-1.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu(config)# exit
bridge-1.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu# show ip ospf interface brief
No such interface name
bridge-1.rajat-mp3-1.ch-geni-net.instageni.rutgers.edu# show ip ospf interface eth2
eth2 is up
  ifindex 4, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.10.3.1/24, Broadcast 10.10.3.255, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 172.17.3.9, Network Type BROADCAST, Cost: 50
  Transmit Delay is 1 sec, State Backup, Priority 1
  Designated Router (ID) 172.17.3.11, Interface Address 10.10.3.2
  Backup Designated Router (ID) 172.17.3.9, Interface Address 10.10.3.1
  Saved Network-LSA sequence number 0x80000020
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 9.210s
  Neighbor Count is 1, Adjacent neighbor count is 1
```

Effects to path when the cost is changed.

```
sethi@host-4:~$ traceroute 10.10.14.1
traceroute to 10.10.14.1 (10.10.14.1), 30 hops max, 60 byte packets
 1  bridge-4-link-10 (10.10.11.2)  0.541 ms  0.463 ms  0.467 ms
 2  bridge-6-link-0 (10.10.1.1)  0.999 ms  0.940 ms  0.892 ms
 3  bridge-1-link-1 (10.10.2.2)  1.461 ms  1.414 ms  1.366 ms
 4  host-1-link-13 (10.10.14.1)  1.779 ms  1.707 ms  1.636 ms
```

tcpdump of bridge-5 (separate from cost change), still receiving LSP packets.

```
sethi@bridge-5:~$ sudo tcpdump -i eth1 ip -vv
tcpdump: listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
07:17:15.803533 IP (tos 0xc0, ttl 1, id 14953, offset 0, flags [none], proto OSPF (89), length 68)
    bridge-5-link-3 > ospf-all.mcast.net: OSPFv2, Hello, length 48
 Router-ID pcvm3-12.instageni.rutgers.edu, Backbone Area, Authentication Type: none (0)
 Options [External]
   Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
   Designated Router bridge-5-link-3, Backup Designated Router bridge-4-link-3
   Neighbor List:
     pcvm3-11.instageni.rutgers.edu
07:17:15.803703 IP (tos 0xc0, ttl 1, id 39479, offset 0, flags [none], proto OSPF (89), length 68)
    bridge-4-link-3 > ospf-all.mcast.net: OSPFv2, Hello, length 48
 Router-ID pcvm3-11.instageni.rutgers.edu, Backbone Area, Authentication Type: none (0)
 Options [External]
   Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
   Designated Router bridge-5-link-3, Backup Designated Router bridge-4-link-3
   Neighbor List:
     pcvm3-12.instageni.rutgers.edu
```

Conclusion:

Overall, I learned about the OSPF protocol and the respective commands to configure the network. Surprisingly, I found the hardest part of this lab to be the bash script. I used bash very sparsely in my prior classes, so working with arrays was a little tougher than I expected. The other difficulty I had was with the pinging when taking down a link. At first, I did not think I was doing the question correctly, as the “ttl” and “icmp_seq” parameters were not changing. I kept changing commands and hosts until I finally got everything to work as expected, which was a good test of patience.