03/12/2023

PROJECT 4

1. TASK 1

Create Topology with 3 routers with 2 hosts per router/LAN.

A. LAN A

   i. LAN A with 50 hosts.
   ii. Network Address 20.10.172.128/26
   iii. Subnet Mask 255.255.255.192
   iv. Smallest IP Address 20.10.172.129/26
   v. Highest IP Address 20.10.172.190/26
   vi. Broadcast IP Address 20.10.172.191/26

B. LAN B

   i. LAN B with 75 hosts.
   ii. Network Address 20.10.172.0/25
   iii. Subnet Mask 255.255.255.128
   iv. Smallest IP Address 20.10.172.1/25
   v. Highest IP Address 20.10.172.126/25
   vi. Broadcast IP Address 20.10.172.127/25

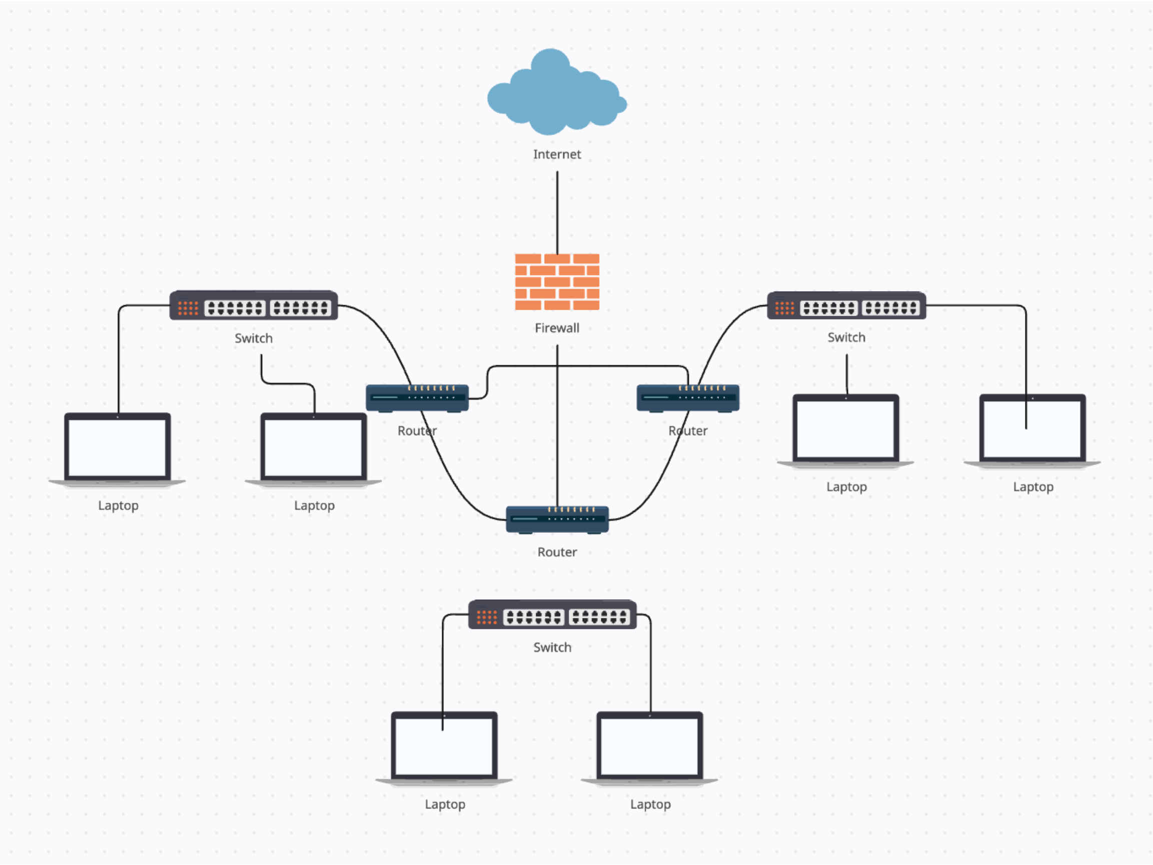
C. LAN C

   i. LAN C with 20
   ii. Network Address 20.10.172.0/27
   iii. Subnet Mask 255.255.255.224
   iv. Smallest IP Address 20.10.172.193/27
   v. Highest IP Address 20.10.172.222/27
   vi. Broadcast IP Address 20.10.172.223/27

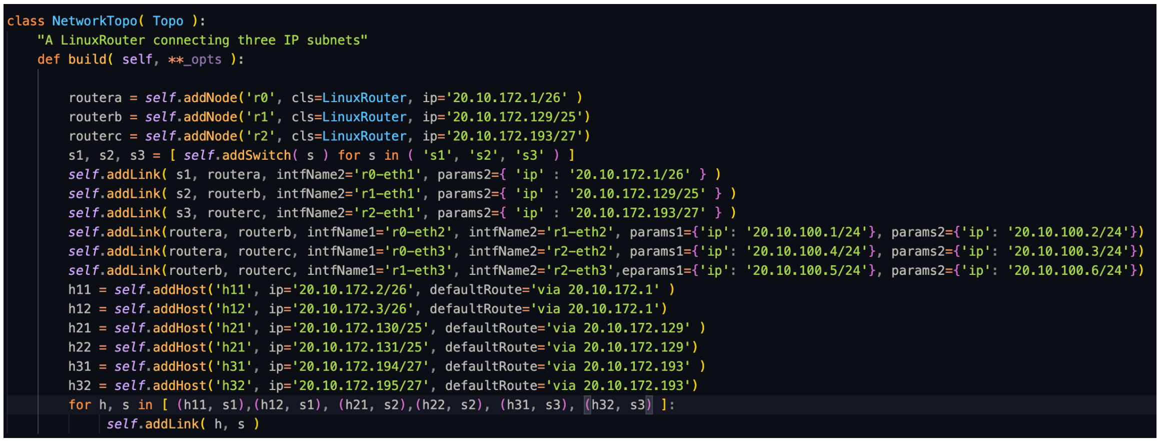
NOTE THAT THE CALCULATIONS FOR SUBNET WERE DONE FROM THE BIGGEST SUBNET REQUIREMENT TO THE SMALLEST.

2. TASK 2

A. Creating Topology and configuring the Network Interface



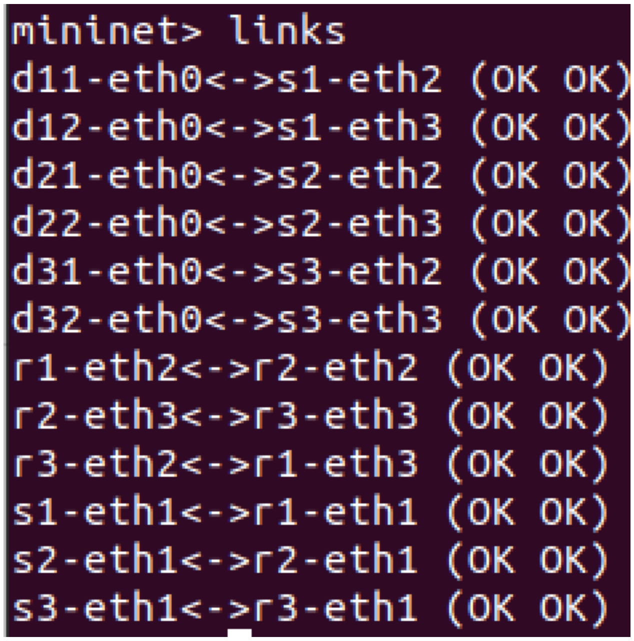
B. Python code implementation of the Topology



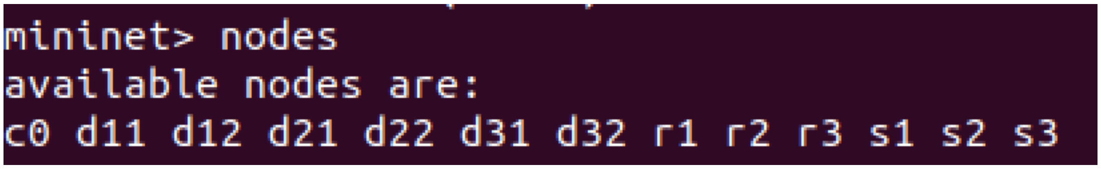
```python
class NetworkTopo( Topo ):
    "A LinuxRouter connecting three IP subnets"
    def build( self, **_opts ):

        routera = self.addNode('r0', cls=LinuxRouter, ip='20.10.172.1/26' )
        routerb = self.addNode('r1', cls=LinuxRouter, ip='20.10.172.129/25')
        routerc = self.addNode('r2', cls=LinuxRouter, ip='20.10.172.193/27')
        s1, s2, s3 = [ self.addSwitch( s ) for s in ( 's1', 's2', 's3' ) ]
        self.addLink( s1, routera, intfName2='r0-eth1', params2={ 'ip' : '20.10.172.1/26' } )
        self.addLink( s2, routerb, intfName2='r1-eth1', params2={ 'ip' : '20.10.172.129/25' } )
        self.addLink( s3, routerc, intfName2='r2-eth1', params2={ 'ip' : '20.10.172.193/27' } )
        self.addLink(routera, routerb, intfName1='r0-eth2', intfName2='r1-eth2', params1={'ip': '20.10.100.1/24'}, params2={'ip': '20.10.100.2/24'})
        self.addLink(routera, routerc, intfName1='r0-eth3', intfName2='r2-eth2', params1={'ip': '20.10.100.4/24'}, params2={'ip': '20.10.100.3/24'})
        self.addLink(routerb, routerc, intfName1='r1-eth3', intfName2='r2-eth3',eparams1={'ip': '20.10.100.5/24'}, params2={'ip': '20.10.100.6/24'})
        h11 = self.addHost('h11', ip='20.10.172.2/26', defaultRoute='via 20.10.172.1' )
        h12 = self.addHost('h12', ip='20.10.172.3/26', defaultRoute='via 20.10.172.1')
        h21 = self.addHost('h21', ip='20.10.172.130/25', defaultRoute='via 20.10.172.129' )
        h22 = self.addHost('h21', ip='20.10.172.131/25', defaultRoute='via 20.10.172.129')
        h31 = self.addHost('h31', ip='20.10.172.194/27', defaultRoute='via 20.10.172.193' )
        h32 = self.addHost('h32', ip='20.10.172.195/27', defaultRoute='via 20.10.172.193')
        for h, s in [ (h11, s1),(h12, s1), (h21, s2),(h22, s2), (h31, s3), (h32, s3) ]:
            self.addLink( h, s )
```

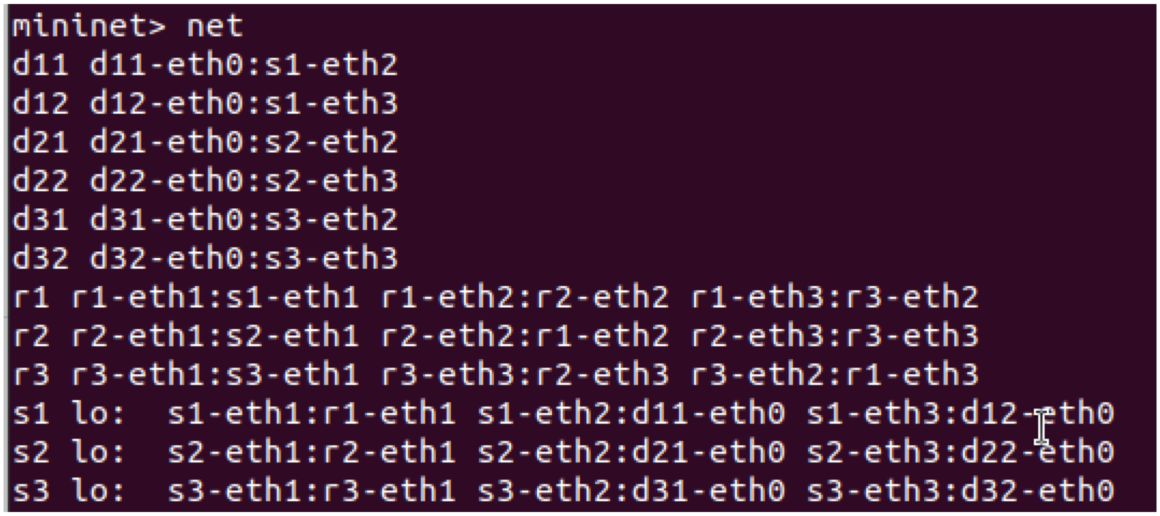
C. Mininet outputs proving the links, nodes and net



```
mininet> links
d11-eth0<->s1-eth2 (OK OK)
d12-eth0<->s1-eth3 (OK OK)
d21-eth0<->s2-eth2 (OK OK)
d22-eth0<->s2-eth3 (OK OK)
d31-eth0<->s3-eth2 (OK OK)
d32-eth0<->s3-eth3 (OK OK)
r1-eth2<->r2-eth2 (OK OK)
r2-eth3<->r3-eth3 (OK OK)
r3-eth2<->r1-eth3 (OK OK)
s1-eth1<->r1-eth1 (OK OK)
s2-eth1<->r2-eth1 (OK OK)
s3-eth1<->r3-eth1 (OK OK)
```



```
mininet> nodes
available nodes are:
c0 d11 d12 d21 d22 d31 d32 r1 r2 r3 s1 s2 s3
```

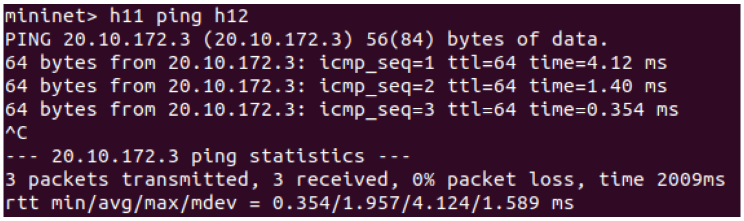


```
mininet> net
d11 d11-eth0:s1-eth2
d12 d12-eth0:s1-eth3
d21 d21-eth0:s2-eth2
d22 d22-eth0:s2-eth3
d31 d31-eth0:s3-eth2
d32 d32-eth0:s3-eth3
r1 r1-eth1:s1-eth1 r1-eth2:r2-eth2 r1-eth3:r3-eth2
r2 r2-eth1:s2-eth1 r2-eth2:r1-eth2 r2-eth3:r3-eth3
r3 r3-eth1:s3-eth1 r3-eth3:r2-eth3 r3-eth2:r1-eth3
s1 lo:  s1-eth1:r1-eth1 s1-eth2:d11-eth0 s1-eth3:d12-eth0
s2 lo:  s2-eth1:r2-eth1 s2-eth2:d21-eth0 s2-eth3:d22-eth0
s3 lo:  s3-eth1:r3-eth1 s3-eth2:d31-eth0 s3-eth3:d32-eth0
```

D. Testing Local LAN ping

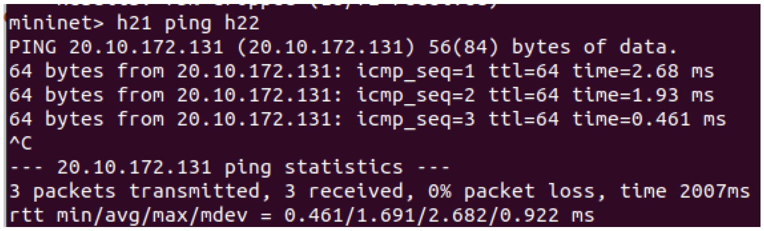
i. LAN A

1.



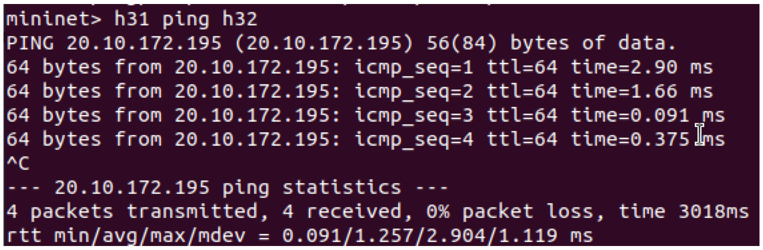
```
mininet> h11 ping h12
PING 20.10.172.3 (20.10.172.3) 56(84) bytes of data.
64 bytes from 20.10.172.3: icmp_seq=1 ttl=64 time=4.12 ms
64 bytes from 20.10.172.3: icmp_seq=2 ttl=64 time=1.40 ms
64 bytes from 20.10.172.3: icmp_seq=3 ttl=64 time=0.354 ms
^C
--- 20.10.172.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2009ms
rtt min/avg/max/mdev = 0.354/1.957/4.124/1.589 ms
```

ii. LAN B

1. 

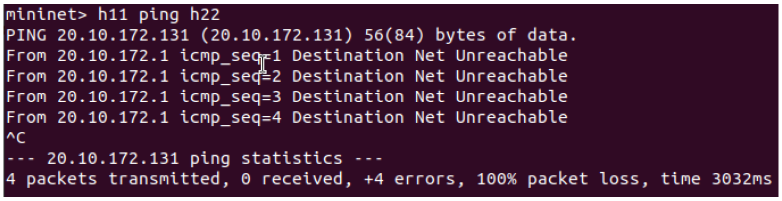
```
mininet> h21 ping h22
PING 20.10.172.131 (20.10.172.131) 56(84) bytes of data.
64 bytes from 20.10.172.131: icmp_seq=1 ttl=64 time=2.68 ms
64 bytes from 20.10.172.131: icmp_seq=2 ttl=64 time=1.93 ms
64 bytes from 20.10.172.131: icmp_seq=3 ttl=64 time=0.461 ms
^C
--- 20.10.172.131 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2007ms
rtt min/avg/max/mdev = 0.461/1.691/2.682/0.922 ms
```

iii. LAN C

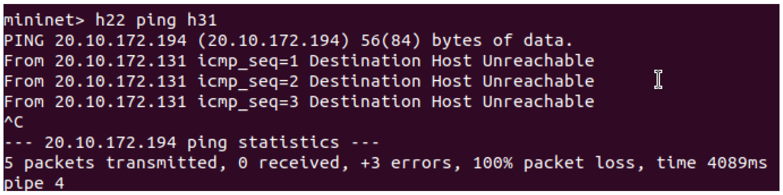
1. 

```
mininet> h31 ping h32
PING 20.10.172.195 (20.10.172.195) 56(84) bytes of data.
64 bytes from 20.10.172.195: icmp_seq=1 ttl=64 time=2.90 ms
64 bytes from 20.10.172.195: icmp_seq=2 ttl=64 time=1.66 ms
64 bytes from 20.10.172.195: icmp_seq=3 ttl=64 time=0.091 ms
64 bytes from 20.10.172.195: icmp_seq=4 ttl=64 time=0.375 ms
^C
--- 20.10.172.195 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3018ms
rtt min/avg/max/mdev = 0.091/1.257/2.904/1.119 ms
```

iv. Pingall Command would fail as there are no links between the routers themselves.

1. 

```
mininet> h11 ping h22
PING 20.10.172.131 (20.10.172.131) 56(84) bytes of data.
From 20.10.172.1 icmp_seq=1 Destination Net Unreachable
From 20.10.172.1 icmp_seq=2 Destination Net Unreachable
From 20.10.172.1 icmp_seq=3 Destination Net Unreachable
From 20.10.172.1 icmp_seq=4 Destination Net Unreachable
^C
--- 20.10.172.131 ping statistics ---
4 packets transmitted, 0 received, +4 errors, 100% packet loss, time 3032ms
```

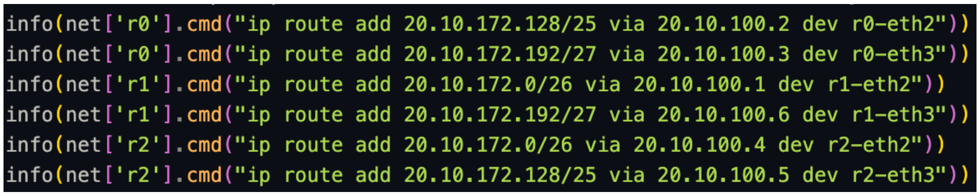


```
mininet> h22 ping h31
PING 20.10.172.194 (20.10.172.194) 56(84) bytes of data.
From 20.10.172.131 icmp_seq=1 Destination Host Unreachable
From 20.10.172.131 icmp_seq=2 Destination Host Unreachable
From 20.10.172.131 icmp_seq=3 Destination Host Unreachable
^C
--- 20.10.172.194 ping statistics ---
5 packets transmitted, 0 received, +3 errors, 100% packet loss, time 4089ms
pipe 4
```

3. TASK 3

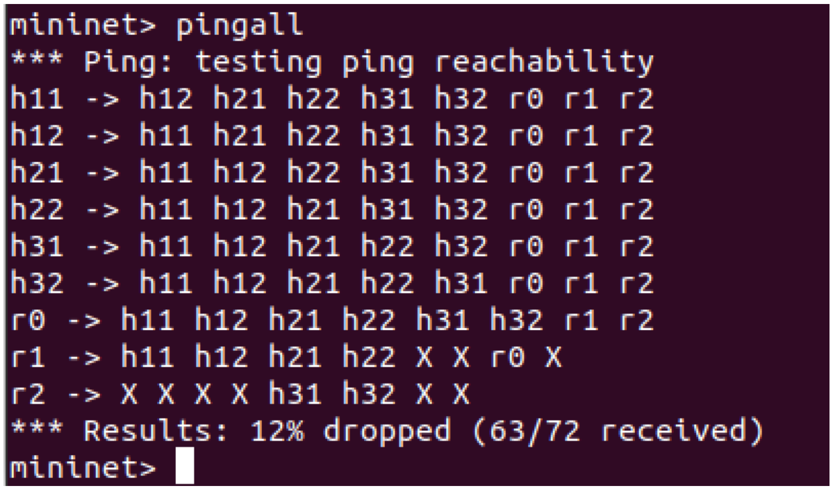
A. Add routing rules on each host for destination

i. Python Implementation for route add –

1. 

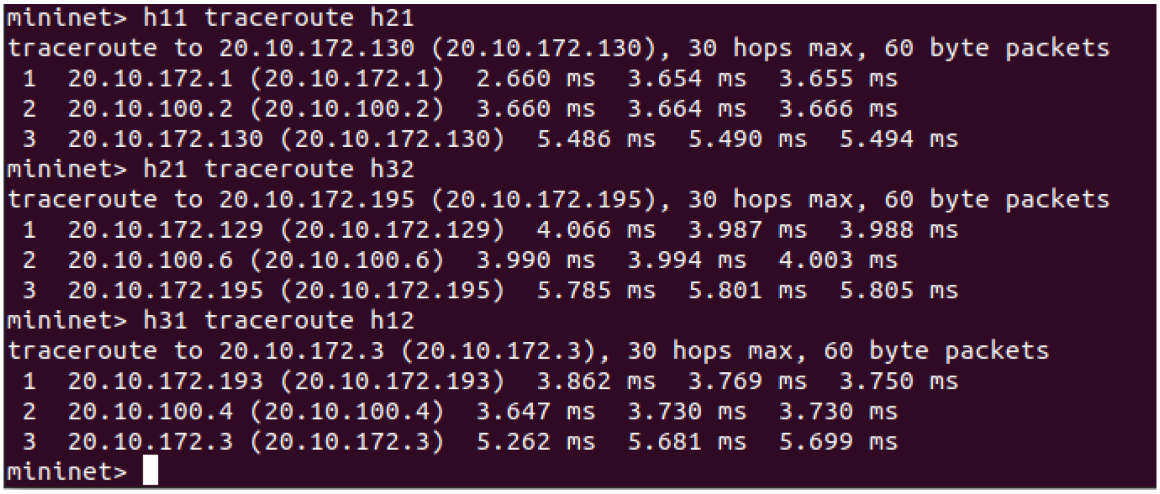
```
info(net['r0'].cmd("ip route add 20.10.172.128/25 via 20.10.100.2 dev r0-eth2"))
info(net['r0'].cmd("ip route add 20.10.172.192/27 via 20.10.100.3 dev r0-eth3"))
info(net['r1'].cmd("ip route add 20.10.172.0/26 via 20.10.100.1 dev r1-eth2"))
info(net['r1'].cmd("ip route add 20.10.172.192/27 via 20.10.100.6 dev r1-eth3"))
info(net['r2'].cmd("ip route add 20.10.172.0/26 via 20.10.100.4 dev r2-eth2"))
info(net['r2'].cmd("ip route add 20.10.172.128/25 via 20.10.100.5 dev r2-eth3"))
```

ii. Running pingall command to test the implementation of the route add –

1. 

```
mininet> pingall
*** Ping: testing ping reachability
h11 -> h12 h21 h22 h31 h32 r0 r1 r2
h12 -> h11 h21 h22 h31 h32 r0 r1 r2
h21 -> h11 h12 h22 h31 h32 r0 r1 r2
h22 -> h11 h12 h21 h31 h32 r0 r1 r2
h31 -> h11 h12 h21 h22 h32 r0 r1 r2
h32 -> h11 h12 h21 h22 h31 r0 r1 r2
r0 -> h11 h12 h21 h22 h31 h32 r1 r2
r1 -> h11 h12 h21 h22 X X r0 X
r2 -> X X X X h31 h32 X X
*** Results: 12% dropped (63/72 received)
mininet> 
```

iii. Running traceroute between LANs

1. 

```
mininet> h11 traceroute h21
traceroute to 20.10.172.130 (20.10.172.130), 30 hops max, 60 byte packets
 1  20.10.172.1 (20.10.172.1)  2.660 ms  3.654 ms  3.655 ms
 2  20.10.100.2 (20.10.100.2)  3.660 ms  3.664 ms  3.666 ms
 3  20.10.172.130 (20.10.172.130)  5.486 ms  5.490 ms  5.494 ms
mininet> h21 traceroute h32
traceroute to 20.10.172.195 (20.10.172.195), 30 hops max, 60 byte packets
 1  20.10.172.129 (20.10.172.129)  4.066 ms  3.987 ms  3.988 ms
 2  20.10.100.6 (20.10.100.6)  3.990 ms  3.994 ms  4.003 ms
 3  20.10.172.195 (20.10.172.195)  5.785 ms  5.801 ms  5.805 ms
mininet> h31 traceroute h12
traceroute to 20.10.172.3 (20.10.172.3), 30 hops max, 60 byte packets
 1  20.10.172.193 (20.10.172.193)  3.862 ms  3.769 ms  3.750 ms
 2  20.10.100.4 (20.10.100.4)  3.647 ms  3.730 ms  3.730 ms
 3  20.10.172.3 (20.10.172.3)  5.262 ms  5.681 ms  5.699 ms
mininet> 
```

4. SOURCES AND TOOLS USED

A. Mininet implementation of code referred from https://github.com/mininet/mininet/blob/master/examples/linuxrouter.py

B. Creately used for pictorial representation of the Topology.

C. Ubuntu OS running on UTM, a MAC application for stable Virtualization on M1 macs.