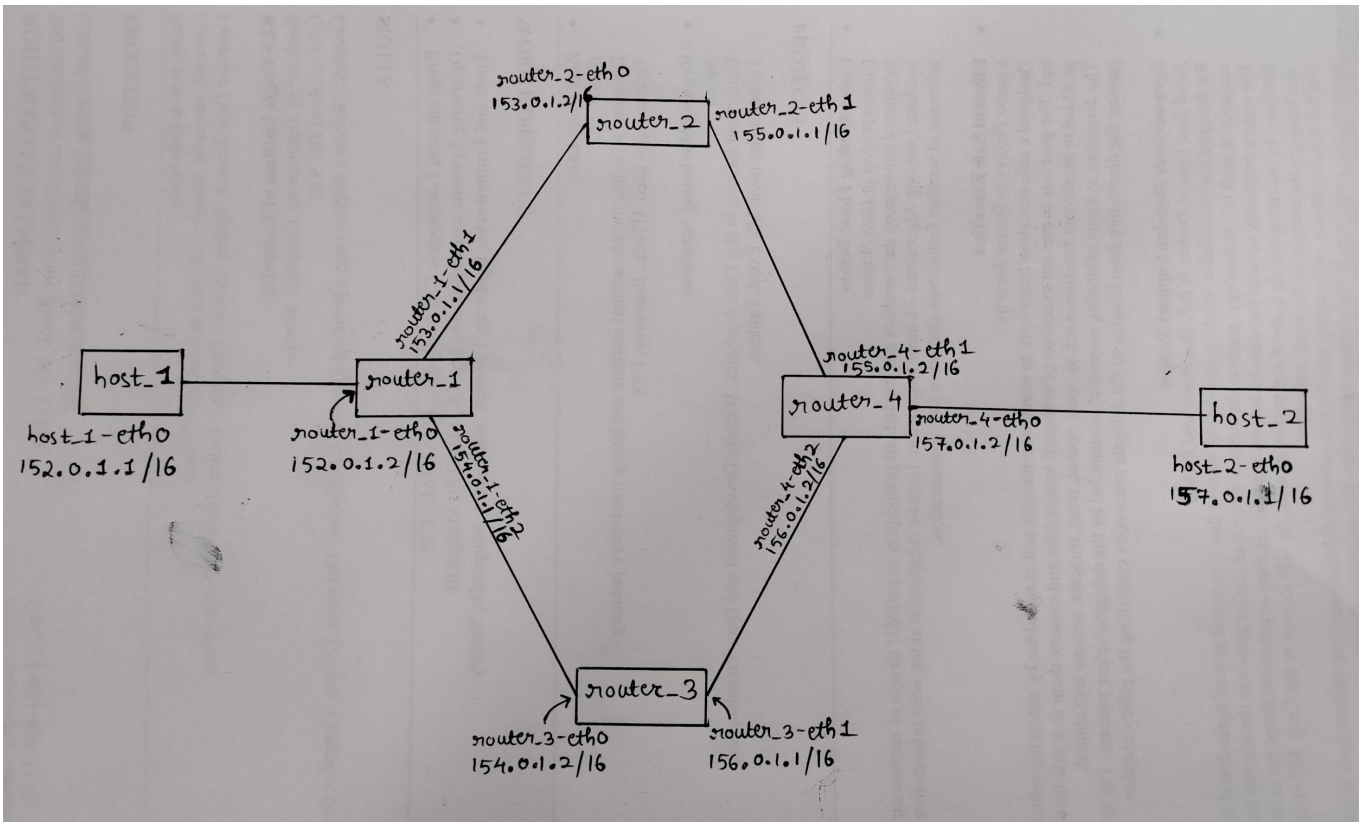
Name: Rohit Inamdar
SBU ID: 114504643

CSE 534: FCN - Assignment 3, Part A

Tasks:

1. Static Topology:

The static topology is as shown below



Each node in the topology is configured with the respective IP address by the following lines of code:

```
def build(self, **_opts):
    router_1 = self.addNode('router_1', cls=LinuxRouter, ip='152.0.1.2/16')
    router_2 = self.addNode('router_2', cls=LinuxRouter, ip='153.0.1.2/16')
    router_3 = self.addNode('router_3', cls=LinuxRouter, ip='154.0.1.2/16')
    router_4 = self.addNode('router_4', cls=LinuxRouter, ip='157.0.1.2/16')

    host_1 = self.addHost('host_1', ip='152.0.1.1/16')
    host_2 = self.addHost('host_2', ip='157.0.1.1/16')
```

The routers have multiple ports i.e IP addresses which is specified by the code snippet:

```
def routerCMD(router, IPAddr, port): # function to assign Ip address to each node in the topology
    mnet.get(router).cmd('ip addr add ' + IPAddr + ' dev ' + port)
```

```
routerCMD('router_1', '153.0.1.1/16', 'router_1-eth1')
routerCMD('router_1', '154.0.1.1/16', 'router_1-eth2')
routerCMD('router_2', '155.0.1.1/16', 'router_2-eth1')
routerCMD('router_3', '156.0.1.1/16', 'router_3-eth1')
routerCMD('router_4', '155.0.1.2/16', 'router_4-eth1')
routerCMD('router_4', '156.0.1.2/16', 'router_4-eth2')
```

Initially, the routing table before configuring the static routes is as shown below:

```
 host_2-eth0:router_4-eth0
_1 router_1-eth0:host_1-eth0 router_1-eth1:router_2-eth0 router_1-eth2:router_3-eth0
_2 router_2-eth0:router_1-eth1 router_2-eth1:router_4-eth1
_3 router_3-eth0:router_1-eth2 router_3-eth1:router_4-eth2
_4 router_4-eth0:host_2-eth0 router_4-eth1:router_2-eth1 router_4-eth2:router_3-eth1
uting Table on host_1
 IP routing table
ation     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0       0.0.0.0         255.255.0.0     U     0      0        0 host_1-eth0
uting Table on router_1
 IP routing table
ation     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_1-eth0
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_1-eth1
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_1-eth2
uting Table on router_2
 IP routing table
ation     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_2-eth0
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_2-eth1
uting Table on router_3
 IP routing table
ation     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_3-eth0
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_3-eth1
uting Table on router_4
 IP routing table
ation     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_4-eth1
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_4-eth2
0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_4-eth0
uting Table on host_2
 IP routing table
ation     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0       0.0.0.0         255.255.0.0     U     0      0        0 host_2-eth0
arting CLI:
t> _
```

We configure the static routes by using the code snippet below:

```
def nodeDefaultPath(router, IPAddr, port): # function for configuring static routes to nodes
    mnet.get(router).cmd('ip route add default via ' + IPAddr + ' dev ' + port)
```

```
nodeDefaultPath('host_1', '152.0.1.2', 'host_1-eth0')
nodeDefaultPath('host_2', '157.0.1.2', 'host_2-eth0')

mnet.get('router_1').cmd('ip route add 156.0.0.0/16 via 154.0.1.2 dev router_1-eth2')
nodeDefaultPath('router_1', '153.0.1.2', 'router_1-eth1')

mnet.get('router_2').cmd('ip route add 152.0.0.0/16 via 153.0.1.1 dev router_2-eth0')
mnet.get('router_2').cmd('ip route add 154.0.0.0/16 via 153.0.1.1 dev router_2-eth0')
nodeDefaultPath('router_2', '155.0.1.2', 'router_2-eth1')

mnet.get('router_3').cmd('ip route add 152.0.0.0/16 via 154.0.1.1 dev router_3-eth0')
mnet.get('router_3').cmd('ip route add 153.0.0.0/16 via 154.0.1.1 dev router_3-eth0')
nodeDefaultPath('router_3', '156.0.1.2', 'router_3-eth1')

mnet.get('router_4').cmd('ip route add 154.0.0.0/16 via 156.0.1.1 dev router_4-eth2')
nodeDefaultPath('router_4', '155.0.1.1', 'router_4-eth1')
```

Now the routing table will be as follows:

```
*** Routing Table on host_1
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         152.0.1.2       0.0.0.0         UG    0      0        0 host_1-eth0
152.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 host_1-eth0
*** Routing Table on router_1
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         153.0.1.2       0.0.0.0         UG    0      0        0 router_1-eth1
152.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_1-eth0
153.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_1-eth1
154.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_1-eth2
156.0.0.0       154.0.1.2       255.255.0.0     UG    0      0        0 router_1-eth2
*** Routing Table on router_2
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         155.0.1.2       0.0.0.0         UG    0      0        0 router_2-eth1
152.0.0.0       153.0.1.1       255.255.0.0     UG    0      0        0 router_2-eth0
153.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_2-eth0
154.0.0.0       153.0.1.1       255.255.0.0     UG    0      0        0 router_2-eth0
155.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_2-eth1
*** Routing Table on router_3
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         156.0.1.2       0.0.0.0         UG    0      0        0 router_3-eth1
152.0.0.0       154.0.1.1       255.255.0.0     UG    0      0        0 router_3-eth0
153.0.0.0       154.0.1.1       255.255.0.0     UG    0      0        0 router_3-eth0
154.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_3-eth0
156.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_3-eth1
*** Routing Table on router_4
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         155.0.1.1       0.0.0.0         UG    0      0        0 router_4-eth1
154.0.0.0       156.0.1.1       255.255.0.0     UG    0      0        0 router_4-eth2
155.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_4-eth1
156.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_4-eth2
157.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 router_4-eth0
*** Routing Table on host_2
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         157.0.1.2       0.0.0.0         UG    0      0        0 host_2-eth0
157.0.0.0       0.0.0.0         255.255.0.0     U     0      0        0 host_2-eth0
*** Starting CLI:
mininet> _
```

From the image above we can see that the Gateway column has now been updated with the necessary IP address.

The trace from host 1 to host 2 and vice versa is shown below:

```
mininet> host_1 traceroute host_2
traceroute to 157.0.1.1 (157.0.1.1), 30 hops max, 60 byte packets
 1  152.0.1.2 (152.0.1.2)  0.037 ms  0.006 ms  0.004 ms
 2  153.0.1.2 (153.0.1.2)  0.022 ms  0.007 ms  0.007 ms
 3  155.0.1.2 (155.0.1.2)  0.022 ms  0.011 ms  0.011 ms
 4  157.0.1.1 (157.0.1.1)  0.031 ms  0.035 ms  0.010 ms
mininet> host_2 traceroute host_1
traceroute to 152.0.1.1 (152.0.1.1), 30 hops max, 60 byte packets
 1  157.0.1.2 (157.0.1.2)  0.021 ms  0.005 ms  0.004 ms
 2  155.0.1.1 (155.0.1.1)  0.010 ms  0.005 ms  0.005 ms
 3  153.0.1.1 (153.0.1.1)  0.010 ms  0.006 ms  0.007 ms
 4  152.0.1.1 (152.0.1.1)  0.012 ms  0.031 ms  0.009 ms
mininet>
```