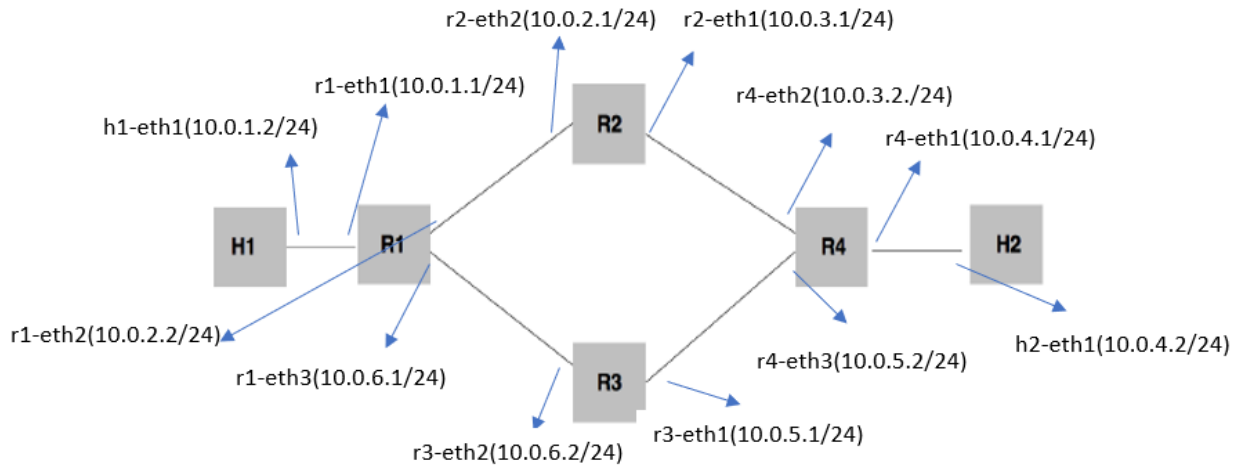
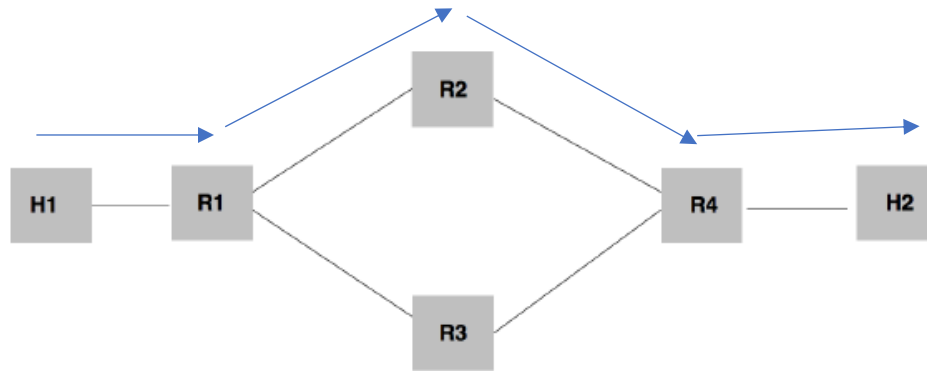


# 1. Network Topology with with IP addresses (including subnets)



# 2. Trace route between h1 and h2



# 3. Router tables at each node

```
mininet> h1 route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
default          10.0.1.1       0.0.0.0         UG    0      0        0 h1-eth0
10.0.1.0         0.0.0.0        255.255.255.0   U      0      0        0 h1-eth0
mininet>
```

```
mininet> r1 route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.1.0         0.0.0.0        255.255.255.0   U        0      0      0 r1-eth1
10.0.2.0         0.0.0.0        255.255.255.0   U        0      0      0 r1-eth2
10.0.3.0         10.0.2.1       255.255.255.0   UG       0      0      0 r1-eth2
10.0.4.0         10.0.2.1       255.255.255.0   UG       0      0      0 r1-eth2
10.0.5.0         10.0.6.2       255.255.255.0   UG       0      0      0 r1-eth3
10.0.6.0         0.0.0.0        255.255.255.0   U        0      0      0 r1-eth3
mininet>
```

```
mininet> r2 route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.1.0         10.0.2.2       255.255.255.0   UG       0      0      0 r2-eth2
10.0.2.0         0.0.0.0        255.255.255.0   U        0      0      0 r2-eth2
10.0.3.0         0.0.0.0        255.255.255.0   U        0      0      0 r2-eth1
10.0.4.0         10.0.3.2       255.255.255.0   UG       0      0      0 r2-eth1
10.0.5.0         10.0.3.2       255.255.255.0   UG       0      0      0 r2-eth1
10.0.6.0         10.0.2.2       255.255.255.0   UG       0      0      0 r2-eth2
mininet>
```

```
mininet> r3 route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.1.0         10.0.6.1       255.255.255.0   UG       0      0      0 r3-eth2
10.0.2.0         10.0.6.1       255.255.255.0   UG       0      0      0 r3-eth2
10.0.3.0         10.0.5.2       255.255.255.0   UG       0      0      0 r3-eth1
10.0.4.0         10.0.5.2       255.255.255.0   UG       0      0      0 r3-eth1
10.0.5.0         0.0.0.0        255.255.255.0   U        0      0      0 r3-eth1
10.0.6.0         0.0.0.0        255.255.255.0   U        0      0      0 r3-eth2
mininet>
```

```
mininet> r4 route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.1.0         10.0.3.1       255.255.255.0   UG       0      0      0 r4-eth2
10.0.2.0         10.0.3.1       255.255.255.0   UG       0      0      0 r4-eth2
10.0.3.0         0.0.0.0        255.255.255.0   U        0      0      0 r4-eth2
10.0.4.0         0.0.0.0        255.255.255.0   U        0      0      0 r4-eth1
10.0.5.0         0.0.0.0        255.255.255.0   U        0      0      0 r4-eth3
10.0.6.0         10.0.5.1       255.255.255.0   UG       0      0      0 r4-eth3
mininet>
```

```
mininet> h2 route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
default         10.0.4.1       0.0.0.0         UG       0      0      0 h2-eth0
10.0.4.0         0.0.0.0        255.255.255.0   U        0      0      0 h2-eth0
mininet>
```

#### 4. Traceroute output between h1 and h2

```
mininet> h1 traceroute h2
traceroute to 10.0.4.2 (10.0.4.2), 30 hops max, 60 byte packets
 1  10.0.1.1 (10.0.1.1)  0.056 ms  0.010 ms  0.007 ms
 2  10.0.2.1 (10.0.2.1)  0.029 ms  0.010 ms  0.009 ms
 3  10.0.3.2 (10.0.3.2)  0.032 ms  0.013 ms  0.012 ms
 4  10.0.4.2 (10.0.4.2)  0.034 ms  0.015 ms  0.044 ms
mininet>
```

First, we have to define an interface for each connection to the router. For Ex: r1-eth1, r1-eth2 and r1-eth3 at the router r2,

Then, we have to add the connection interface which a packet needs to take to approach its destination. This should be added at each router for all possible hosts.

```
info(net['r1'].cmd("ip route add 10.0.4.0/24 via 10.0.2.1 dev r1-eth2"))
info(net['r1'].cmd("ip route add 10.0.3.0/24 via 10.0.2.1 dev r1-eth2"))
info(net['r2'].cmd("ip route add 10.0.4.0/24 via 10.0.3.2 dev r2-eth1"))
info(net['r2'].cmd("ip route add 10.0.1.0/24 via 10.0.2.2 dev r2-eth2"))

info(net['r2'].cmd("ip route add 10.0.5.0/24 via 10.0.3.2 dev r2-eth1"))
info(net['r2'].cmd("ip route add 10.0.6.0/24 via 10.0.2.2 dev r2-eth2"))

info(net['r4'].cmd("ip route add 10.0.1.0/24 via 10.0.3.1 dev r4-eth2"))
info(net['r4'].cmd("ip route add 10.0.2.0/24 via 10.0.3.1 dev r4-eth2"))

info(net['r1'].cmd("ip route add 10.0.4.0/24 via 10.0.6.2 dev r1-eth3"))
info(net['r1'].cmd("ip route add 10.0.5.0/24 via 10.0.6.2 dev r1-eth3"))
info(net['r3'].cmd("ip route add 10.0.4.0/24 via 10.0.5.2 dev r3-eth1"))
info(net['r3'].cmd("ip route add 10.0.1.0/24 via 10.0.6.1 dev r3-eth2"))

info(net['r3'].cmd("ip route add 10.0.3.0/24 via 10.0.5.2 dev r3-eth1"))
info(net['r3'].cmd("ip route add 10.0.2.0/24 via 10.0.6.1 dev r3-eth2"))

info(net['r4'].cmd("ip route add 10.0.1.0/24 via 10.0.5.1 dev r4-eth3"))
info(net['r4'].cmd("ip route add 10.0.6.0/24 via 10.0.5.1 dev r4-eth3"))
```

The above picture tells how we should add static routes at each router.