

Part1

1. Take routing tables screenshot before/after on [r1-r4]

Before:

```
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-eth0
192.168.1.0     0.0.0.0         255.255.255.192 U        0      0      0 r1-eth1
192.168.1.64    0.0.0.0         255.255.255.192 U        0      0      0 r1-eth2
mininet> r2 route
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
10.0.0.0         0.0.0.0         255.255.255.0    U        0      0      0 r2-eth0
10.0.1.0         0.0.0.0         255.255.255.0    U        0      0      0 r2-eth1
mininet> r3 route
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
10.0.0.0         0.0.0.0         255.255.255.0    U        0      0      0 r3-eth0
10.0.2.0         0.0.0.0         255.255.255.0    U        0      0      0 r3-eth1
mininet> r4 route
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
10.0.2.0         0.0.0.0         255.255.255.0    U        0      0      0 r4-eth0
140.114.0.0     0.0.0.0         255.255.255.0    U        0      0      0 r4-eth1
mininet>
```

After:

```
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-eth0
140.114.0.0     10.0.1.1       255.255.0.0      UG       20     0      0 r1-eth0
192.168.1.0     0.0.0.0         255.255.255.192 U        0      0      0 r1-eth1
192.168.1.64    0.0.0.0         255.255.255.192 U        0      0      0 r1-eth2
mininet> r2 route
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
10.0.0.0         0.0.0.0         255.255.255.0    U        0      0      0 r2-eth0
10.0.1.0         0.0.0.0         255.255.255.0    U        0      0      0 r2-eth1
140.113.0.0     10.0.1.2       255.255.0.0      UG       20     0      0 r2-eth1
140.114.0.0     10.0.0.2       255.255.0.0      UG       20     0      0 r2-eth0
mininet> r3 route
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
10.0.0.0         0.0.0.0         255.255.255.0    U        0      0      0 r3-eth0
10.0.2.0         0.0.0.0         255.255.255.0    U        0      0      0 r3-eth1
140.113.0.0     10.0.0.1       255.255.0.0      UG       20     0      0 r3-eth0
140.114.0.0     10.0.2.3       255.255.0.0      UG       20     0      0 r3-eth1
mininet> r4 route
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
10.0.2.0         0.0.0.0         255.255.255.0    U        0      0      0 r4-eth0
140.113.0.0     10.0.2.1       255.255.0.0      UG       20     0      0 r4-eth0
140.114.0.0     0.0.0.0         255.255.255.0    U        0      0      0 r4-eth1
mininet>
```

2. Telnet zebra and bgpd daemons of [r1-r4] and take screenshots of routes in zebra and bgpd daemons.

R1

```
zebra> sho ip route bgp
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

B>* 140.114.0.0/16 [20/0] via 10.0.1.1, r1-eth0, 00:00:42
zebra>
```

R2

```
zebra> sho ip route bgp
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

B>* 140.113.0.0/16 [20/0] via 10.0.1.2, r2-eth1, 00:02:03
B>* 140.114.0.0/16 [20/0] via 10.0.0.2, r2-eth0, 00:01:58
zebra>
```

R3

```
zebra> sho ip route bgp
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

B>* 140.113.0.0/16 [20/0] via 10.0.0.1, r3-eth0, 00:02:40
B>* 140.114.0.0/16 [20/0] via 10.0.2.3, r3-eth1, 00:02:45
zebra>
```

R4

```
zebra> sho ip route bgp
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

B>* 140.113.0.0/16 [20/0] via 10.0.2.1, r4-eth0, 00:03:13
zebra>
```

3. Capture BGP packets from wireshark and take screenshot to verify your answer for the following questions

I. Show BGP packets (OPEN, UPDATE, KEEP ALIVE) exchanged by r2 and r3

The picture shows packets captured on R2-eth0 (connect with r3).

| | | | | | | |
|----|--------------|----------|----------|-----|-----|--------------------------------------|
| 23 | 32.971844173 | 10.0.0.2 | 10.0.0.1 | BGP | 125 | OPEN Message |
| 29 | 33.768355352 | 10.0.0.1 | 10.0.0.2 | BGP | 125 | OPEN Message |
| 31 | 33.768570182 | 10.0.0.2 | 10.0.0.1 | BGP | 144 | OPEN Message, KEEPALIVE Message |
| 33 | 33.768683878 | 10.0.0.1 | 10.0.0.2 | BGP | 104 | KEEPALIVE Message, KEEPALIVE Message |
| 35 | 33.768743278 | 10.0.0.2 | 10.0.0.1 | BGP | 85 | KEEPALIVE Message |
| 37 | 34.769012429 | 10.0.0.1 | 10.0.0.2 | BGP | 89 | UPDATE Message |
| 39 | 34.770596591 | 10.0.0.2 | 10.0.0.1 | BGP | 140 | UPDATE Message, UPDATE Message |
| 41 | 36.769553386 | 10.0.0.2 | 10.0.0.1 | BGP | 85 | KEEPALIVE Message |
| 43 | 36.769717817 | 10.0.0.1 | 10.0.0.2 | BGP | 85 | KEEPALIVE Message |

II. What will happen to the routing table if you set r4-eth0 down?

Before

```
mininet> r3 route
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.0       0.0.0.0         255.255.255.0   U        0      0        0 r3-eth0
10.0.2.0       0.0.0.0         255.255.255.0   U        0      0        0 r3-eth1
140.113.0.0    10.0.0.1        255.255.0.0     UG       20     0        0 r3-eth0
140.114.0.0    10.0.2.3        255.255.0.0     UG       20     0        0 r3-eth1
```

After

```
mininet> r3 route
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.0       0.0.0.0         255.255.255.0   U         0      0      0 r3-eth0
10.0.2.0       0.0.0.0         255.255.255.0   U         0      0      0 r3-eth1
140.113.0.0    10.0.0.1        255.255.0.0     UG        20     0      0 r3-eth0
mininet>
```

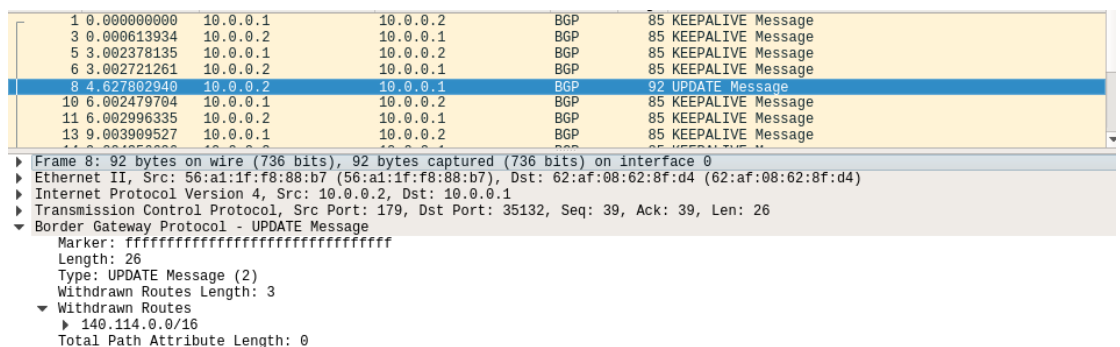
The screenshot above shows routing table of r3. Similarly, other devices will also remove the routing entry 140.114.0.0/16.

III. How does r3 know r4 is unreachable? Explain how

R3 found that it did not receive BGP KEEPALIVE Message from r4 over timeout timer set in quagga configuration. As a result, it determined that r4 is unreachable.

IV. How does r2 know r4 is unreachable? Explain how

R2 received BGP Update message telling it withdrawn routes 140.114.0.0/16 from r3 (as the screenshot below). That is how r2 know r4 is unreachable.



```

1 0.0000000000 10.0.0.1 10.0.0.2 BGP 85 KEEPALIVE Message
3 0.000613934 10.0.0.2 10.0.0.1 BGP 85 KEEPALIVE Message
5 3.002378135 10.0.0.1 10.0.0.2 BGP 85 KEEPALIVE Message
6 3.002721261 10.0.0.2 10.0.0.1 BGP 85 KEEPALIVE Message
8 4.627802948 10.0.0.2 10.0.0.1 BGP 92 UPDATE Message
10 6.002479704 10.0.0.1 10.0.0.2 BGP 85 KEEPALIVE Message
11 6.002996335 10.0.0.2 10.0.0.1 BGP 85 KEEPALIVE Message
13 9.003909527 10.0.0.1 10.0.0.2 BGP 85 KEEPALIVE Message
14 9.004000000 10.0.0.2 10.0.0.1 BGP 85 KEEPALIVE Message

Frame 8: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface 0
  Ethernet II, Src: 56:a1:1f:f8:88:b7 (56:a1:1f:f8:88:b7), Dst: 62:af:08:62:8f:d4 (62:af:08:62:8f:d4)
  Internet Protocol Version 4, Src: 10.0.0.2, Dst: 10.0.0.1
  Transmission Control Protocol, Src Port: 179, Dst Port: 35132, Seq: 39, Ack: 39, Len: 26
  Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 26
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 3
    Withdrawn Routes
      140.114.0.0/16
    Total Path Attribute Length: 0

```

Part2

1. Take screenshot of curl result

```
mininet> h4 curl 140.113.0.40:80
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 3.2 Final//EN"><html>
<title>Directory listing for /</title>
<body>
<h2>Directory listing for /</h2>
<hr>
<ul>
<li><a href="configs/">configs/</a>
<li><a href="dhcpd.conf">dhcpd.conf</a>
<li><a href="Makefile">Makefile</a>
<li><a href="topology.py">topology.py</a>
<li><a href="zebra.conf">zebra.conf</a>
</ul>
<hr>
</body>
</html>
mininet>
```

2. Check reachability and take screenshot

```
mininet> h1 ping h4 -c 1
PING 140.114.0.1 (140.114.0.1) 56(84) bytes of data.
64 bytes from 140.114.0.1: icmp_seq=1 ttl=60 time=0.229 ms

--- 140.114.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.229/0.229/0.229/0.000 ms
mininet> h2 ping h4 -c 1
PING 140.114.0.1 (140.114.0.1) 56(84) bytes of data.
64 bytes from 140.114.0.1: icmp_seq=1 ttl=60 time=0.287 ms

--- 140.114.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.287/0.287/0.287/0.000 ms
mininet> h3 ping h4 -c 1
PING 140.114.0.1 (140.114.0.1) 56(84) bytes of data.
64 bytes from 140.114.0.1: icmp_seq=1 ttl=60 time=0.192 ms

--- 140.114.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.192/0.192/0.192/0.000 ms
mininet> █
```

3. Run wireshark on r1 to take screenshot of input/output packet

| *r1-eth0 | | | | | | |
|--|--------------|--------------|--------------|----------|--------|--|
| File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help | | | | | | |
| icmp | | | | | | |
| No. | Time | Source | Destination | Protocol | Length | Info |
| 18 | 13.875112021 | 140.113.0.30 | 140.114.0.1 | ICMP | 98 | Echo (ping) request id=0x38cc, seq=1/256, ttl=63 (r... |
| 19 | 13.875138532 | 140.114.0.1 | 140.113.0.30 | ICMP | 98 | Echo (ping) reply id=0x38cc, seq=1/256, ttl=61 (r... |
| 26 | 19.019145761 | 140.113.0.40 | 140.114.0.1 | ICMP | 98 | Echo (ping) request id=0x38cf, seq=1/256, ttl=63 (r... |
| 27 | 19.019170509 | 140.114.0.1 | 140.113.0.40 | ICMP | 98 | Echo (ping) reply id=0x38cf, seq=1/256, ttl=61 (r... |

| *r1-eth1 | | | | | | |
|--|-------------|--------------|--------------|----------|--------|--|
| File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help | | | | | | |
| icmp | | | | | | |
| No. | Time | Source | Destination | Protocol | Length | Info |
| 1 | 0.000000000 | 192.168.1.18 | 140.114.0.1 | ICMP | 98 | Echo (ping) request id=0x38cc, seq=1/256, ttl=64 (r... |
| 2 | 0.000045854 | 140.114.0.1 | 192.168.1.18 | ICMP | 98 | Echo (ping) reply id=0x38cc, seq=1/256, ttl=60 (r... |

| *r1-eth2 | | | | | | |
|--|-------------|--------------|--------------|----------|--------|--|
| File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help | | | | | | |
| icmp | | | | | | |
| No. | Time | Source | Destination | Protocol | Length | Info |
| 1 | 0.000000000 | 192.168.1.65 | 140.114.0.1 | ICMP | 98 | Echo (ping) request id=0x38cf, seq=1/256, ttl=64 (r... |
| 2 | 0.000043547 | 140.114.0.1 | 192.168.1.65 | ICMP | 98 | Echo (ping) reply id=0x38cf, seq=1/256, ttl=60 (r... |

The first two packets are result from “h1 ping h4 -c 1” and the last two packets are result from “h2 ping h4 -c 1”. Source ip of ICMP request captured on r1-eth1 is 192.168.1.18, which is h1’s IP address. Source ip of ICMP request captured on r1-eth2 is 192.168.1.65, which is h2’s IP address. 140.113.0.30 is IP address after SNAT from h1 and 140.113.0.40 is IP address after SNAT from h2.