Group 6
Li Wen Ong - 1004663
Chua Qi Bao - 1004494
Madhumitha Balaji - 1004471
Harshit Garg - 1004422
Caryl Beatrice Peneyra - 1004618

Question 2.2

To see the network topology and the connections, we used `net` command in mininet.
To find the IP addresses of a router 's interfaces, for example router R1, we ran the command `R1 ifconfig`
To find IP addresses of a host, for example host h1 in AS1, we ran `h1 ifconfig`

	Host	IP Address
AS 1 (11.0.0.0/8)	h11	11.0.1.1
	h12	11.0.2.1
	h13	11.0.3.1
AS 2 (12.0.0.0/8)	h21	12.0.1.1
	h22	12.0.2.1
	h23	12.0.3.1
AS 3 (13.0.0.0/8)	h31	13.0.1.1
	h32	13.0.2.1
	h33	13.0.3.1

Note: In the below tables, Rx : Ry represents the interface of Router x to Router y

Router Interface	IP Address
R1 : R2	9.0.0.1
R1 : h11	11.0.1.254
R1 : h12	11.0.2.254
R1 : h13	11.0.3.254
R1 : R4	9.0.4.1

Router Interface	IP Address
R2 : R1	9.0.0.2
R2 : h21	12.0.1.254
R2 : h22	12.0.2.254
R2 : h23	12.0.3.254
R2 : R3	9.0.1.1

Router Interface	IP Address
R3 : R2	9.0.1.2
R3 : h31	13.0.1.254
R3 : h32	13.0.2.254
R3 : h33	13.0.3.254
R4 : R1	9.0.4.2

Question 2.3a

Describe in detail the BGP traffic you were able to observe during re-establishment of routes.

From analysing the **BGP traffic in R2-eth4**, we observed that R1 (9.0.0.1) and R2 (9.0.0.2) exchanged OPEN messages and KEEPALIVE messages to establish a connection (in Figure 1). Then they exchanged Network Layer Reachability Information (NLRI) through UPDATE messages (in Figure 1). R2 indicated 12.0.0.8/8 and 13.0.0.0/8 in its NLRI to R1 (in Figure 2), while R1 indicated 11.0.0.0/8 in its NLRI to R2 (in Figure 3). After that, R1 and R2 exchanged KEEPALIVE messages continually (in Figure 1).

From analysing the **BGP traffic in R2-eth5,** similar exchanges of OPEN, KEEPALIVE and UPDATE messages can be found between R2 (9.0.1.1) and R3 (9.0.1.2) during re-establishment of routes (in Figure 5). R2 indicated 11.0.0.0/8 and 12.0.0.0/8 in its NLRI to R3 (Figure 6), while R3 indicated 13.0.0.0/8 in its NLRI to R2 (Figure 7).

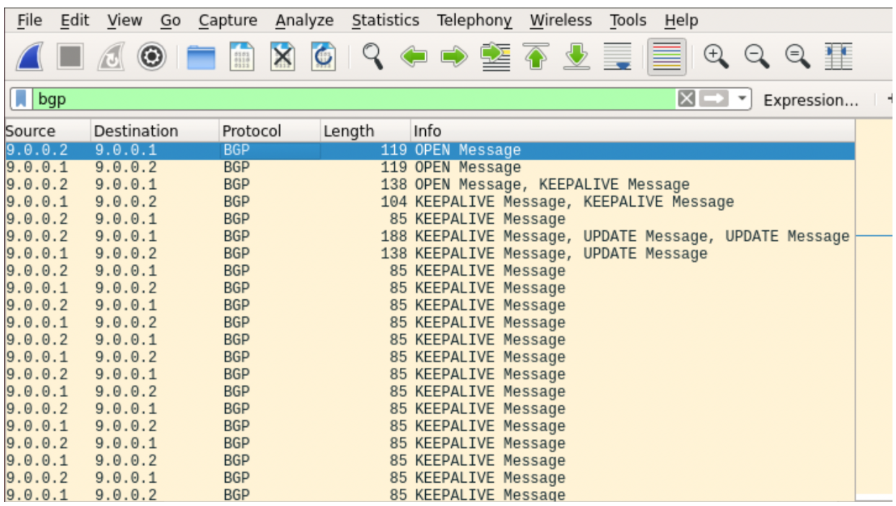

Figure 1: BGP traffic from R2-eth4 during re-establishment of routes

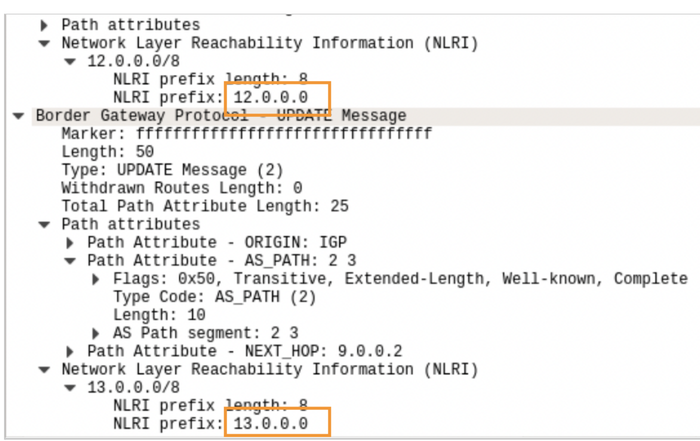

Figure 2: UPDATE Message from R2 to R1 indicating NLRI 12.0.0.0/8 and 13.0.0.0/8

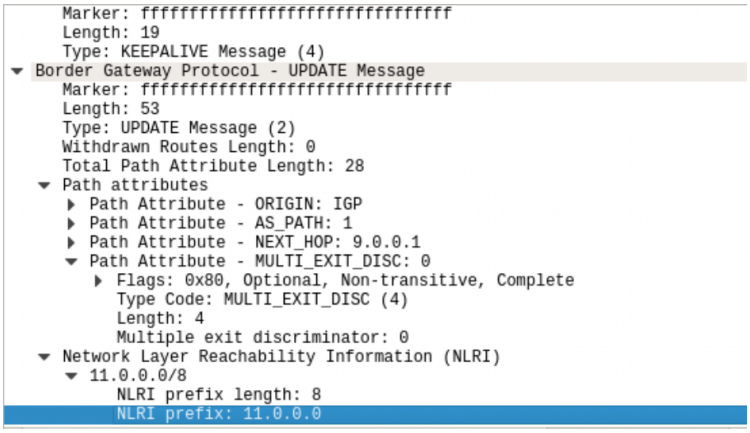

Figure 3: UPDATE Message from R1 to R2 indicating NLRI 11.0.0.0/8

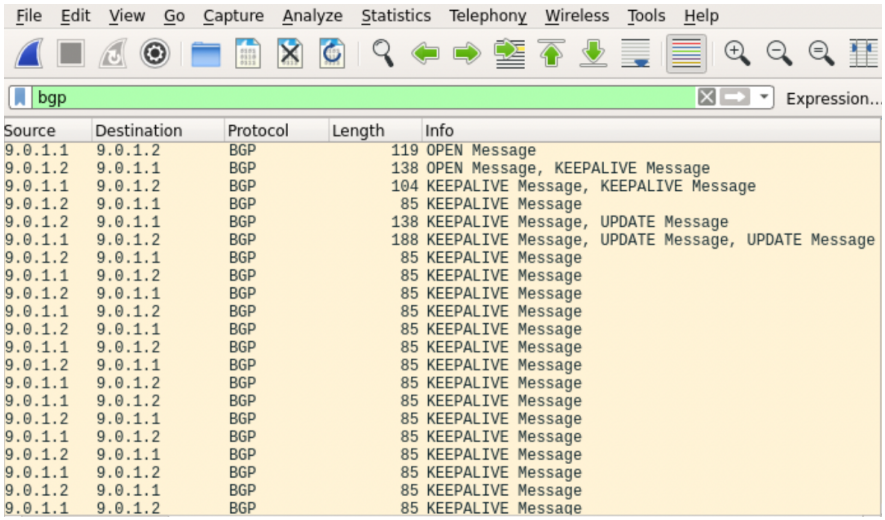

Figure 4: BGP traffic from R2-eth5 during re-establishment of routes

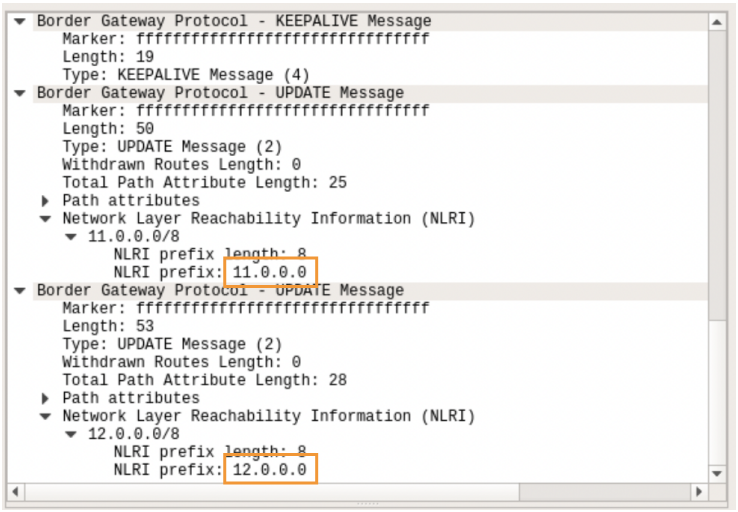

Figure 5: UPDATE Message from R2 to R3 indicating NLRI 11.0.0.0/8 and 12.0.0.0/8

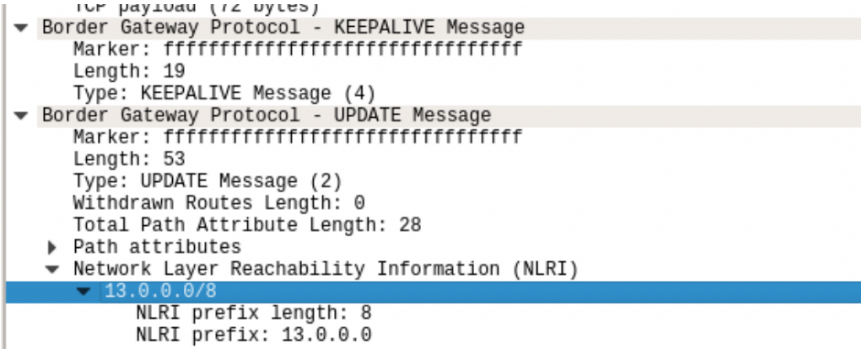

Figure 6: UPDATE Message from R3 to R2 indicating NLRI 13.0.0.0/8

Question 2.3b

Was it initially possible to reach 13.0.1.1 from AS1 (h11 and R1, respectively)? If it didn't work initially, what caused that and what did you do to fix it?

It is possible to reach 13.0.1.1 from h11 (in Figure 7).

```
mininet> h11 ping 13.0.1.1
PING 13.0.1.1 (13.0.1.1) 56(84) bytes of data.
64 bytes from 13.0.1.1: icmp_seq=1 ttl=61 time=0.089 ms
64 bytes from 13.0.1.1: icmp_seq=2 ttl=61 time=0.061 ms
64 bytes from 13.0.1.1: icmp_seq=3 ttl=61 time=0.132 ms
64 bytes from 13.0.1.1: icmp_seq=4 ttl=61 time=0.058 ms
64 bytes from 13.0.1.1: icmp_seq=5 ttl=61 time=0.197 ms
64 bytes from 13.0.1.1: icmp_seq=6 ttl=61 time=0.096 ms
64 bytes from 13.0.1.1: icmp_seq=7 ttl=61 time=0.189 ms
64 bytes from 13.0.1.1: icmp_seq=8 ttl=61 time=0.118 ms
64 bytes from 13.0.1.1: icmp_seq=9 ttl=61 time=0.053 ms
```

Figure 7: h11 can reach 13.0.1.1

But it's not possible to reach 13.0.1.1 from R1 (in Figure 8), because R1 has no gateway that leads to AS3 (in Figure 9). The only way for R1 to reach AS3 is via R2 but R2 did not advertise that it can connect R1 to AS3.

```
mininet> R1 ping 13.0.1.1
```

Figure 8: R1 cannot reach 13.0.1.1

```
mininet> R1 route                                    Note: R1 has no connection to AS3
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref
  Use Iface
9.0.0.0         *               255.255.255.0   U     0      0
    0 R1-eth4
9.0.4.0         *               255.255.255.0   U     0      0
    0 R1-eth5
11.0.1.0        *               255.255.255.0   U     0      0
    0 R1-eth1
11.0.2.0        *               255.255.255.0   U     0      0
    0 R1-eth2
11.0.3.0        *               255.255.255.0   U     0      0
    0 R1-eth3
12.0.0.0  AS2   9.0.0.2   R2    255.0.0.0       UG    0      0
    0 R1-eth4
```

Figure 9: R1 Routing table showing that R1 has no connection to AS3 (before)

I added `network 9.0.0.1/8` and `network 9.0.1.2/8` to `bgpd-R2.conf` (in Figure 10) so that R2 will advertise that it can connect R1 (9.0.0.1/8) to R3 (9.0.1.2/8). Now R1 has a gateway that leads to AS3 (in Figure 11).

bgpd-R2.conf

```
!
! $Id: bgpd.conf.sample,v 1.1 2002/12/13 20:15:29 paul Exp $
!

hostname bgpd-R2
password zebra
enable password zebra

router bgp 2
  bgp router-id 9.0.0.2
  network 12.0.0.0/8
  network 9.0.0.1/8
  network 9.0.1.2/8
  neighbor 9.0.0.1 remote-as 1
  neighbor 9.0.0.1 update-source 9.0.0.2
  neighbor 9.0.0.1 ebgp-multihop
  neighbor 9.0.0.1 next-hop-self
  neighbor 9.0.0.1 timers 5 5
  neighbor 9.0.1.2 remote-as 3

  neighbor 9.0.1.2 update-source 9.0.1.1
  neighbor 9.0.1.2 ebgp-multihop

  neighbor 9.0.1.2 next-hop-self
  neighbor 9.0.1.2 timers 5 5

log file /tmp/R2-bgpd.log
```

Figure 10: Edited version of `bgpd-R2.conf`

```
mininet> R1 route                                    Note: Now R1 has connection to AS3
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref
  Use Iface
9.0.0.0         *               255.255.255.0   U     0      0
    0 R1-eth4
9.0.0.0         9.0.0.2         255.0.0.0       UG    0      0
    0 R1-eth4
9.0.4.0         *               255.255.255.0   U     0      0
    0 R1-eth5
11.0.1.0        *               255.255.255.0   U     0      0
    0 R1-eth1
11.0.2.0        *               255.255.255.0   U     0      0
    0 R1-eth2
11.0.3.0        *               255.255.255.0   U     0      0
    0 R1-eth3
12.0.0.0  AS2   9.0.0.2   R2    255.0.0.0       UG    0      0
    0 R1-eth4
13.0.0.0  AS3   9.0.0.2   R2    255.0.0.0       UG    0      0
    0 R1-eth4
```

Figure 11: R1 Routing table showing that R1 has connection to AS3 (after)

Question 2.4

Q: Assume the following setting: a user from AS1 wants to visit a website on 13.0.1.1. A malicious attacker wants to redirect the user to its own webserver instead. The attacker has control over AS4, which is BGP-peering with AS1. How can the attacker reach his goal?

Answer: Since attacker has control AS4, he can change R4 network IP address to 13.0.0.0/8. When AS4 peer with AS1, it will advertise its gateway router's IP address to the gateway router in AS1 (i.e. R1 will add R4's IP address to its forwarding table). When a user from AS1 visits the website on 13.0.1.1., it will redirect the user to the attacker's webserver instead.

Details before and after the attack on BGP

Before the attack on BGP, the user visits the website from the default web server (as seen in Figure 12).

After the attack on BGP, the user is redirected to AS4 and the attack is successful as seen in Figure 13. To execute the attack, we modified the /etc/quagga/bgpd-R4.conf file by changing the network IP address to 13.0.0.0/8 (as seen in Figure 14).

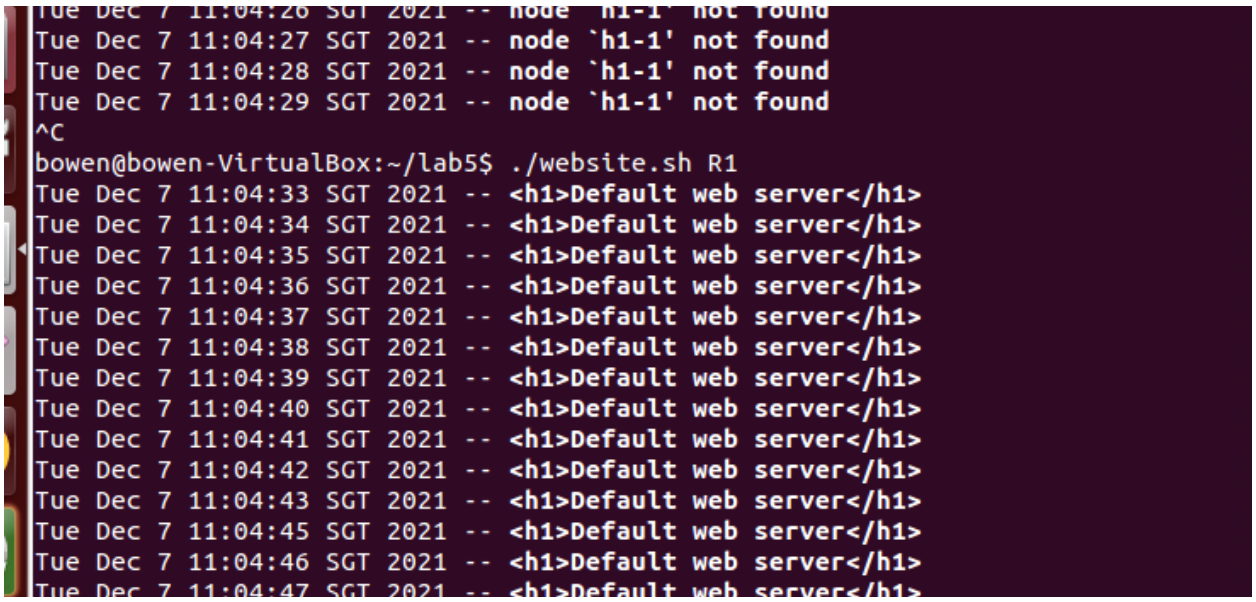

Figure 12: Before attack on BGP

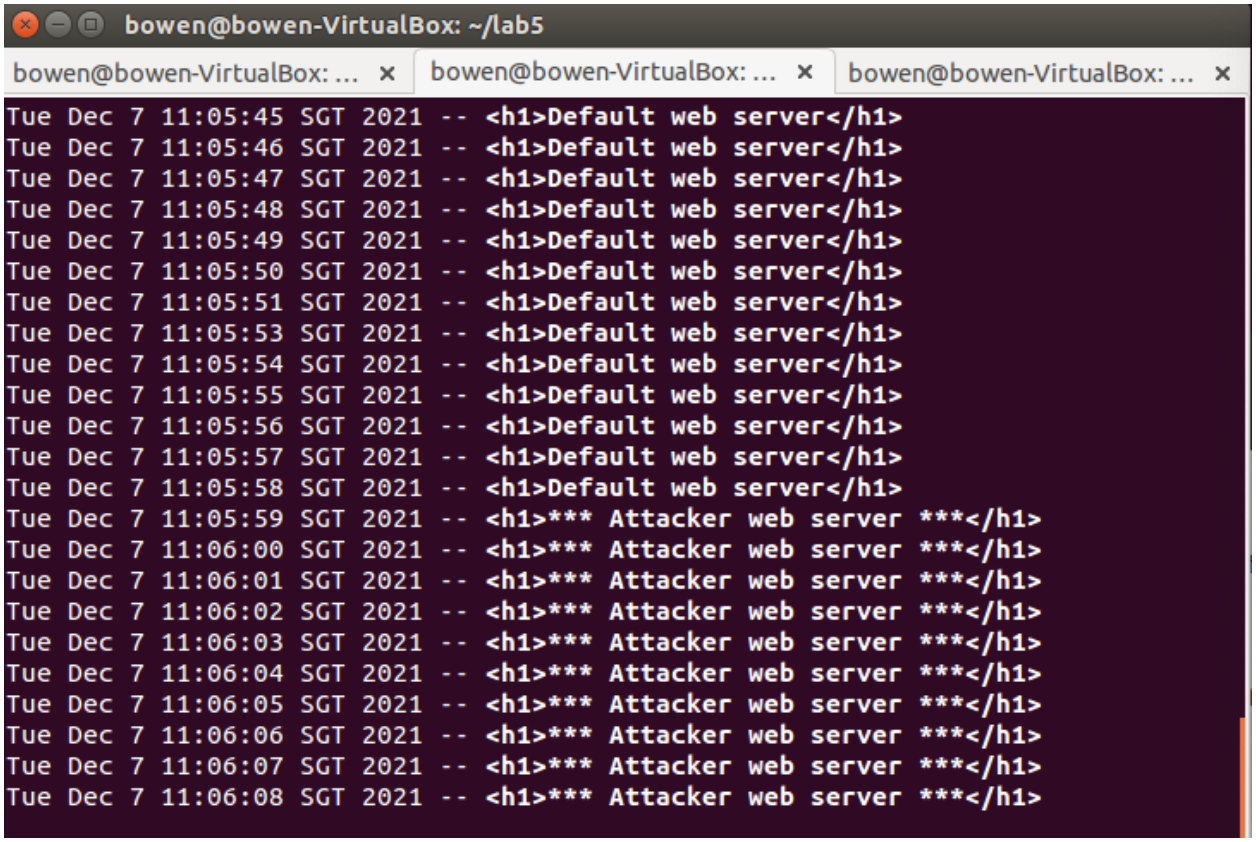

Figure 13: After attack on BGP

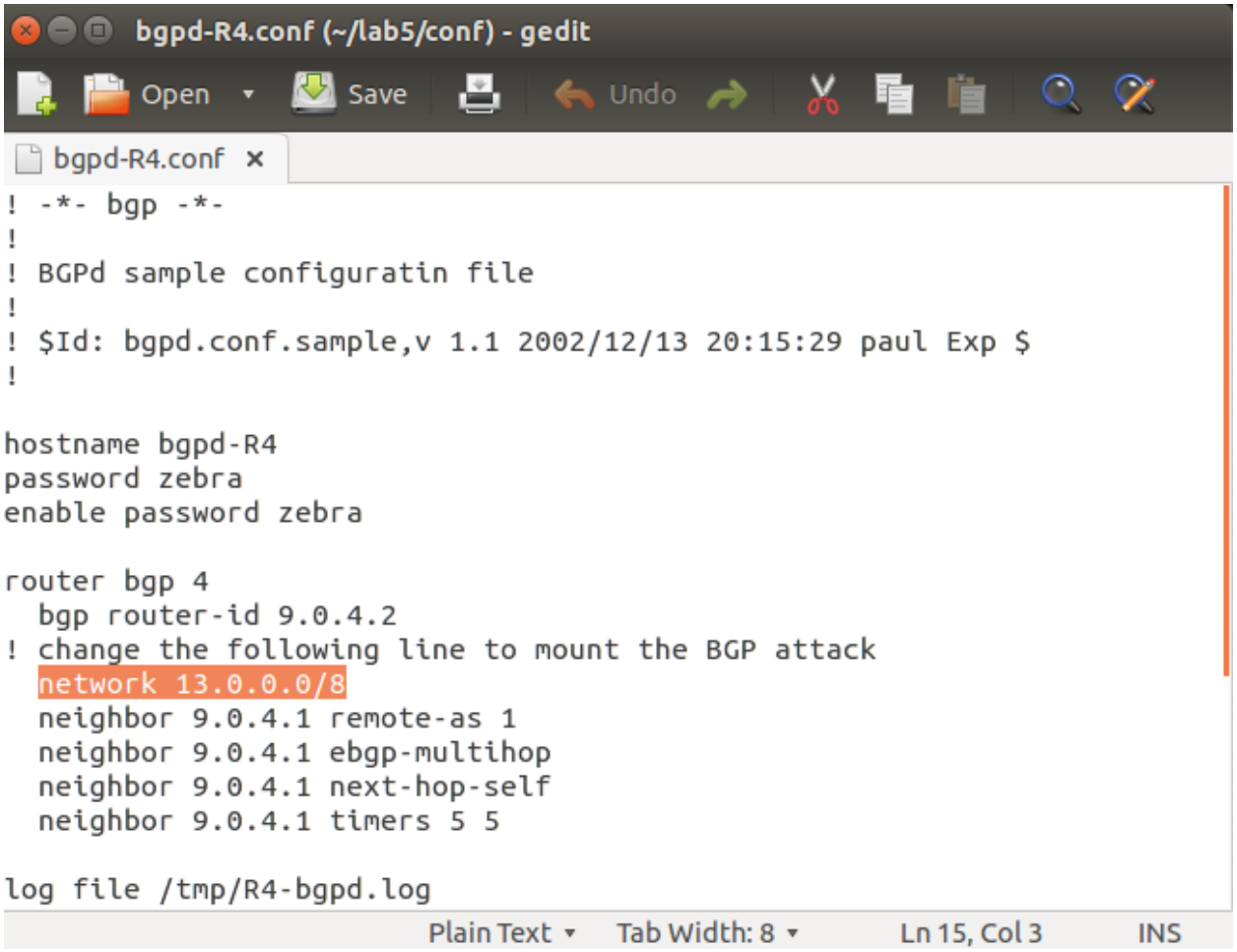

Figure 13: Modified `/etc/quagga/bgpd-R4.conf` file