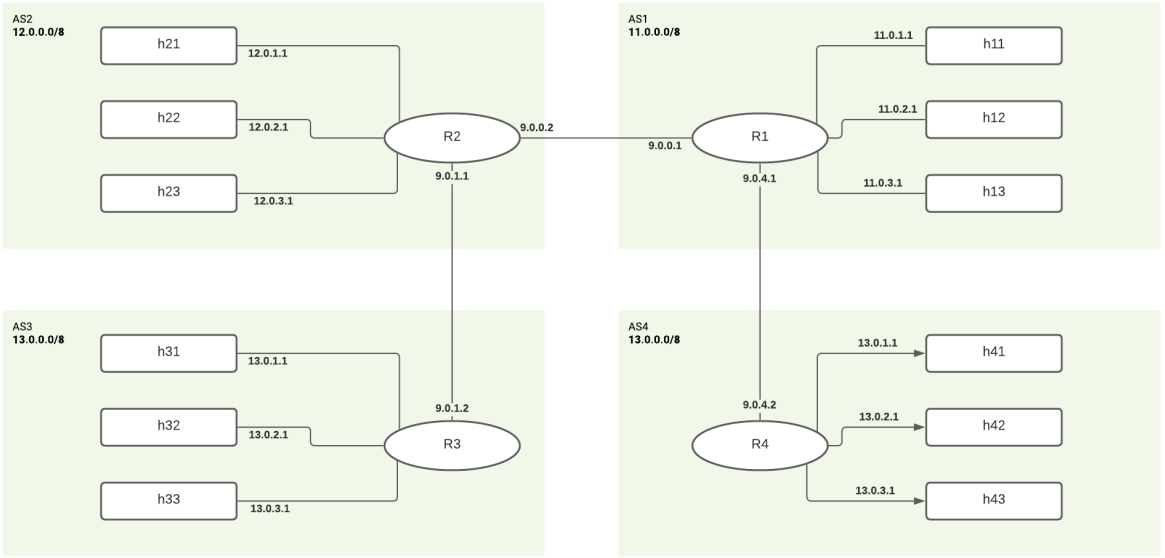
Lab 5: BGP routing

Part 1: Topology with IP addresses of all routers and Hosts/IPs in the ASes



Part 2: BGP traffic observed during re-establishment of routes

Following BGP traffic is seen between routers:

- When “clear bgp external” is run, we see a NOTIFICATION message sent to the other routers with major code 6 - telling them to close the connection[1]. When the connection is closed.
- After this a three way handshake is done between the two routers and an OPEN KEEPALIVE message is sent to initiate BGP communication. This message includes the IP address of the router and other information necessary to open the connection. The KEEPALIVE message is used to keep the session running even if there are no BGP messages to be passed.
- UPDATE messages are exchanged between the routers which contain Network Layer Reachability Information sections that detail ASes that are reachable from their routers.
- The routers subsequently keep passing KEEPALIVE messages to each other to keep the connection open.

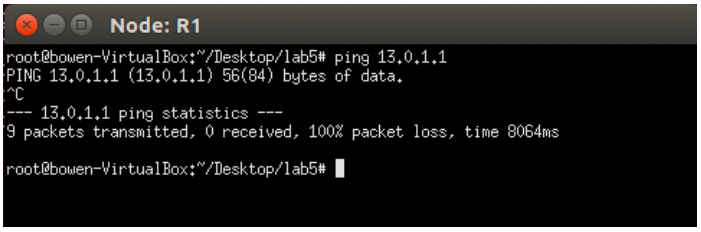
```
▼ Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 53
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 28
  ▶ Path attributes
  ▼ Network Layer Reachability Information (NLRI)
    ▼ 11.0.0.0/8
        NLRI prefix length: 8
        NLRI prefix: 11.0.0.0
```

Part 3: Reaching 13.0.1.1 from AS1 (h11 and R1), and modifications

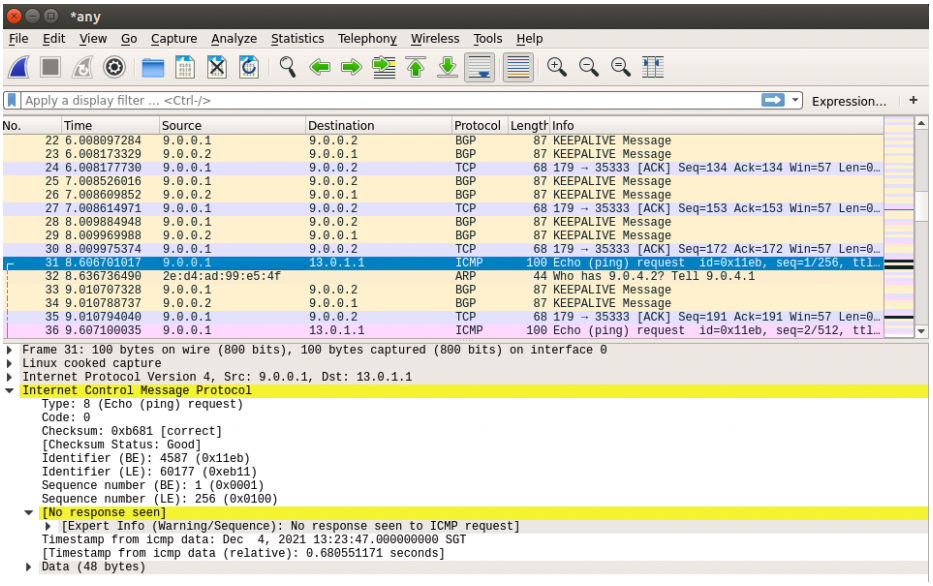
Trying to reach h33 from h11:
Yes, we are able to ping h33 from h11, but this is only after the routing table for all the relevant routers have been fully updated, which takes a few seconds.

```
mininet> h11 ping h33
PING 13.0.3.1 (13.0.3.1) 56(84) bytes of data.
64 bytes from 13.0.3.1: icmp_seq=1 ttl=61 time=0.045 ms
64 bytes from 13.0.3.1: icmp_seq=2 ttl=61 time=0.042 ms
64 bytes from 13.0.3.1: icmp_seq=3 ttl=61 time=0.045 ms
64 bytes from 13.0.3.1: icmp_seq=4 ttl=61 time=0.038 ms
64 bytes from 13.0.3.1: icmp_seq=5 ttl=61 time=0.045 ms
64 bytes from 13.0.3.1: icmp_seq=6 ttl=61 time=0.033 ms
^C
--- 13.0.3.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5135ms
rtt min/avg/max/mdev = 0.033/0.041/0.045/0.006 ms
```

Try to reach 13.0.1.1 from R1:
We are unable to ping 13.0.1.1 from R1 as shown below.



When using wireshark to analyze and monitor the network, it is seen that when router R1 with source IP 9.0.0.1 tries to send an ICMP request generated by the ping utility to host with IP 13.0.1.1, no response was observed. This meant that R1 could reach 13.0.1.1 but there is a possibility of 13.0.1.1 not being able to reply to R1.



When taking a look at the routing tables for all 3 routers R1, R2 and R3, it became evident that the cause for this was due to router R3 not knowing how to route to router R1 that has the IP 9.0.0.0/8.

```
mininet> R1 route
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        9.0.0.2         255.0.0.0       UG    0      0        0 R1-eth4
13.0.0.0        9.0.0.2         255.0.0.0       UG    0      0        0 R1-eth4
mininet> R2 route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
9.0.0.0         *               255.255.255.0   U     0      0        0 R2-eth4
9.0.1.0         *               255.255.255.0   U     0      0        0 R2-eth5
11.0.0.0        9.0.0.1         255.0.0.0       UG    0      0        0 R2-eth4
12.0.1.0        *               255.255.255.0   U     0      0        0 R2-eth1
12.0.2.0        *               255.255.255.0   U     0      0        0 R2-eth2
12.0.3.0        *               255.255.255.0   U     0      0        0 R2-eth3
13.0.0.0        9.0.1.2         255.0.0.0       UG    0      0        0 R2-eth5
mininet> R3 route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
9.0.1.0         *               255.255.255.0   U     0      0        0 R3-eth4
11.0.0.0        9.0.1.1         255.0.0.0       UG    0      0        0 R3-eth4
12.0.0.0        9.0.1.1         255.0.0.0       UG    0      0        0 R3-eth4
13.0.1.0        *               255.255.255.0   U     0      0        0 R3-eth1
13.0.2.0        *               255.255.255.0   U     0      0        0 R3-eth2
13.0.3.0        *               255.255.255.0   U     0      0        0 R3-eth3
mininet>
```

To resolve this, we had to modify the bgpd-R2.conf file by adding the line "**network 9.0.0.0/8**". What this does is that R2 will now announce that it has a path to 9.0.0.0/8. Hence, when we observe the routing table for R3 again, we can see that it now knows that there is a path to 9.0.0.0/8, and that it is through R2. Therefore, hosts in R3 will now be able to respond to R1's ping request. This is confirmed by wireshark showing that there is an ICMP reply to 9.0.0.1

Routing table for R3:

```
   Network          Next Hop            Metric LocPrf Weight Path
*> 9.0.0.0          9.0.1.1                  0             0 2 i
*> 11.0.0.0         9.0.1.1                                0 2 1 i
*> 12.0.0.0         9.0.1.1                  0             0 2 i
*> 13.0.0.0         0.0.0.0                  0         32768 i
```

```
Node: R1
root@bowen-VirtualBox:~/Desktop/lab5# 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=62 time=0.050 ms
64 bytes from 13.0.1.1: icmp_seq=2 ttl=62 time=0.072 ms
^C
--- 13.0.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.050/0.061/0.072/0.011 ms
root@bowen-VirtualBox:~/Desktop/lab5#
```

```
    20 5.648385195   9.0.0.1          13.0.1.1         ICMP     100 Echo (ping) request  id=0x1627, seq=1/256, ttl…
    21 5.648410620   13.0.1.1         9.0.0.1          ICMP     100 Echo (ping) reply    id=0x1627, seq=1/256, tt…
    22 6.008229492   9.0.0.1          9.0.0.2          BGP       87 KEEPALIVE Message
    23 6.008330241   9.0.0.2          9.0.0.1          BGP       87 KEEPALIVE Message
    24 6.008337181   9.0.0.1          9.0.0.2          TCP       68 43004 → 179 [ACK] Seq=134 Ack=134 Win=58 Len=0…
    25 6.648118892   9.0.0.1          13.0.1.1         ICMP     100 Echo (ping) request  id=0x1627, seq=2/512, ttl…
    26 6.648146046   13.0.1.1         9.0.0.1          ICMP     100 Echo (ping) reply    id=0x1627, seq=2/512, ttl…
    27 7.009523560   9.0.0.1          9.0.0.2          BGP       87 KEEPALIVE Message
    28 7.009638645   9.0.0.2          9.0.0.1          BGP       87 KEEPALIVE Message
    29 7.009643963   9.0.0.1          9.0.0.2          TCP       68 43004 → 179 [ACK] Seq=153 Ack=153 Win=58 Len=0…
▸ Frame 21: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0
▸ Linux cooked capture
▸ Internet Protocol Version 4, Src: 13.0.1.1, Dst: 9.0.0.1
▾ Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x2a65 [correct]
    [Checksum Status: Good]
    Identifier (BE): 5671 (0x1627)
    Identifier (LE): 10006 (0x2716)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Request frame: 20]
    [Response time: 0.025 ms]
```

Part 4: Malicious attack on BGP

When we run the script ./website.sh R1, we can see that hosts in AS1 would continuously contact the web server on 13.0.1.1, by using R1's interface IP address of 9.0.0.1, which routes through AS2.

```
Mon Dec 6 12:16:16 SGT 2021 -- <h1>Default web server</h1>
Mon Dec 6 12:16:17 SGT 2021 -- <h1>Default web server</h1>
Mon Dec 6 12:16:18 SGT 2021 -- <h1>Default web server</h1>
Mon Dec 6 12:16:19 SGT 2021 -- <h1>Default web server</h1>
```

```
     4 0.166411974   9.0.0.1         13.0.1.1        TCP      76 40056 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=146…
     5 0.166480392   13.0.1.1        9.0.0.1         TCP      76 80 → 40056 [SYN, ACK] Seq=0 Ack=1 Win=28960 Le…
     6 0.166488274   9.0.0.1         13.0.1.1        TCP      68 40056 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 T…
     7 0.166610857   9.0.0.1         13.0.1.1        HTTP    140 GET / HTTP/1.1
     8 0.166651266   13.0.1.1        9.0.0.1         TCP      68 80 → 40056 [ACK] Seq=1 Ack=73 Win=29184 Len=0 …
     9 0.166827353   13.0.1.1        9.0.0.1         TCP      85 80 → 40056 [PSH, ACK] Seq=1 Ack=73 Win=29184 L…
    10 0.166829177   9.0.0.1         13.0.1.1        TCP      68 40056 → 80 [ACK] Seq=73 Ack=18 Win=29696 Len=0…
    11 0.166843050   13.0.1.1        9.0.0.1         TCP     105 80 → 40056 [PSH, ACK] Seq=18 Ack=73 Win=29184 …
    12 0.166844830   9.0.0.1         13.0.1.1        TCP      68 40056 → 80 [ACK] Seq=73 Ack=55 Win=29696 Len=0…
    13 0.166859703   13.0.1.1        9.0.0.1         TCP     105 80 → 40056 [PSH, ACK] Seq=55 Ack=73 Win=29184 …
    14 0.166860688   9.0.0.1         13.0.1.1        TCP      68 40056 → 80 [ACK] Seq=73 Ack=92 Win=29696 Len=0…
    15 0.166868790   13.0.1.1        9.0.0.1         TCP      93 80 → 40056 [PSH, ACK] Seq=92 Ack=73 Win=29184 …
    16 0.166869687   9.0.0.1         13.0.1.1        TCP      68 40056 → 80 [ACK] Seq=73 Ack=117 Win=29696 Len=…
    17 0.166878400   13.0.1.1        9.0.0.1         TCP      70 80 → 40056 [PSH, ACK] Seq=117 Ack=73 Win=29184…
```

To be able to perform the attack, we had to modify the **bgpd-R4.conf** file by having R4 announce that it has a path to **13.0.0.0/8** instead. This will update the path in the routing table of R1 if the attacker has advertised a route to the same destination IP address but with a **better path**, which is the case as seen in the routing table below, since the path to 13.0.0.0 is now via 9.0.4.2.

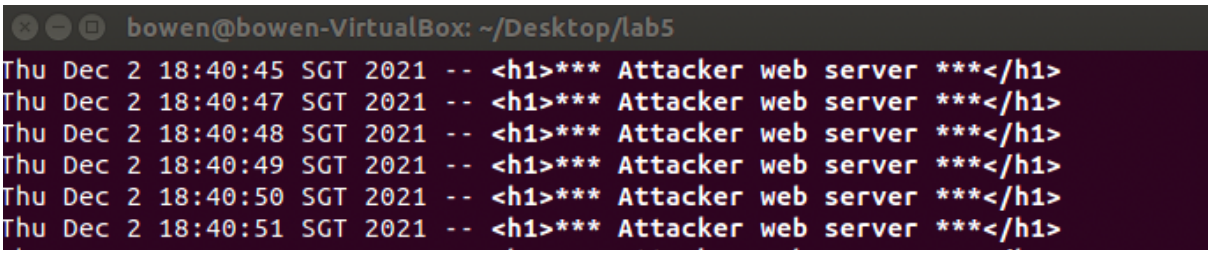
bgpd-R4.conf file

```
router bgp 4
  bgp router-id 9.0.4.2
! change the following line to mount the BGP attack
  network 13.0.0.0/8
  neighbor 9.0.4.1 remote-as 1
  neighbor 9.0.4.1 ebgp-multihop
  neighbor 9.0.4.1 next-hop-self
  neighbor 9.0.4.1 timers 5 5
```

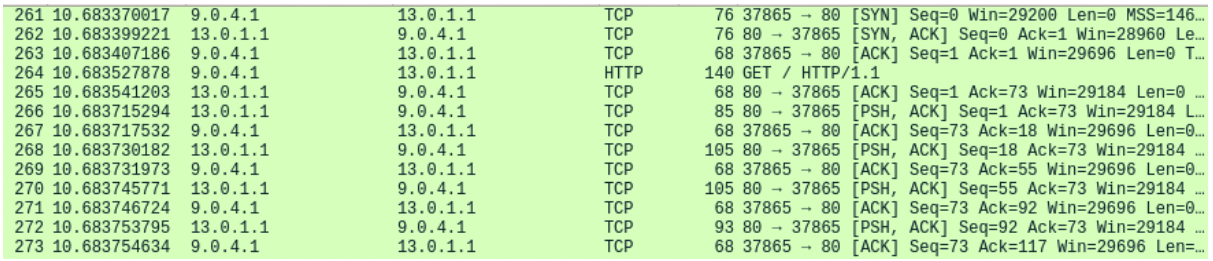
Routing table of R1 when attack has started

```
   Network          Next Hop            Metric LocPrf Weight Path
*> 9.0.0.0          9.0.0.2                  0             0 2 i
*> 11.0.0.0         0.0.0.0                  0         32768 i
*> 12.0.0.0         9.0.0.2                  0             0 2 i
*> 13.0.0.0         9.0.4.2                  0             0 4 i
*                   9.0.0.2                                0 2 3 i
```

When the attack has started, it can be seen that the website results are now different from before, as R1 has actually contacted the attacker web server instead of the default web server



When using wireshark to monitor the results, it is now observed that for hosts in AS1 to get to the destination 13.0.1.1, they use R1's interface IP address of **9.0.4.1** instead, which is linked to R4 and goes through AS4 that is used by the malicious attacker. Hence, the malicious attacker has successfully redirected users in AS1 to its own web server by broadcasting a better path to the same destination.



Extra

Nodes:

```
mininet> nodes
available nodes are:
R1 R2 R3 R4 c0 h11 h12 h13 h21 h22 h23 h31 h32 h33 h41 h42 h43
```

Links:

```
mininet> net
h11 h11-eth0:R1-eth1
h12 h12-eth0:R1-eth2
h13 h13-eth0:R1-eth3
h21 h21-eth0:R2-eth1
h22 h22-eth0:R2-eth2
h23 h23-eth0:R2-eth3
h31 h31-eth0:R3-eth1
h32 h32-eth0:R3-eth2
h33 h33-eth0:R3-eth3
h41 h41-eth0:R4-eth1
h42 h42-eth0:R4-eth2
h43 h43-eth0:R4-eth3
R1 R1-eth1:h11-eth0 R1-eth2:h12-eth0 R1-eth3:h13-eth0 R1-eth4:R2-eth4 R1-eth5:R4
-eth4
R2 R2-eth1:h21-eth0 R2-eth2:h22-eth0 R2-eth3:h23-eth0 R2-eth4:R1-eth4 R2-eth5:R3
-eth4
R3 R3-eth1:h31-eth0 R3-eth2:h32-eth0 R3-eth3:h33-eth0 R3-eth4:R2-eth5
R4 R4-eth1:h41-eth0 R4-eth2:h42-eth0 R4-eth3:h43-eth0 R4-eth4:R1-eth5
c0
```

All info:

```
mininet> dump
<Host h11: h11-eth0:10.0.0.1 pid=5560>
<Host h12: h12-eth0:10.0.0.2 pid=5561>
<Host h13: h13-eth0:10.0.0.3 pid=5562>
<Host h21: h21-eth0:10.0.0.4 pid=5564>
<Host h22: h22-eth0:10.0.0.5 pid=5565>
<Host h23: h23-eth0:10.0.0.6 pid=5566>
<Host h31: h31-eth0:10.0.0.7 pid=5567>
<Host h32: h32-eth0:10.0.0.8 pid=5568>
<Host h33: h33-eth0:10.0.0.9 pid=5569>
<Host h41: h41-eth0:10.0.0.10 pid=5570>
<Host h42: h42-eth0:10.0.0.11 pid=5571>
<Host h43: h43-eth0:10.0.0.12 pid=5572>
<Router R1: R1-eth1:None,R1-eth2:None,R1-eth3:None,R1-eth4:None,R1-eth5:None pid=5573>
<Router R2: R2-eth1:None,R2-eth2:None,R2-eth3:None,R2-eth4:None,R2-eth5:None pid=5574>
<Router R3: R3-eth1:None,R3-eth2:None,R3-eth3:None,R3-eth4:None pid=5575>
<Router R4: R4-eth1:None,R4-eth2:None,R4-eth3:None,R4-eth4:None pid=5576>
<OVSController c0: 127.0.0.1:6633 pid=5552>
```