**50.012 Networks Lab 5**

(Screenshot evidences at the back of the writeup)

**Topology**

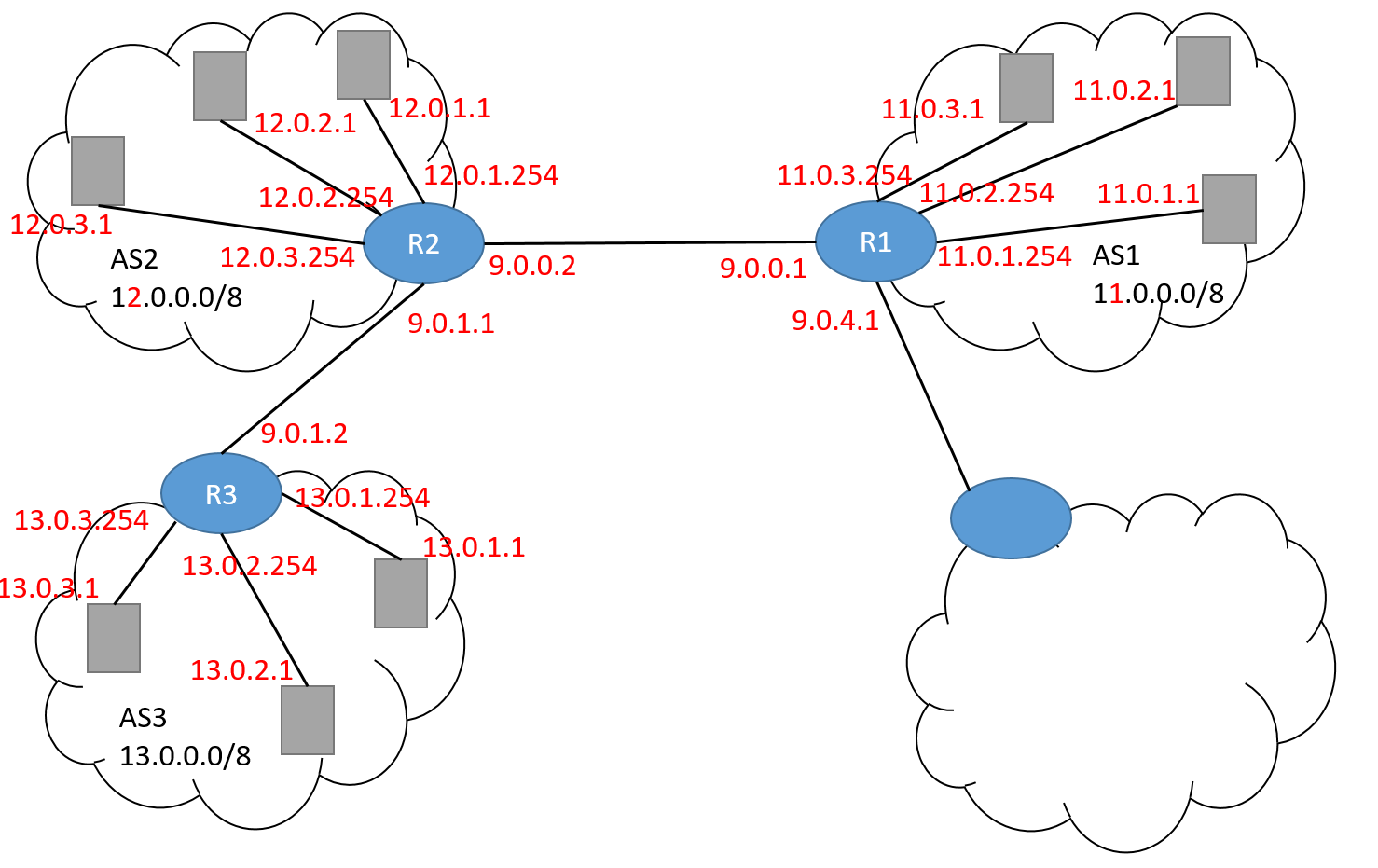
IP addresses of all routers:

|  |  |  |
| --- | --- | --- |
| **Router** | **Interface** | **IP Address** |
| R1 | R1-eth1 | 11.0.1.254 |
| R1-eth2 | 11.0.2.254 |
| R1-eth3 | 11.0.3.254 |
| R1-eth4 | 9.0.0.1 |
| R1-eth5 | 9.0.4.1 |
| R2 | R2-eth1 | 12.0.1.254 |
| R2-eth2 | 12.0.2.254 |
| R2-eth3 | 12.0.3.254 |
| R2-eth4 | 9.0.0.2 |
| R2-eth5 | 9.0.1.1 |
| R3 | R3-eth1 | 13.0.1.254 |
| R3-eth2 | 13.0.2.254 |
| R3-eth3 | 13.0.3.254 |
| R3-eth4 | 9.0.1.2 |

Hosts/IPs in all the ASs:

|  |  |  |  |
| --- | --- | --- | --- |
| **AS** | **Host** | **Interface** | **IP Address** |
| AS1 | h11 | h11-eth0 | 11.0.1.1 |
| h12 | h12-eth0 | 11.0.2.1 |
| h13 | h13-eth0 | 11.0.3.1 |
| AS2 | h21 | h21-eth0 | 12.0.1.1 |
| h22 | h22-eth0 | 12.0.2.1 |
| h23 | h23-eth0 | 12.0.3.1 |
| AS3 | h31 | h31-eth0 | 13.0.1.1 |
| h32 | h32-eth0 | 13.0.2.1 |
| h33 | h33-eth0 | 13.0.3.1 |

Annotated diagram:



**BGP Traffic**

Observation (before modification)

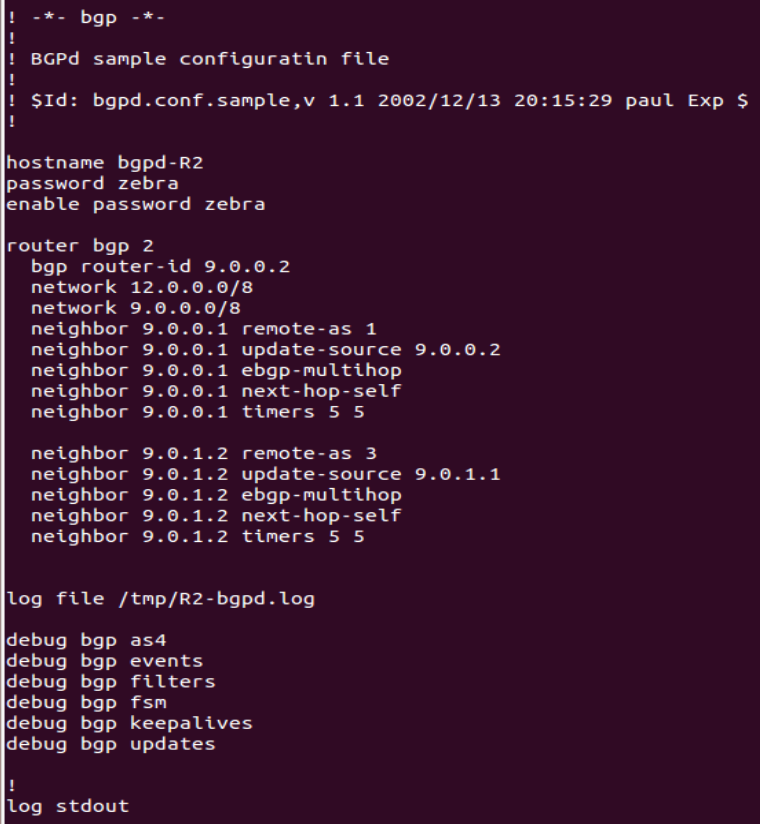
After clear bgp external was executed to clear the routes, according to the wireshark’s capture of packets on R1-eth0, the occasional BGP ‘KEEP ALIVE’ and TCP packets were stopped completely for a few seconds. Then, special packets such as BGP ‘NOTIFICATION’ and ‘OPEN’ message packets were received and the BGP connection was re-established and the occasional BGP and TCP packets resume.

During the moment where the BGP ‘KEEP ALIVE’ packets were stopped, h11 is unable to reach h33 (checked by executing h11 ping h33 on the mininet console). When the BGP packets resumed the packet exchange, h11 can reach h33 again. The connection between the 2 hosts is temporarily lost when the routes were cleared because without the BGP protocol to advertise the path to hosts on the other AS, the BGP routing would not be set up and a host in AS1 would not be able to reach AS3 and vice versa.

R1 cannot reach h33 (checked by executing R1 ping h33 on the mininet console) throughout the re-establishing routes process.

Modification

In bgpd-R2.conf, an extra line was added: “network 9.0.0.0/8”.



Observation (after modification)

Both h11 and R1 can reach h33 (checked by executing h11 ping h33 and R1 ping h33 respectively).

Explanation

Initially, h11 can reach h33 (13.0.1.1) but R1 cannot reach h33.

h11 can reach h33 because R3 advertises AS3’s subnet (i.e. 13.0.0.0/8) to its neighbor R2 and R2 advertises this fact to R1. Thus all hosts in AS1 can reach all hosts in AS3. In fact, since all gateway routers advertise their AS’s subnet to their neighbors, any host in any AS can reach any host in any other subnet.

However, R1 cannot reach h33 because R1’s IP addresses are not considered part of AS1’s hosts. R1’s network interfaces are links to the hosts in AS1 but these interfaces (i.e. R1-eth1, R1-eth2, R1-eth3) are not hosts themselves.

In order R1 to reach a host in AS3, R2 has to advertise that it can reach both routers as hosts. This is done by adding the 9.0.0.0/8 subnet to R2’s advertising, so that the BGP protocol will recognise R1’s eth4 and R3’s eth4 as hosts to be reached. Once this is added, hosts in AS3 can reach R1 and hosts in AS1 can reach R3.

**BGP Attack**

Before the attack started, h11 continuously contacts a webserver on 13.0.1.1 from R1. This can be seen as the packets captured on Wireshark flow to and from 9.0.0.1 and 13.0.1.1, as well as 9.0.0.1 and 9.0.0.2 which reflect the KEEPALIVE message between R1 and R2. The website.sh script also reflects that h11 is connecting to the default webserver, and we deduce that this webserver originates from h31 in AS3 since the packets reveal that the eth4 (9.0.0.1) interface of R1 is interacting with the webserver.

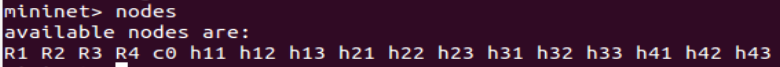
After the attack started, a TCP connection between 9.0.4.1 and 9.0.4.2 was established. Soon after, packets stop flowing to and from 9.0.0.1 and 13.0.0.1, and starts slowing to and from 9.0.4.1 and 13.0.0.1. The KEEPALIVE messages now show that there is a connection between 9.0.0.1 and 9.0.0.2, as well as 9.0.4.1 and 9.0.4.2. The website.sh script now reflects that h11 is connecting to the attacker web server, and we deduce that this webserver originates from h41 in AS4 since the packets reveal that the eth5 (9.0.4.1) interface of R1 is interacting with the webserver.

The fact that h11 contacts the webserver from AS4 instead of AS3 suggests that this path has a lower cost of traversal.

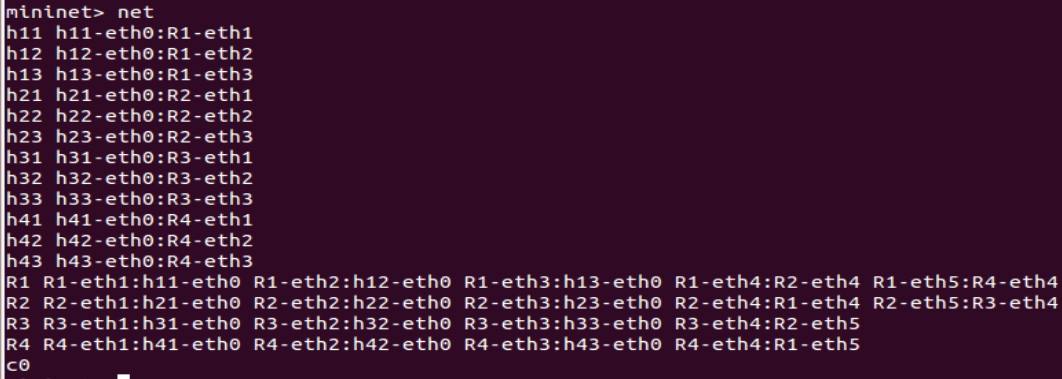
**Screenshots**

Topology

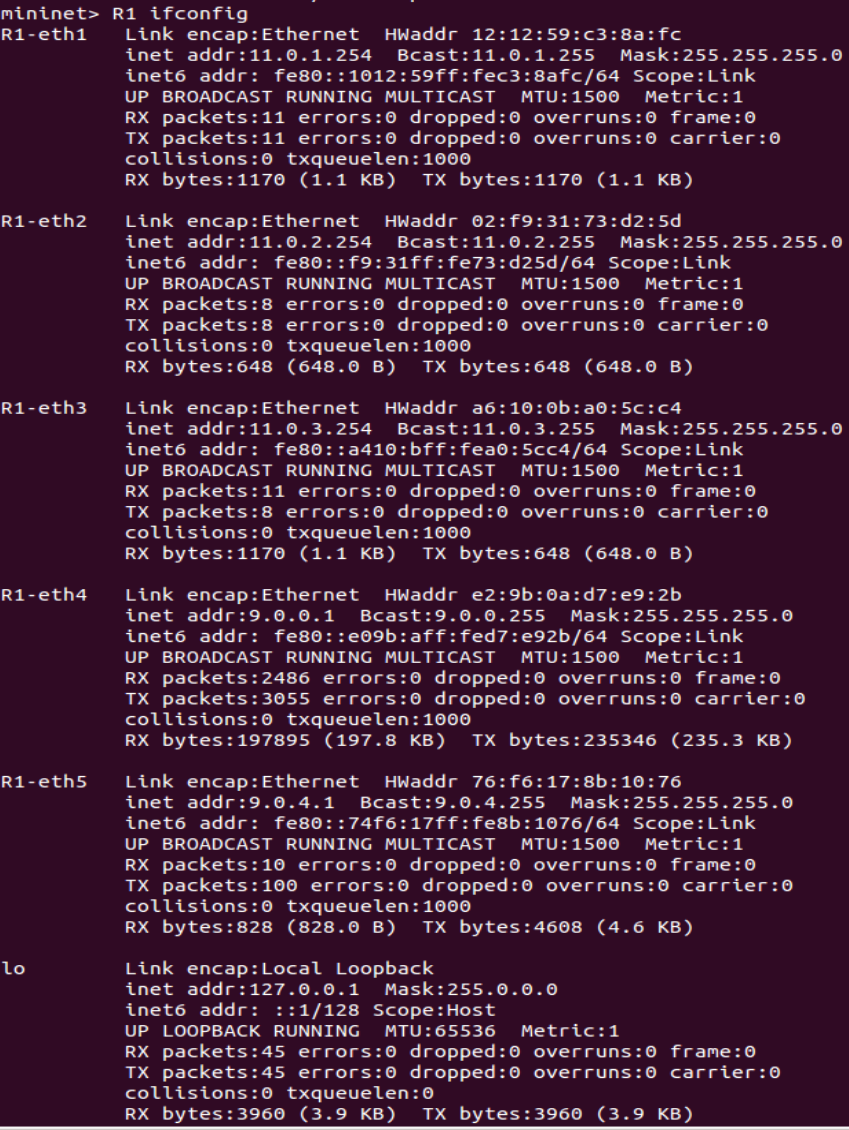
Nodes:

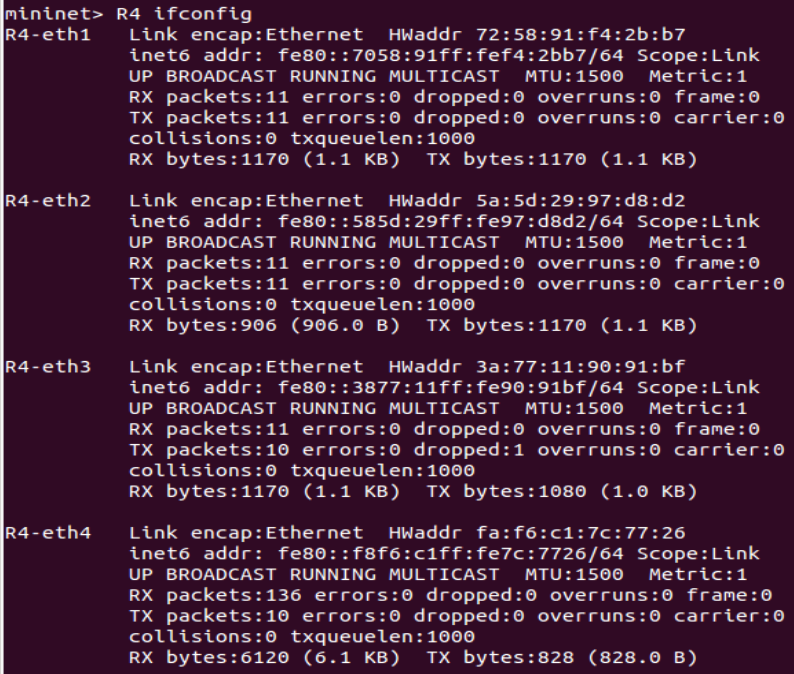
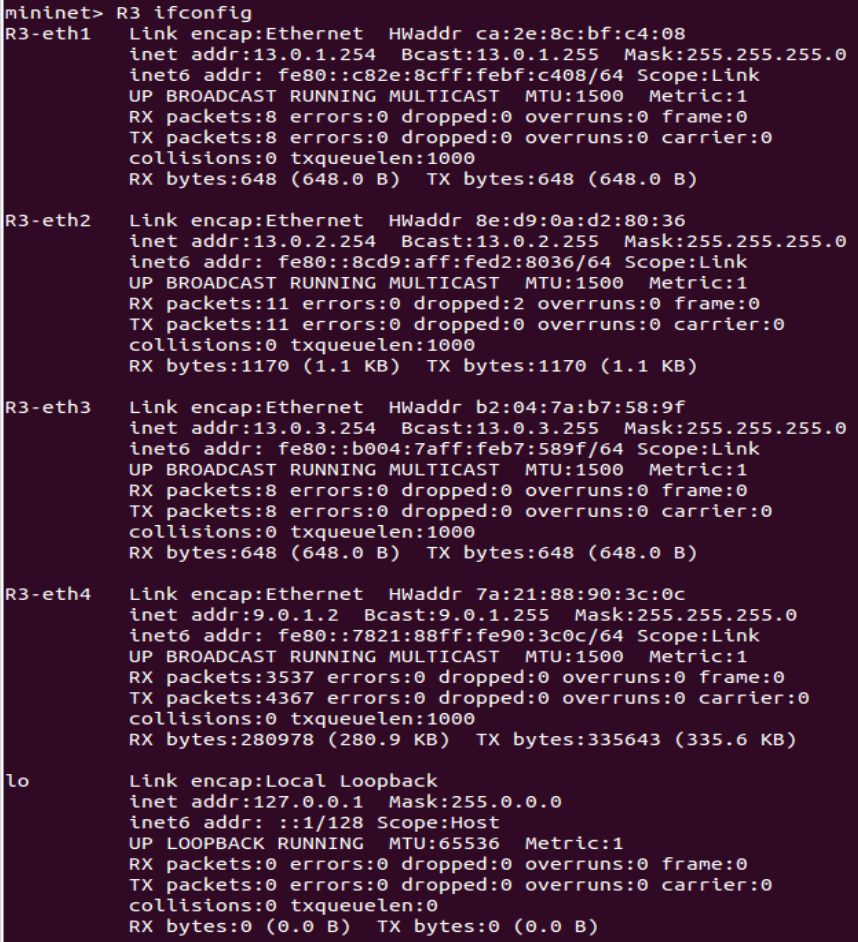


Net (network connections):

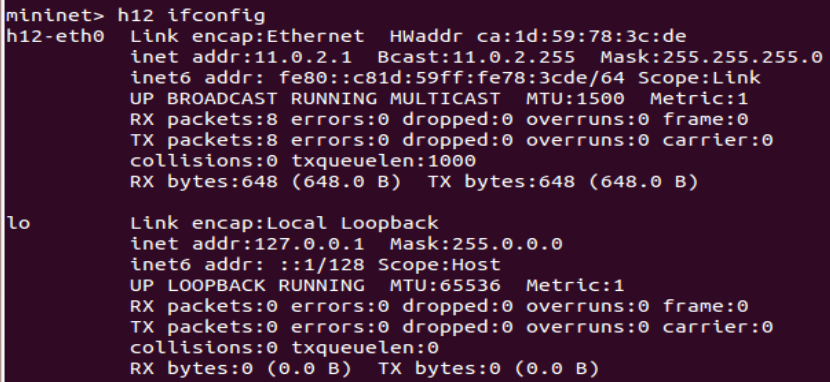
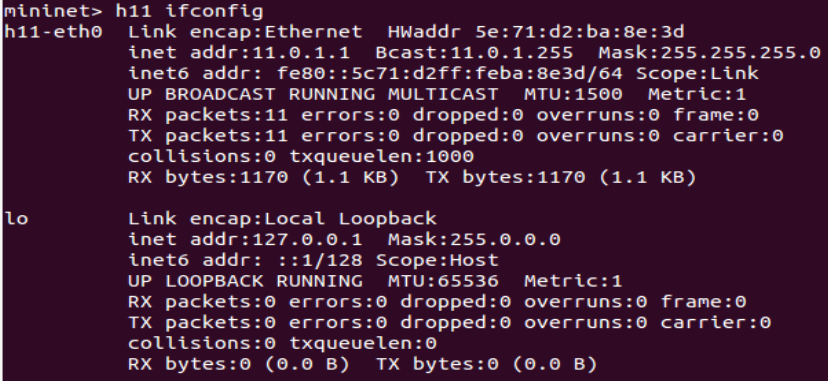


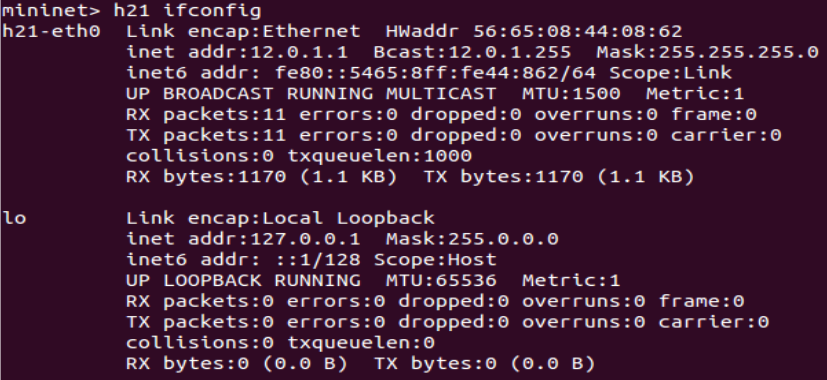
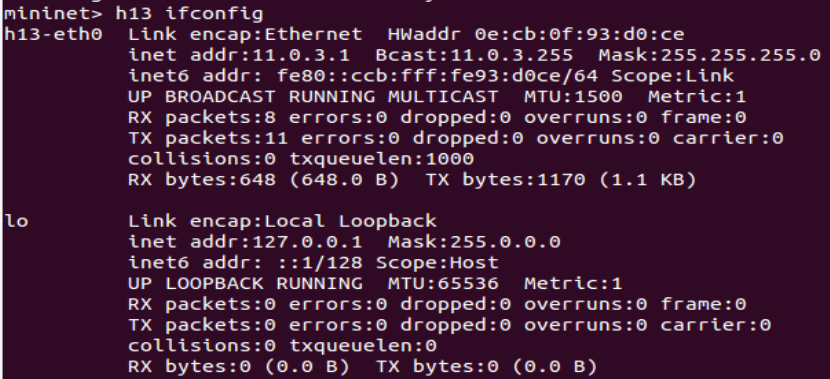
Ifconfig (for each router):

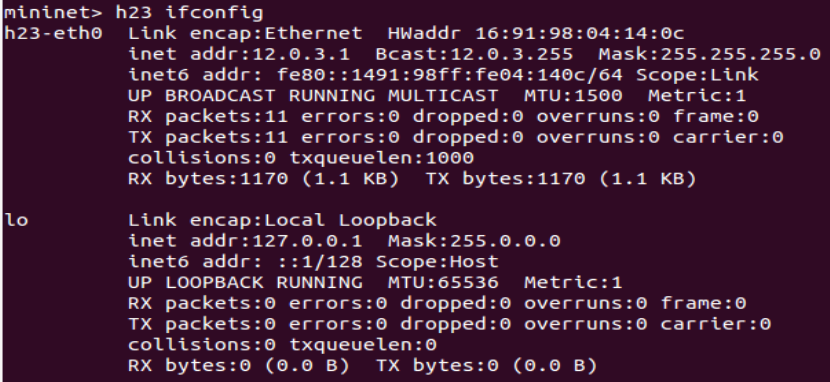
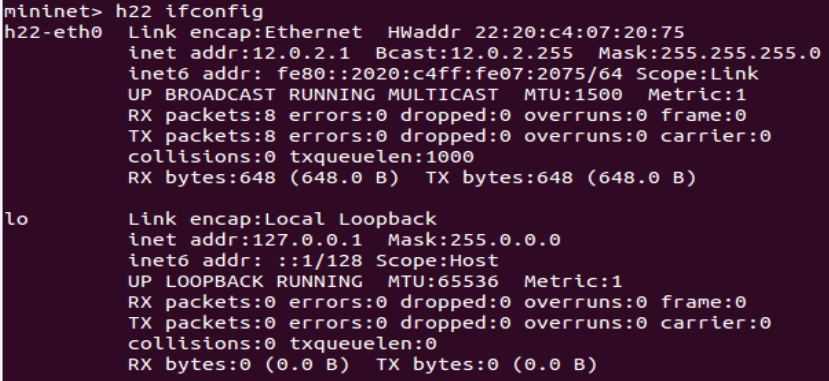


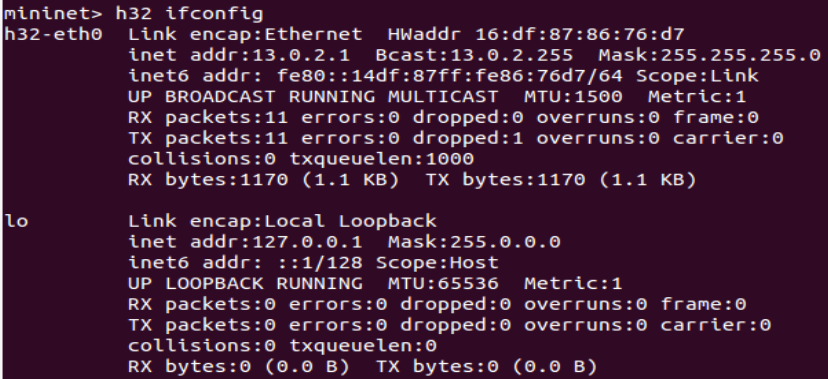
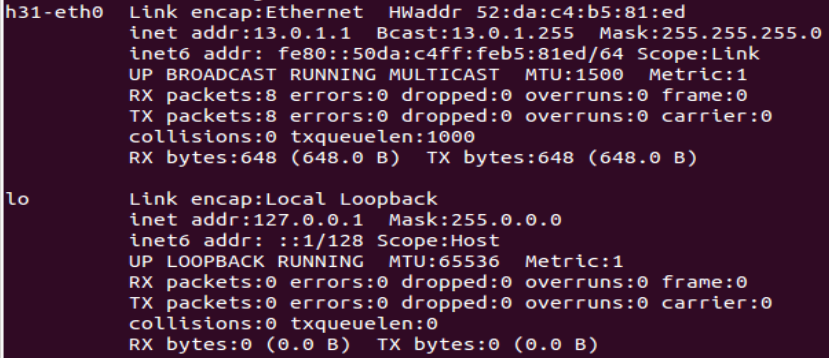


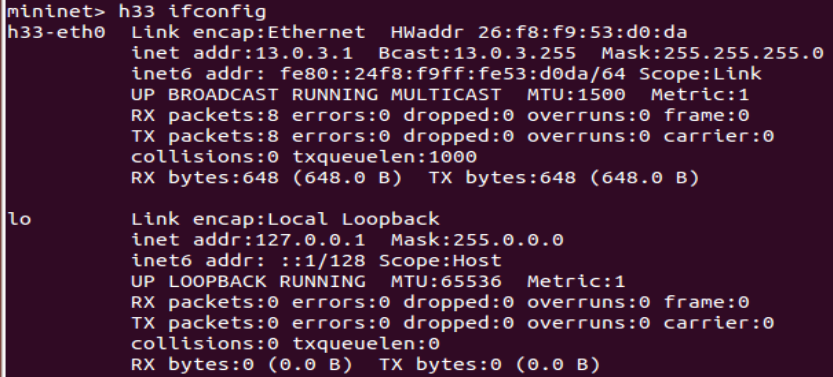
Ifconfig (for each node)





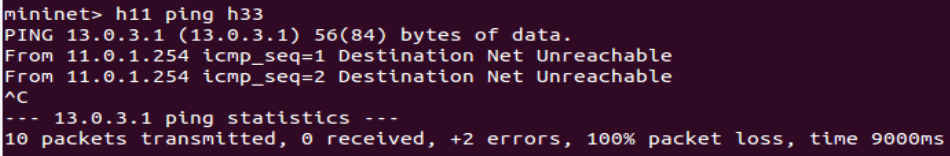






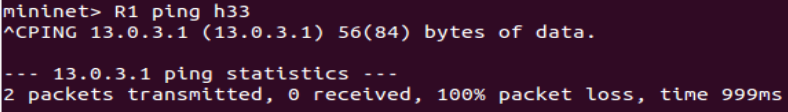
BGP Traffic

h11 pinging h33 (before modification):

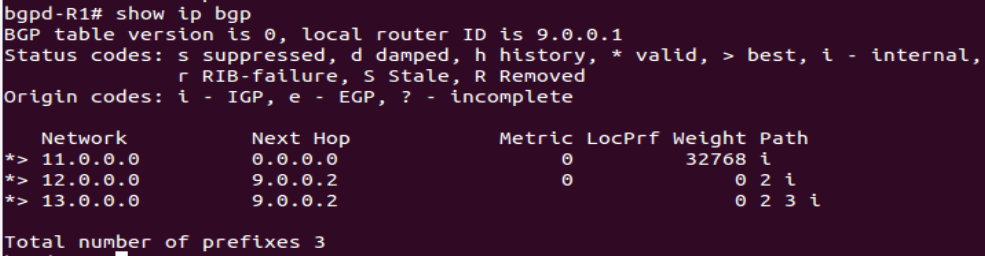




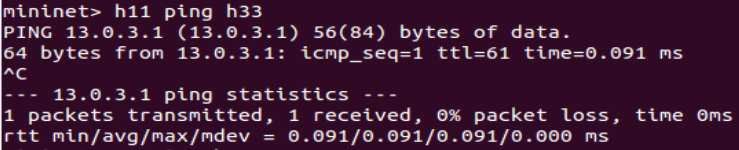
R1 pinging h33 (before modification):



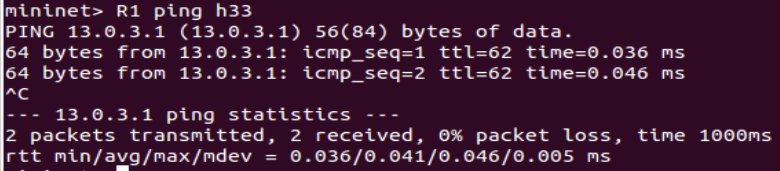
bgpd show ip bgp (before modification):



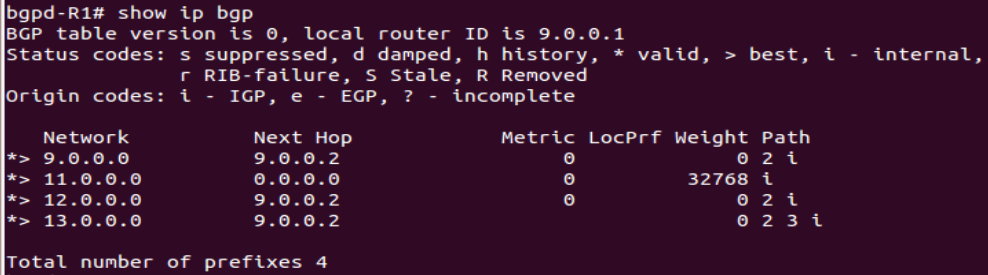
h11 pinging h33 (after modification):



R1 pinging h33 (after modification):

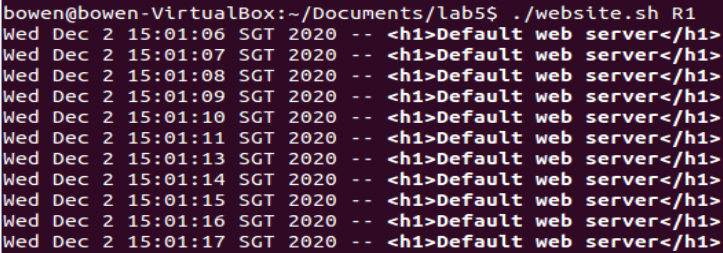


bgpd show ip bgp (after modification):



BGP Attack

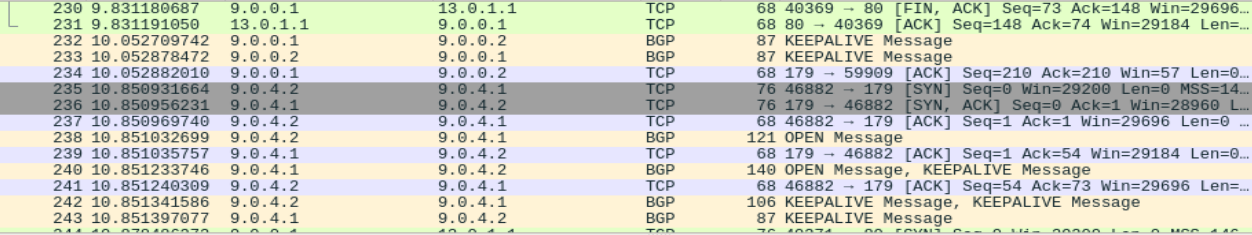
Connecting to the website (before the attack):



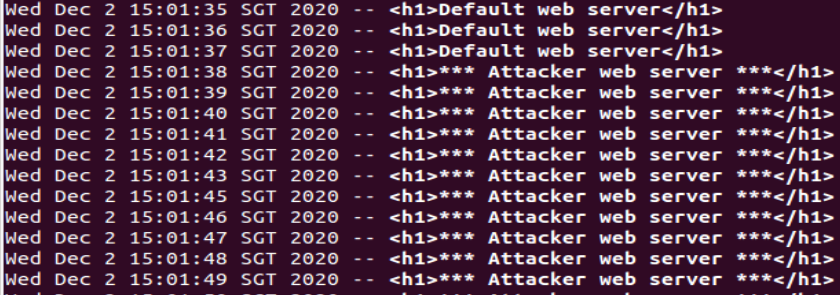
Packet capture on R1 (before the attack):



Packet capture on R1 (DURING the attack):



Connecting to the website (after the attack):



Packet capture on R1 (after the attack):

