Project Report

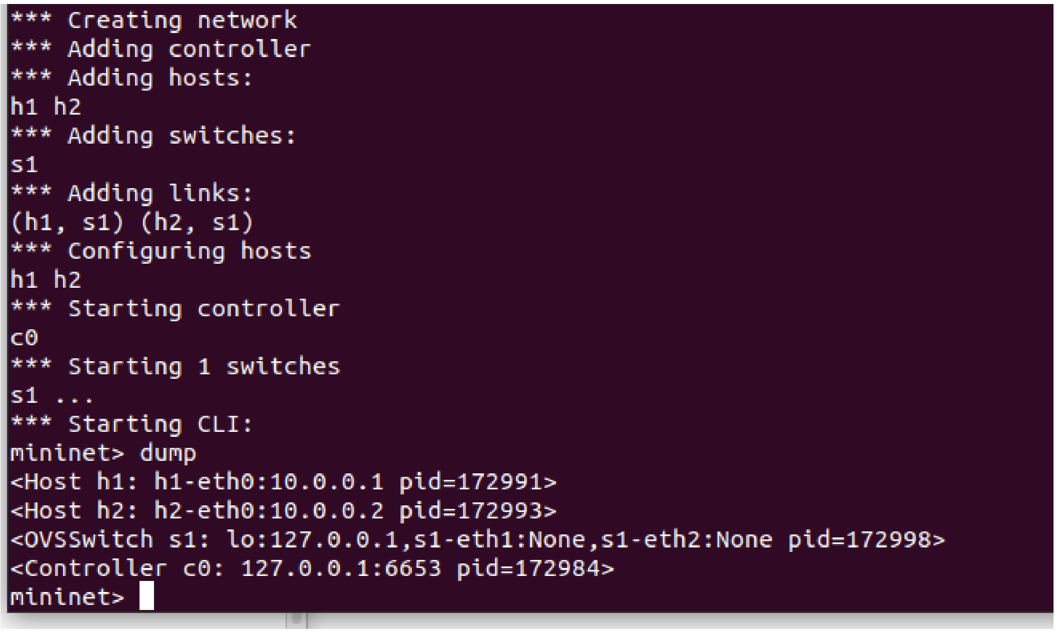
Hieu Mai

November 2020

This document explains about the creating virtualized network with data center topology and DDos attack on it. Project is done on Ubuntu 18. Mininet and python were installed as prerequisite.

1 Part A – Python code for data center network topology

For this topology, I used OVSKernelSwitch type 20 switches, 16 host with predefined Ips, added links between switches and hosts. I also added specific Ips for the switches. I install and ran the python code from terminal window. I called the dump command to see the connection details.



```
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=172991>
<Host h2: h2-eth0:10.0.0.2 pid=172993>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None pid=172998>
<Controller c0: 127.0.0.1:6653 pid=172984>
mininet> 
```

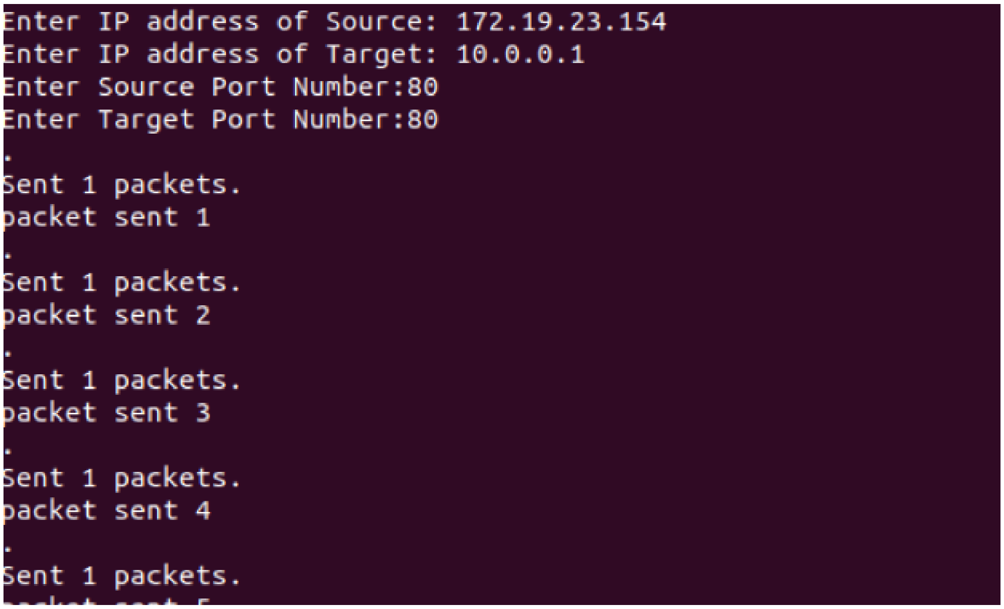
Figure 1: Start of mininet application

2 Part B - DDos attack

Installed Scapy by using following command:

sudo pip3 install scapy

I ran the code and entered source/system IP, target/host IP (I chose 10.0.0.1 Ip for DDos attack), I used both port as 80. I got permission error. I ran the code in sudo, it worked and start sending packets from source to target.



```
Enter IP address of Source: 172.19.23.154
Enter IP address of Target: 10.0.0.1
Enter Source Port Number:80
Enter Target Port Number:80
.
Sent 1 packets.
packet sent 1
.
Sent 1 packets.
packet sent 2
.
Sent 1 packets.
packet sent 3
.
Sent 1 packets.
packet sent 4
.
Sent 1 packets.
packet sent 5
```

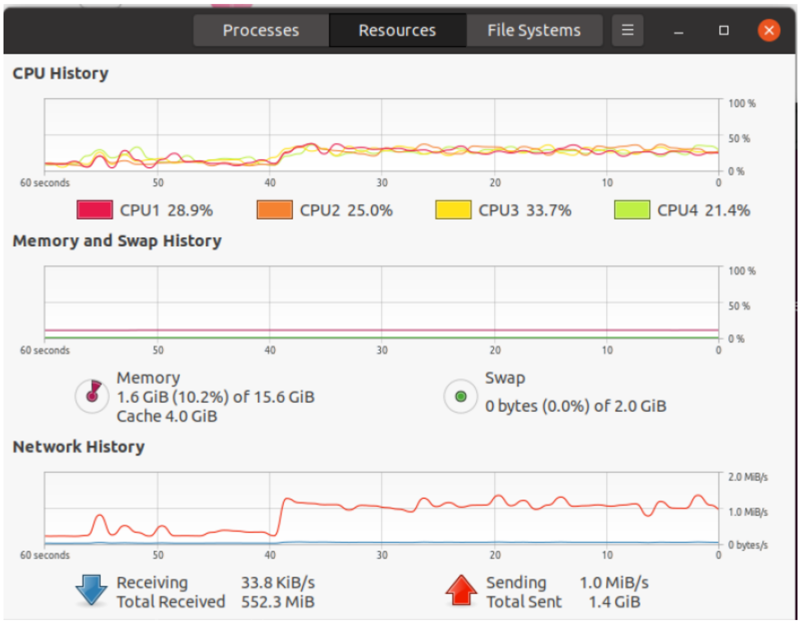
Figure 2: DDoS attack

3 Part C - Multiple Host DDos attack

I modified my code for part B and used it for configuration multiple sources. I added 'cpu=f' while adding all hosts and 'bw=10' while adding all links as asked in the assignment.

For single targeted host, it is observed that the data center topology with no bandwidth host limitation's network was full and not recovering.

While the data center topology with cpu and bandwidth limitation was not exceeding a certain amount of network.



CPU History

🟥 CPU1 28.9% 🟧 CPU2 25.0% 🟨 CPU3 33.7% 🟩 CPU4 21.4%

Memory and Swap History

Memory
1.6 GiB (10.2%) of 15.6 GiB
Cache 4.0 GiB

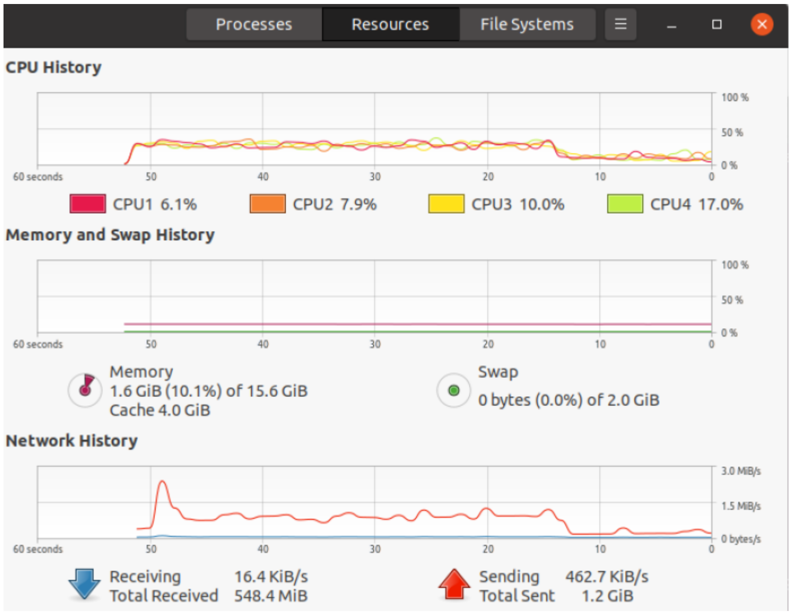
Swap
0 bytes (0.0%) of 2.0 GiB

Network History

Receiving 33.8 KiB/s
Total Received 552.3 MiB

Sending 1.0 MiB/s
Total Sent 1.4 GiB

Figure 3: Without limit network



CPU History

🟥 CPU1 6.1% 🟧 CPU2 7.9% 🟨 CPU3 10.0% 🟩 CPU4 17.0%

Memory and Swap History

Memory
1.6 GiB (10.1%) of 15.6 GiB
Cache 4.0 GiB

Swap
0 bytes (0.0%) of 2.0 GiB

Network History

Receiving 16.4 KiB/s
Total Received 548.4 MiB

Sending 462.7 KiB/s
Total Sent 1.2 GiB

Figure 4: With limit network