Project/Lab 3

Submitted by : Nandini Krupa Krishnamurthy

Task:

To launch man-in-the-middle attack in SDN and understand how attackers can steal information and to understand the importance of a secure communication between the controller and switch.

Experiment:

All the steps are explained along with the screenshots.

Step1 : Starting the experiment by creating a profile.

Step 1.1: Creating controller profile.

Figures starting from Fig1 to Fig12 show the steps needed to create a profile for controller and also for starting it.

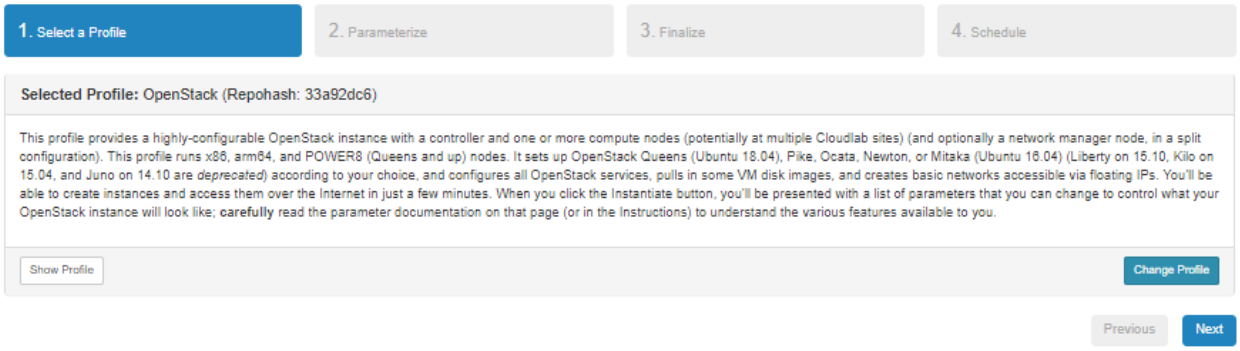

Fig 1 : Changing a profile for the experiment.

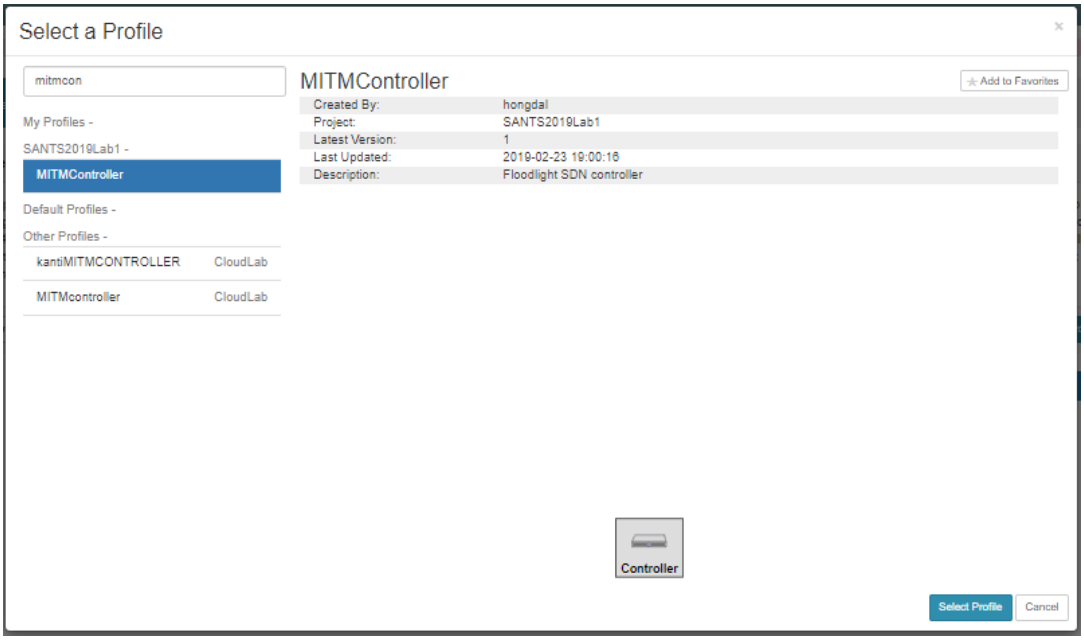

Fig 2: MITMController is necessary for this experiment.

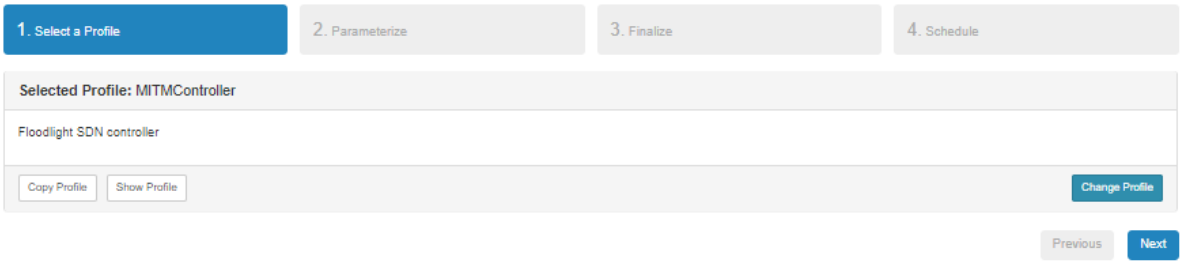

Fig 3: Selecting and changing a profile.

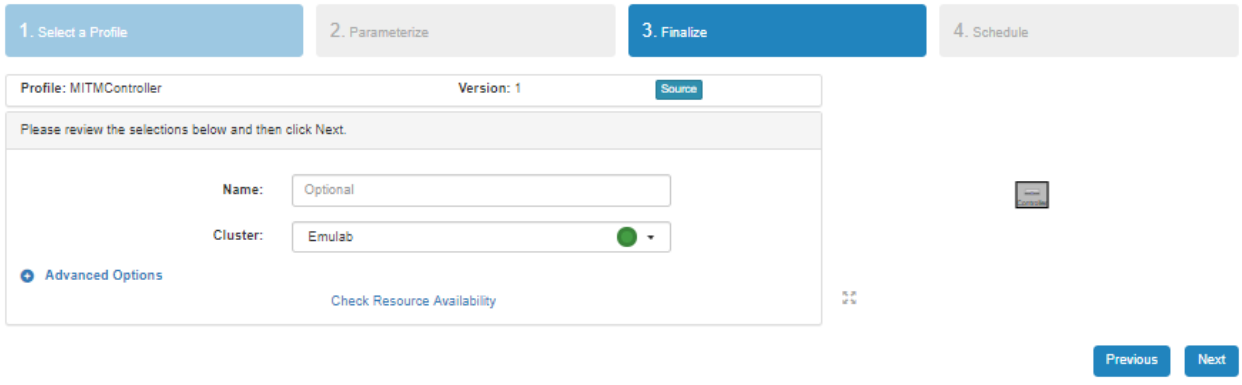

Fig 4: Selecting a Cluster (the one chosen here is Emulab).

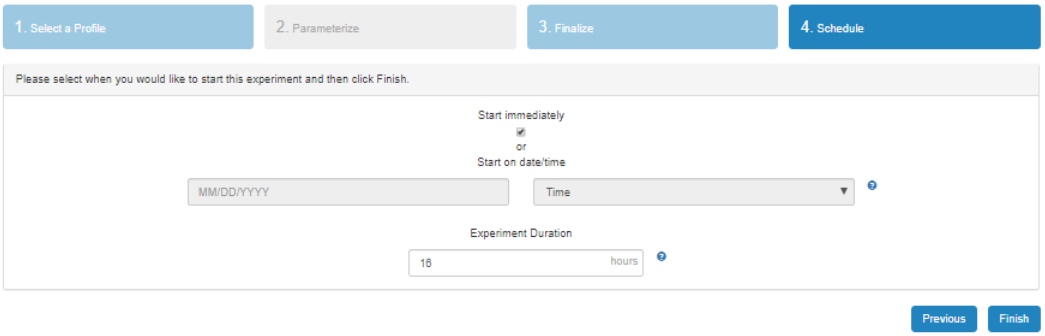

Fig 5: Selecting the duration of the experiment.

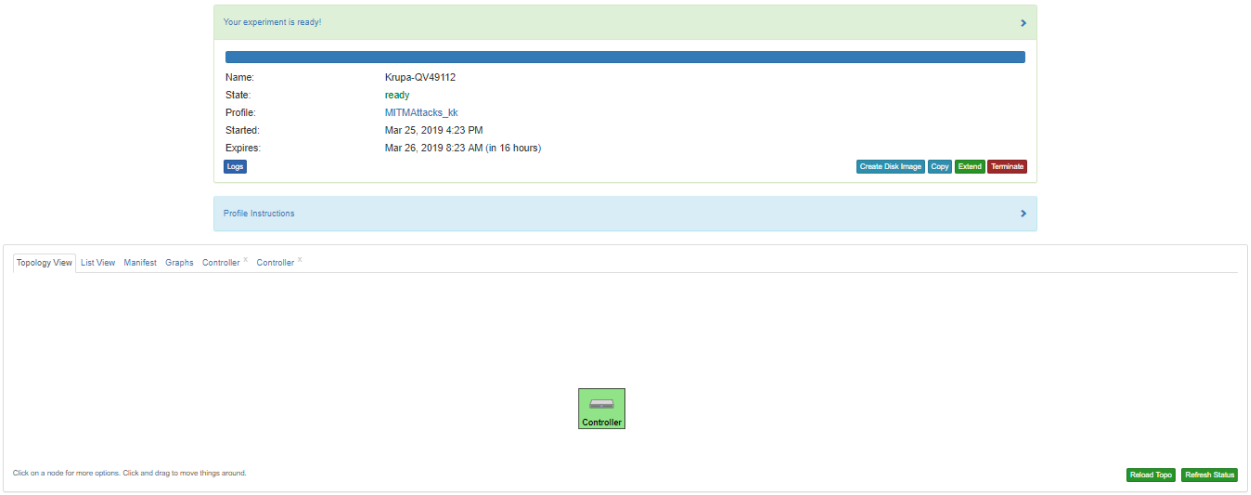

Fig 6: The controller is running successfully.

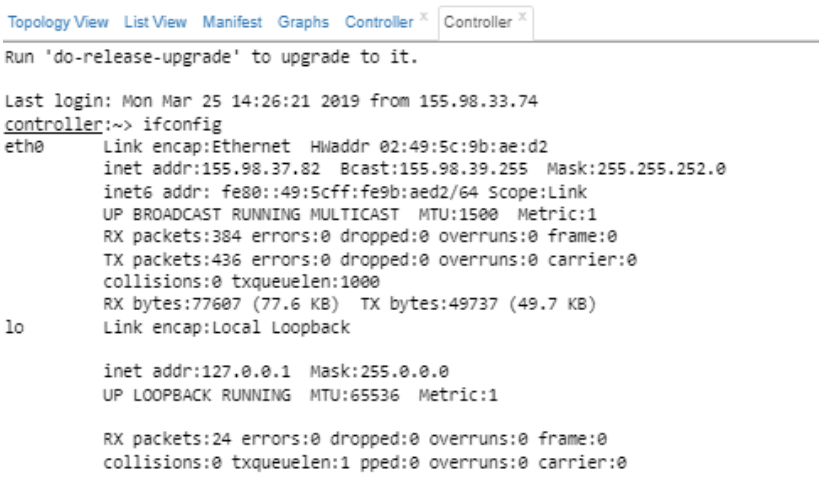

Fig 7: Finding IP of the controller using “ipconfig”.

Fig 8: Switching to "root" using "sudo su –".

Fig 9: Downloading floodlight using
"wget https://people.cs.clemson.edu/~hongdal/set_floodlight.sh ."

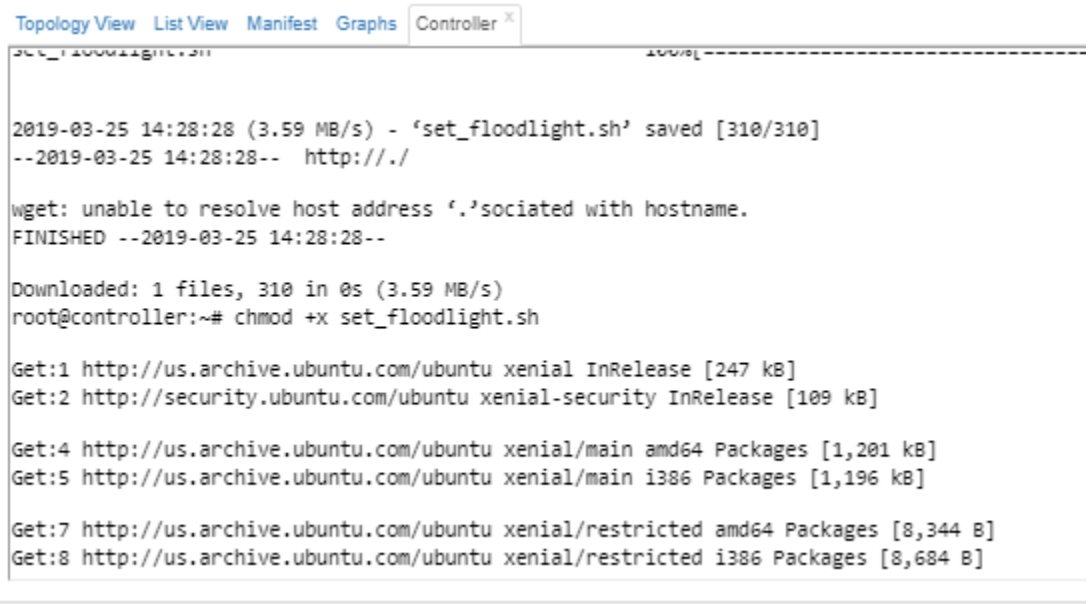

Fig 10: Using "chmod +x set_floodlight.sh" to change permission and then using "./set_floodlight.sh" to set it.

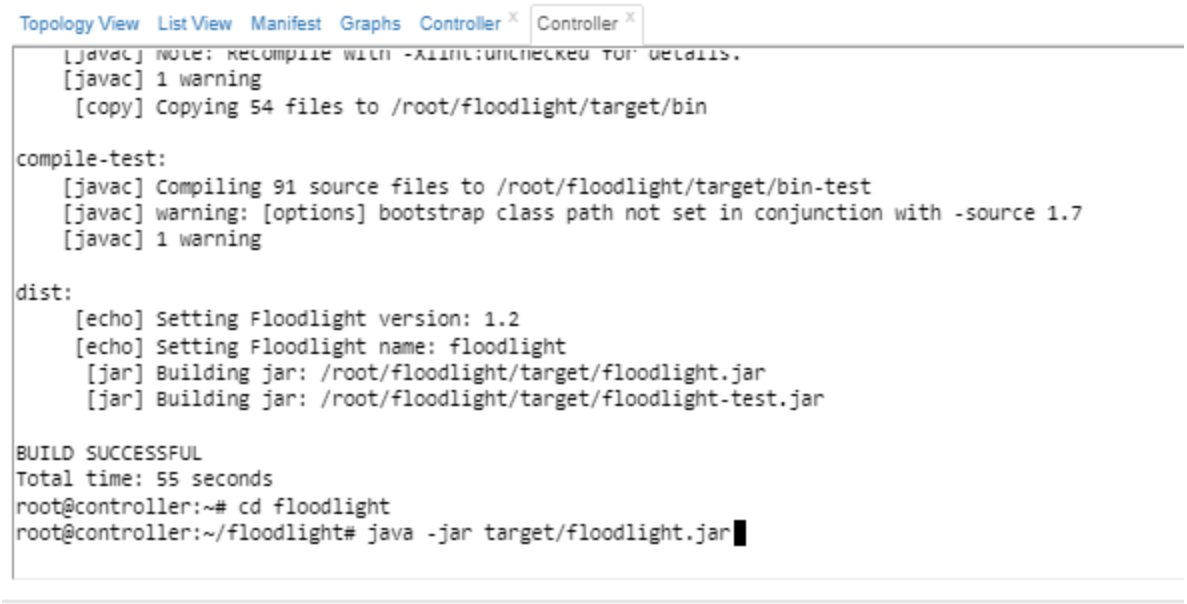

Fig 11: Changing directory to floodlight using "cd floodlight" and starting floodlight using "java -jar target/floodlight.jar".

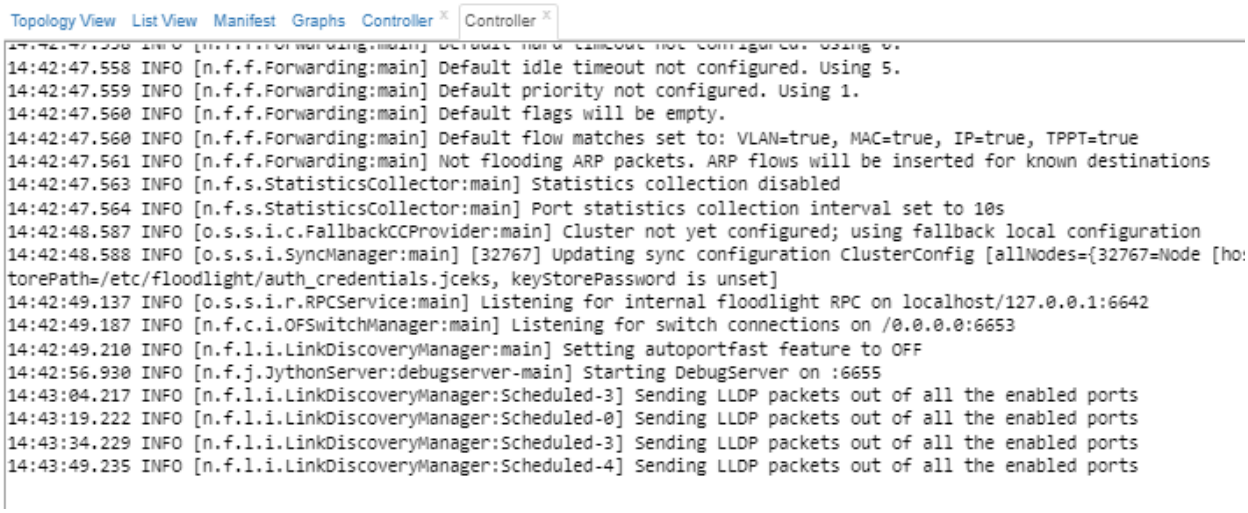

Fig 12: Controller has successfully started to send packets.

Step 1.2 – Creating profile for nodes needed for experiment.

Figures from Fig 13 to Fig 18 show the steps needed to create a profile for nodes and for configuring it with the controller.

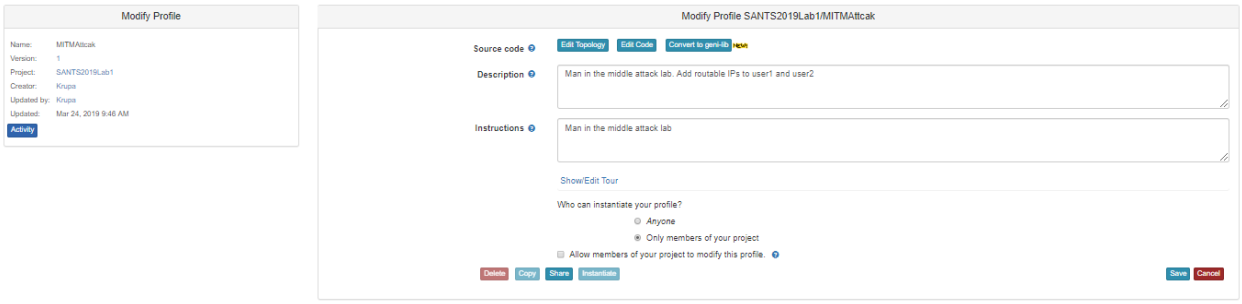

Fig 13: Adding the topology downloaded from https://people.cs.clemson.edu/~hongdal/mitm-attack.xml into the new profile here by using "Edit Topology" button.

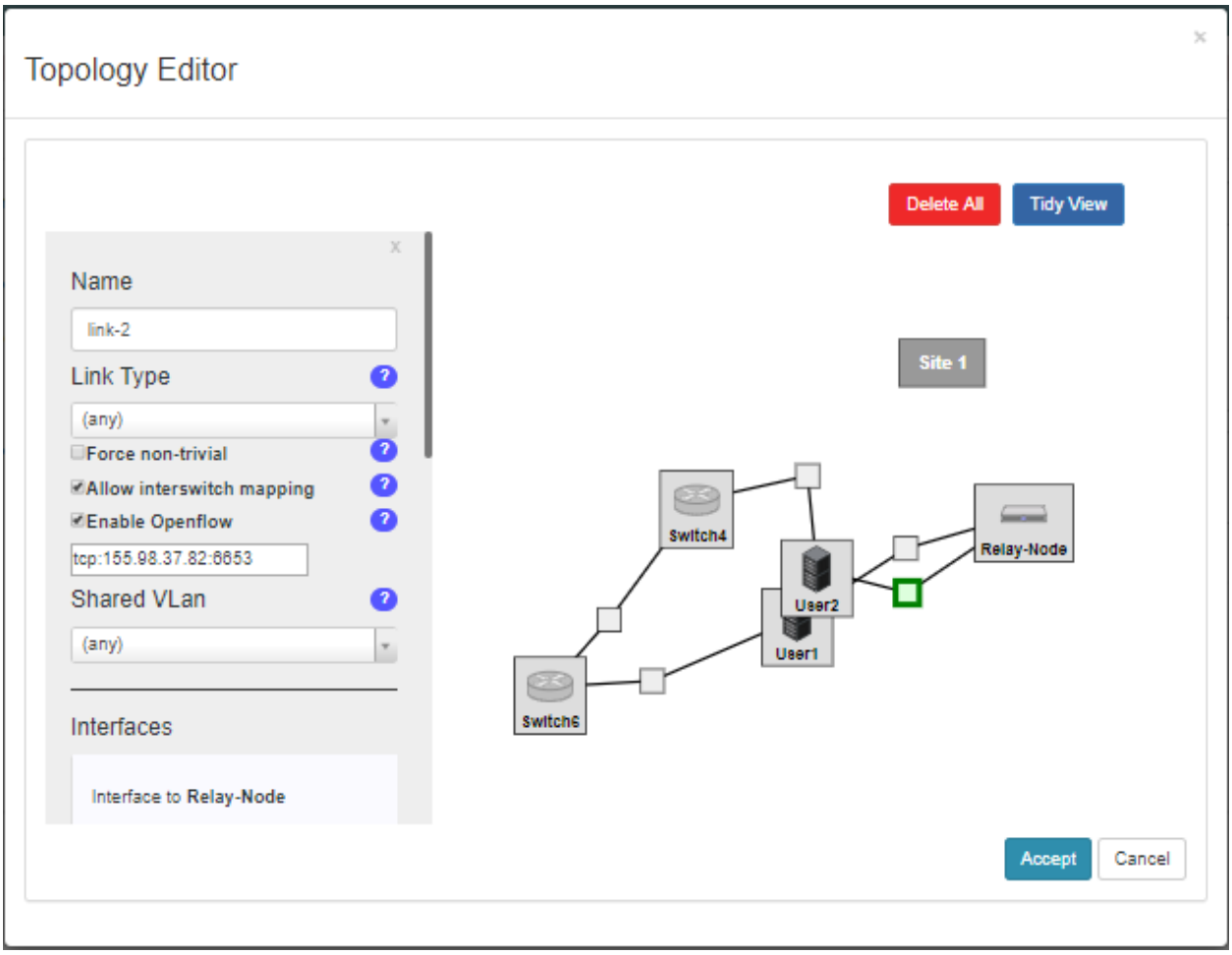

Fig 14: IP address of the controller is added to link2. Here IP of the controller is : 155.98.37.82.

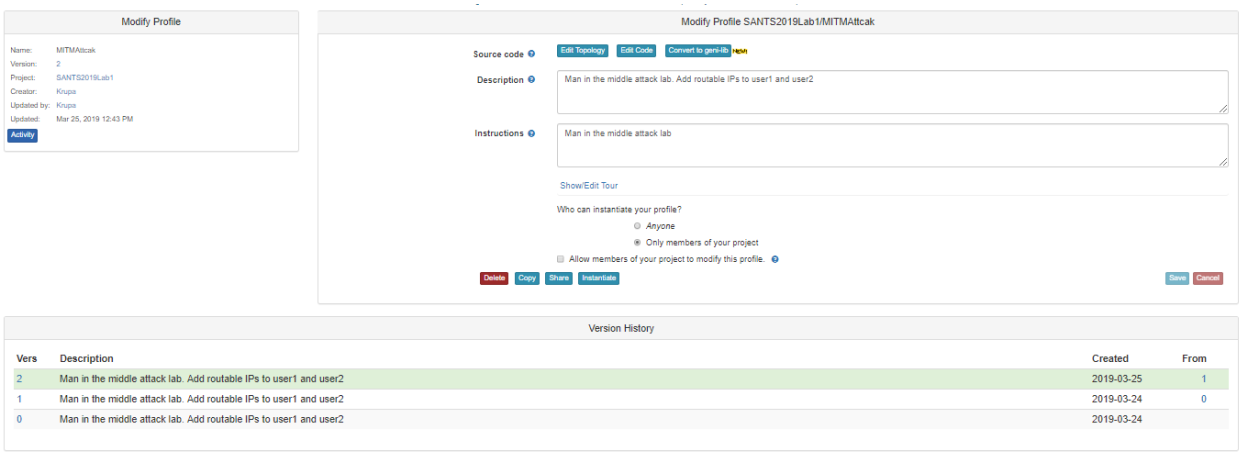

Fig 15: Saving the profile and instantiating it.

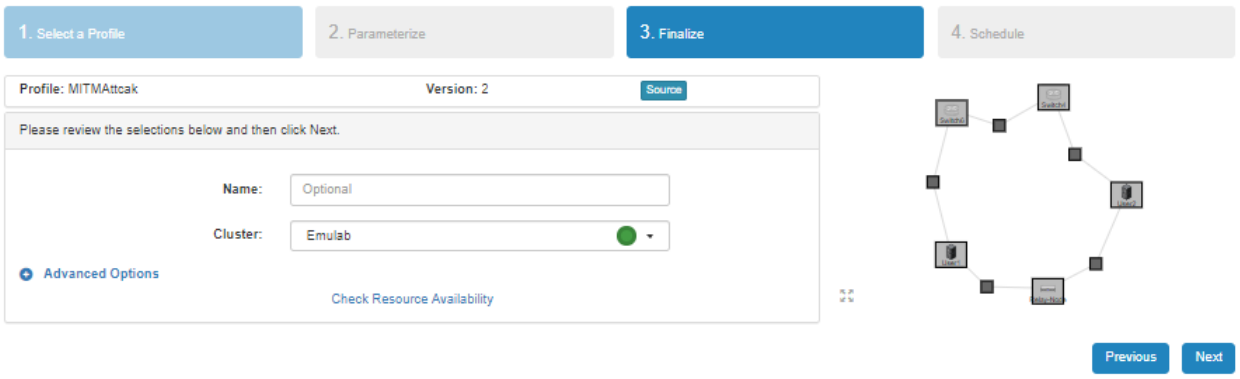

Fig16: Selecting a Cluster (the one chosen here is Emulab).

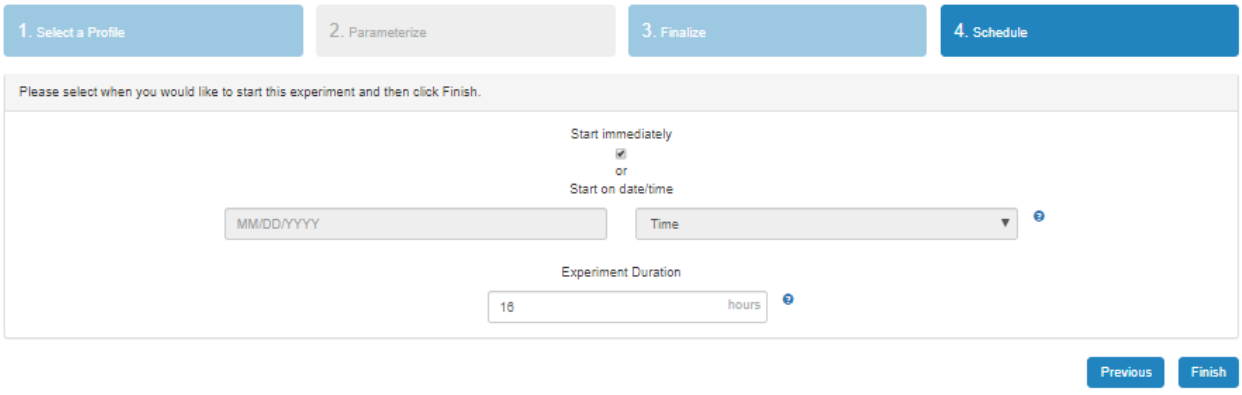

Fig17: Selecting the duration of the experiment.

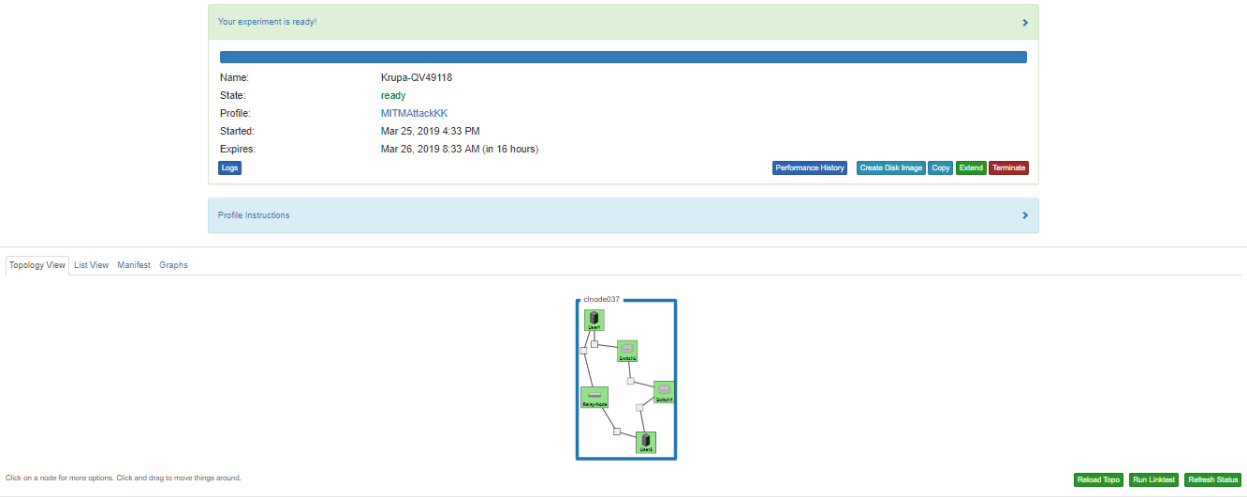

Fig 18: Profile is successfully created.

Step 2: Starting the Experiment.

Figures starting from Fig 19 to Fig 34 shows the steps needed to start this experiment.

Step 2.1: Setting up OVS for User1.

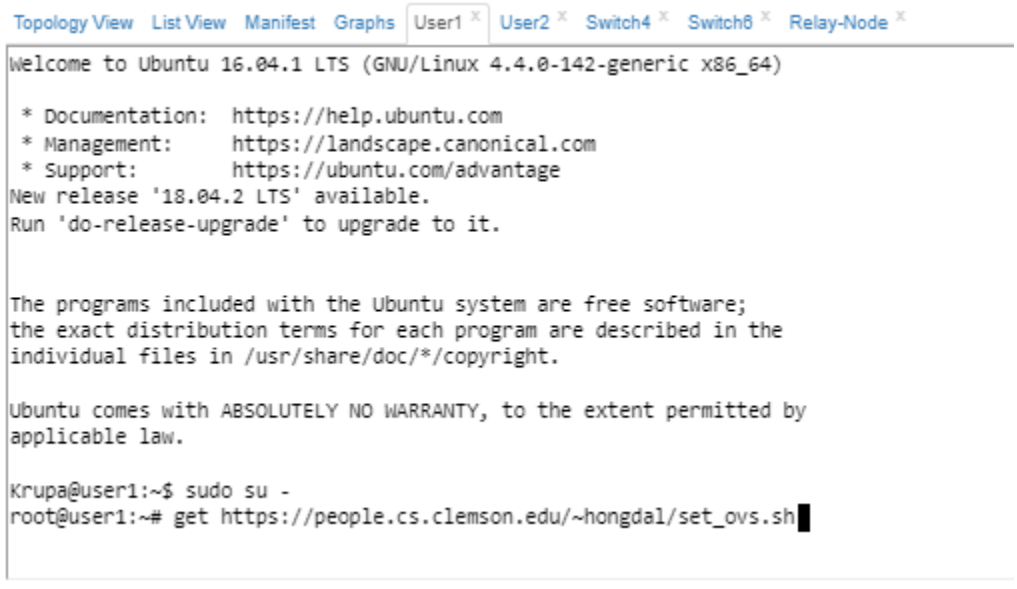

Fig 19: Switching to root using "sudo su –" and downloading script - set_ovs.sh using "wget https://people.cs.clemson.edu/~hongdal/set_ovs.sh" for User1.

Fig 20: Script is downloaded successfully.

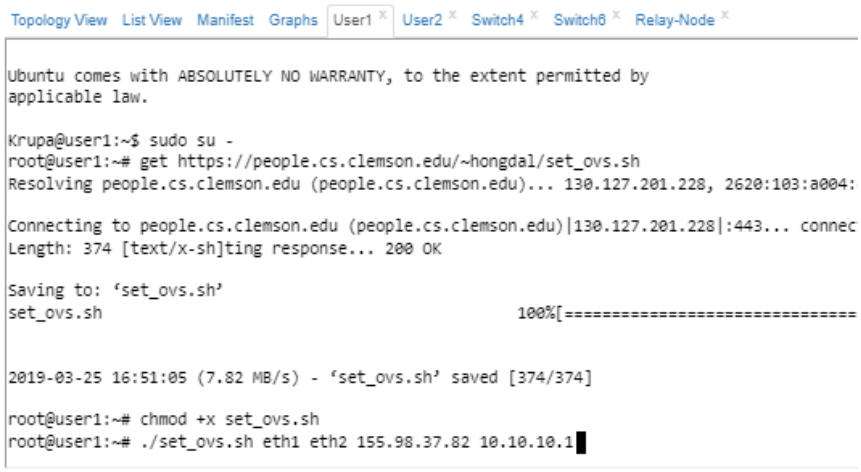

Fig 21: Changing permissions using "chmod +x set_ovs.sh" and setting the User1 with the controller using "./set_ovs.sh eth1 eth2 155.98.37.82 10.10.10.1". Here 10.10.10.1 is the experimental IP we have provided for User1.

Fig 22: openswitch network is successfully established.

Step 2.2: Setting up OVS for User1.

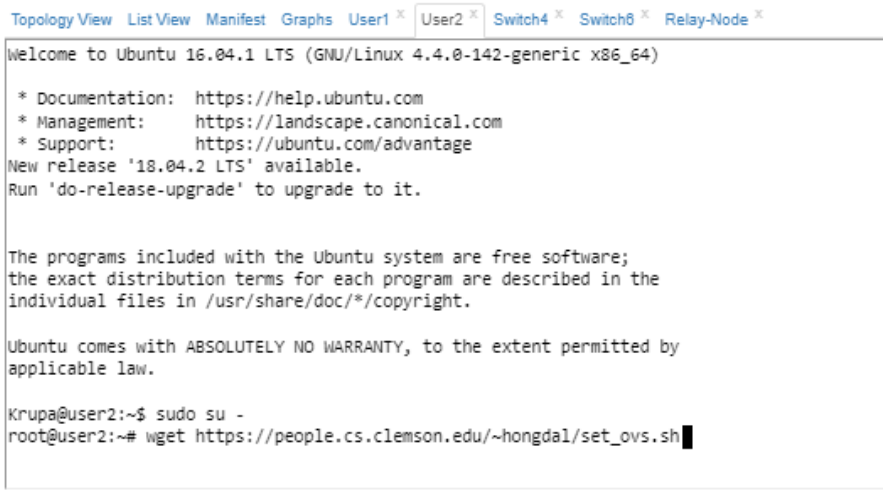

Fig 23: Switching to root using "sudo su –" and downloading script - set_ovs.sh using "wget https://people.cs.clemson.edu/~hongdal/set_ovs.sh" for User2.

Fig 24: Script is downloaded successfully.

Fig 25: Changing permissions using "chmod +x set_ovs.sh" and setting the User1 with the controller using "./set_ovs.sh eth1 eth2 155.98.37.82 10.10.10.2". Here 10.10.10.2 is the experimental IP we have provided for User2.

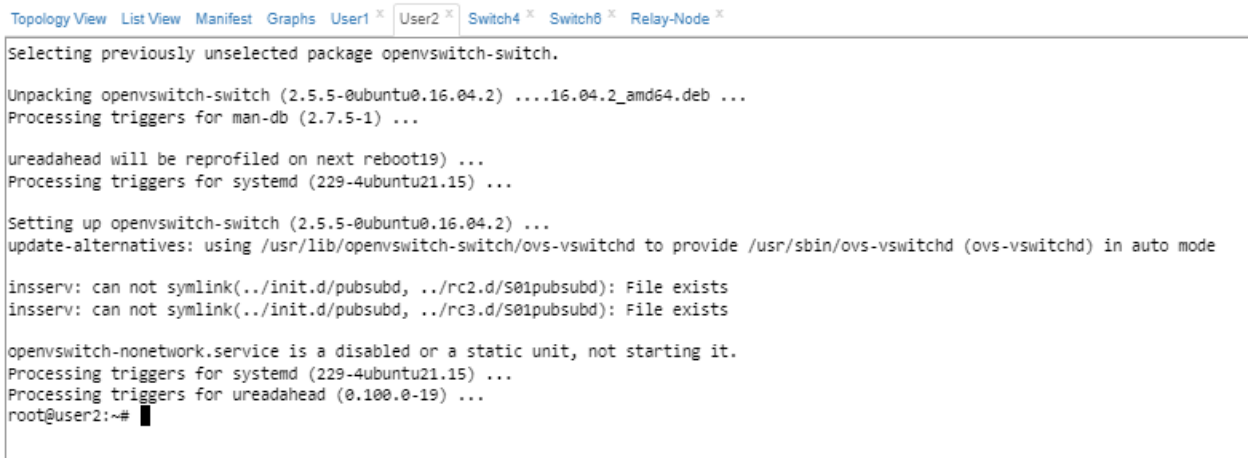

Fig 26: openswitch network is successfully established.

Step 2.3: Setting up OVS for Switch4

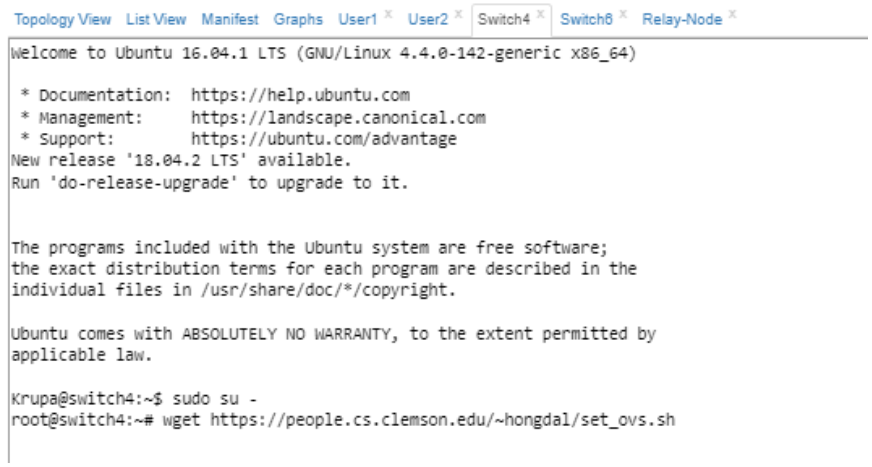

Fig 27: Switching to root using "sudo su –" and downloading script - set_ovs.sh using "wget https://people.cs.clemson.edu/~hongdal/set_ovs.sh" for Switch4.

Fig 28: Script is downloaded successfully.

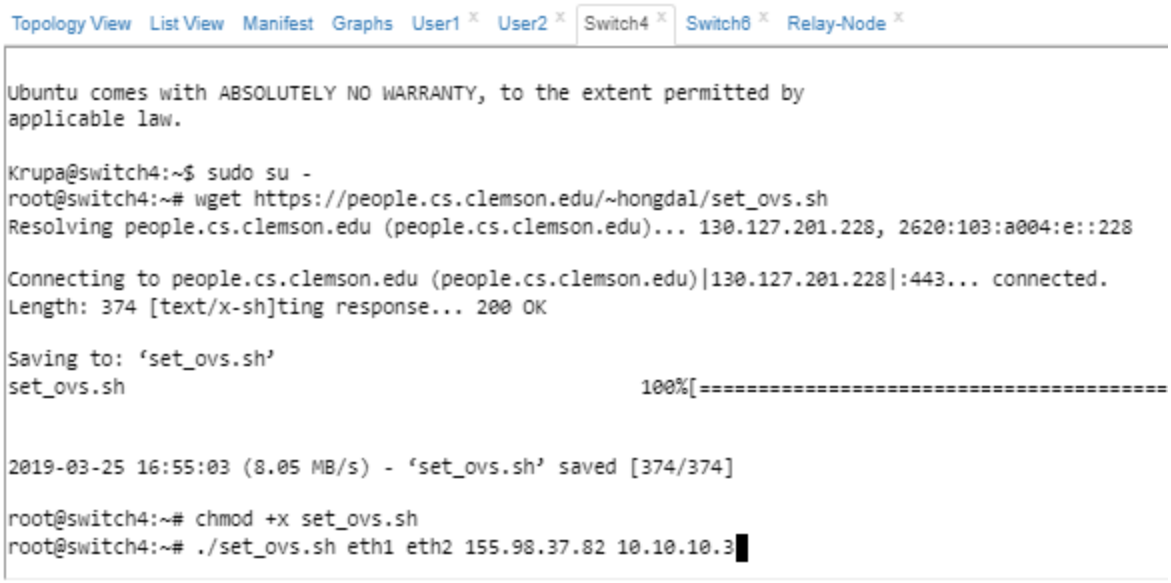

Fig 29: Changing permissions using "chmod +x set_ovs.sh" and setting the User1 with the controller using "./set_ovs.sh eth1 eth2 155.98.37.82 10.10.10.3". Here 10.10.10.3 is the experimental IP we have provided for Switch4.

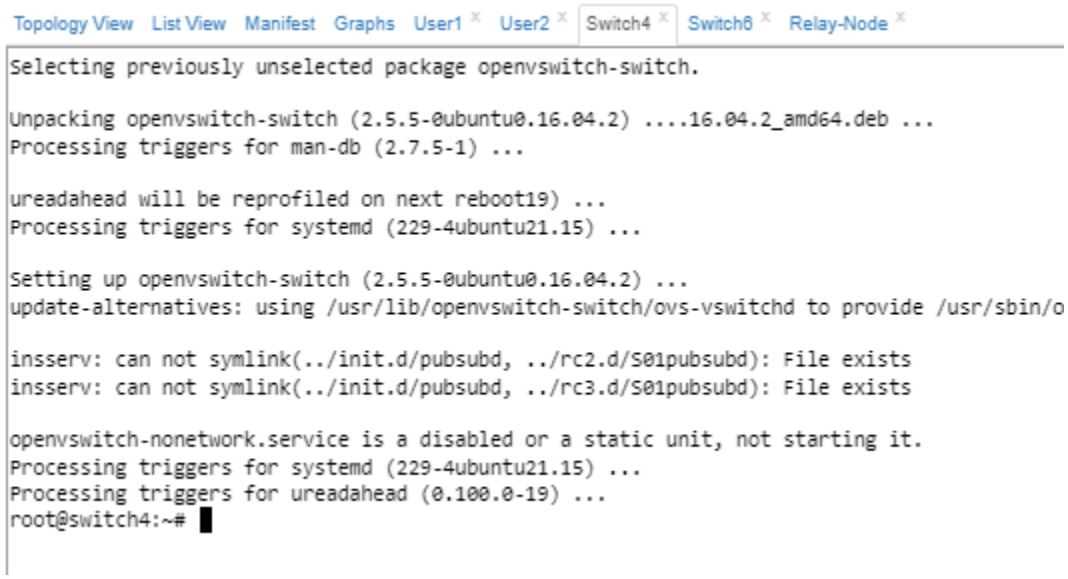

Fig 30: openswitch network is successfully established.

Step 2.4: Setting up OVS for Switch 6

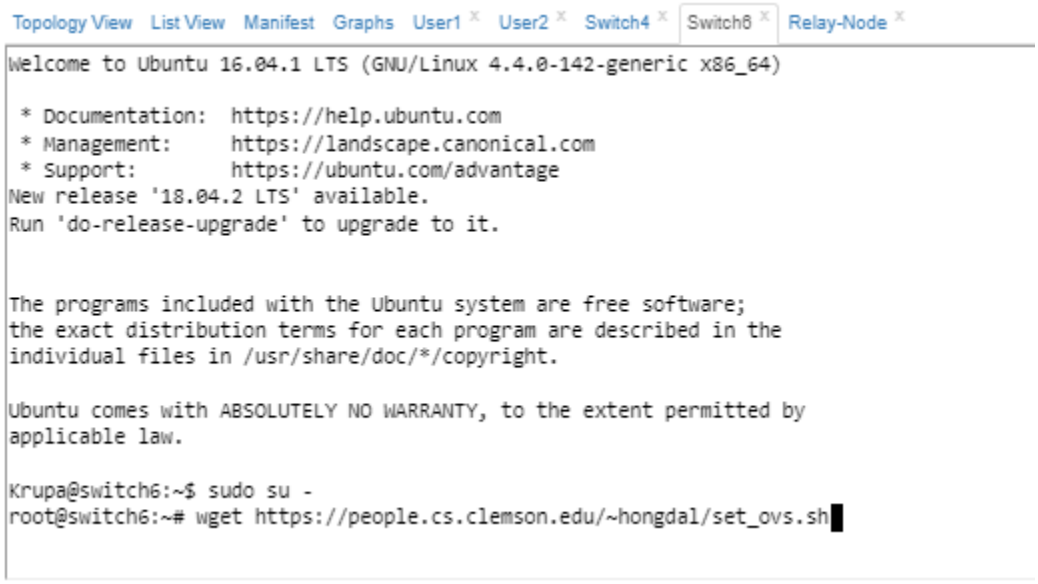

Fig 31: Switching to root using "sudo su –" and downloading script - set_ovs.sh using "wget https://people.cs.clemson.edu/~hongdal/set_ovs.sh" for Switch6.

Fig 32: Script is downloaded successfully.

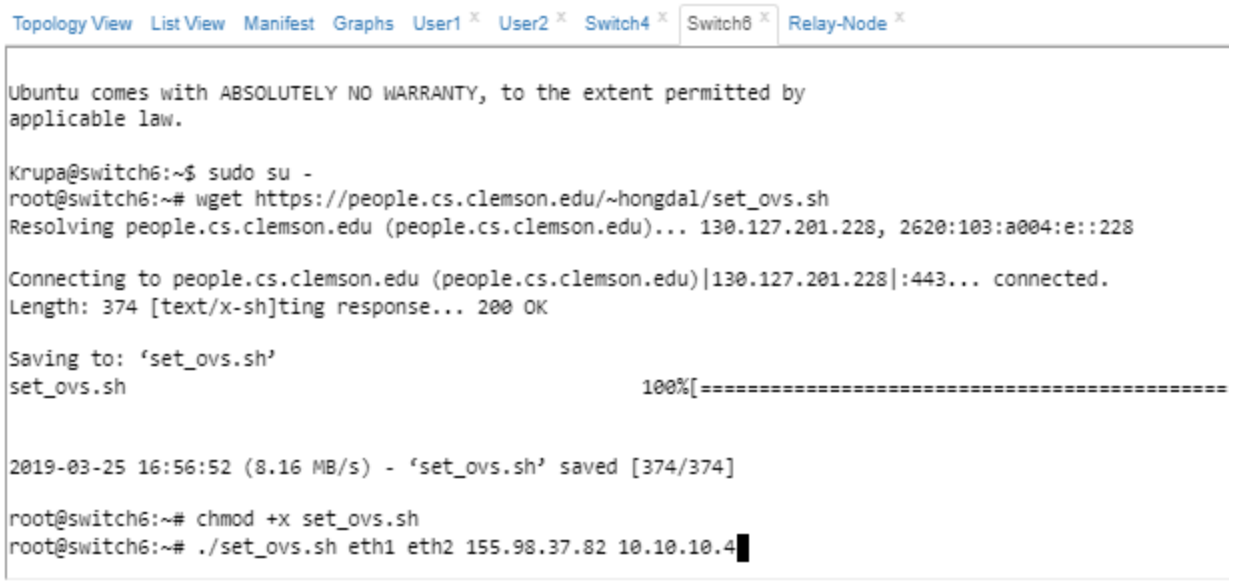

Fig 33: Changing permissions using "chmod +x set_ovs.sh" and setting the User1 with the controller using "./set_ovs.sh eth1 eth2 155.98.37.82 10.10.10.4". Here 10.10.10.4 is the experimental IP we have provided for Switch6.

Fig 34: openswitch network is successfully established.

Step 3: Conducting the Experiment (Normal Flow)

Figures starting from Fig 35 to Fig 43 show the normal flow of packets when there is no attack.

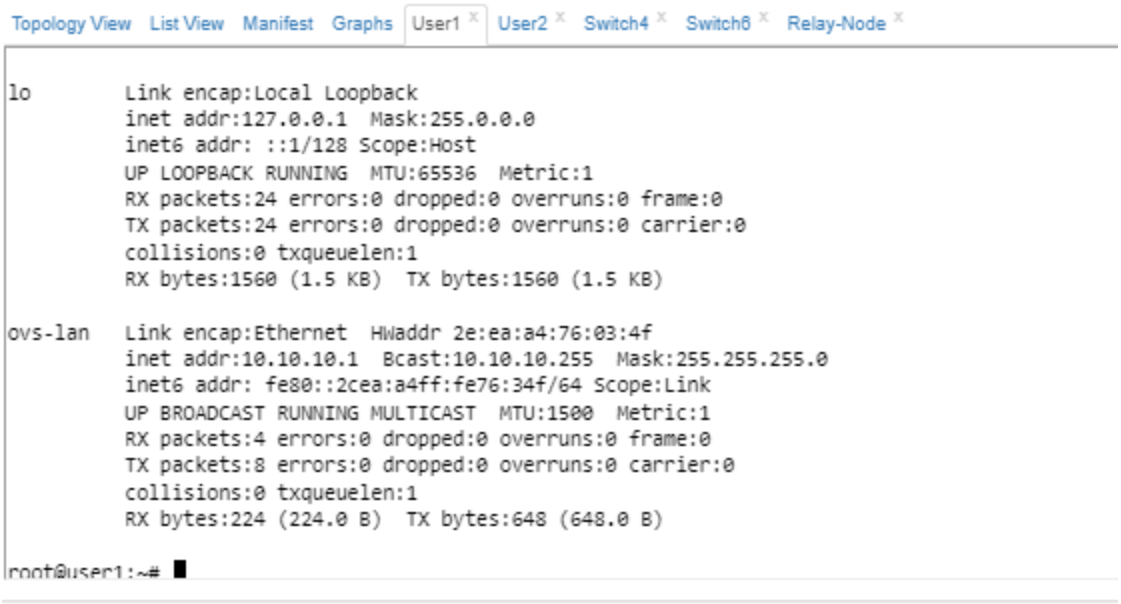

Fig 35: Experimental IP address of the User1: 10.10.10.1 using "ipconfig".

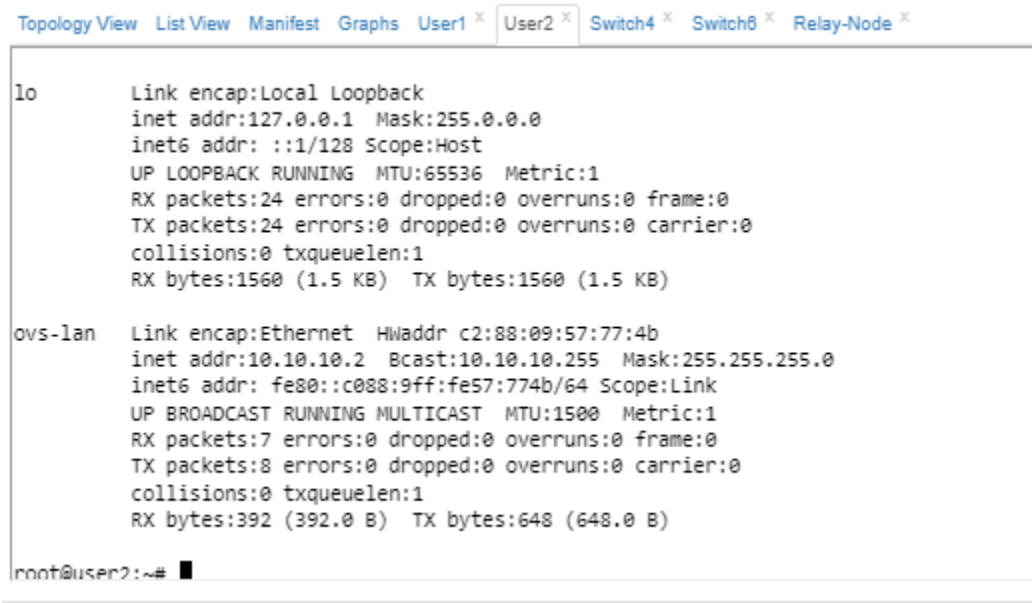

Fig 36: Experimental IP address of the User1: 10.10.10.2 using "ipconfig".

Fig 37: Pinging from User1 to User2 using “ping 10.10.10.2”.

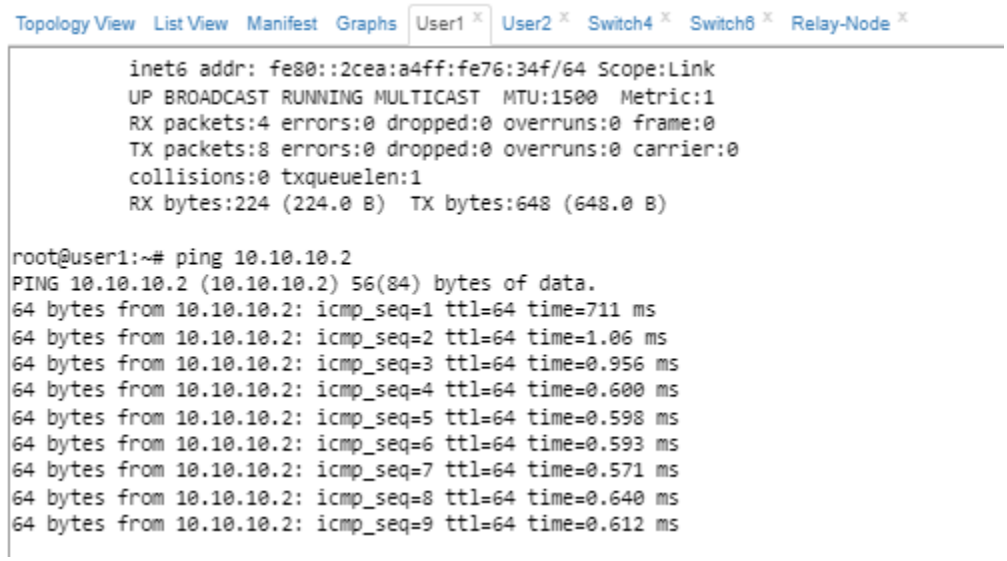

Fig 38: Ping successful.

Fig 39: Using "tcpdump -i eth1 -nq icmp" to view the packets through Switch4.

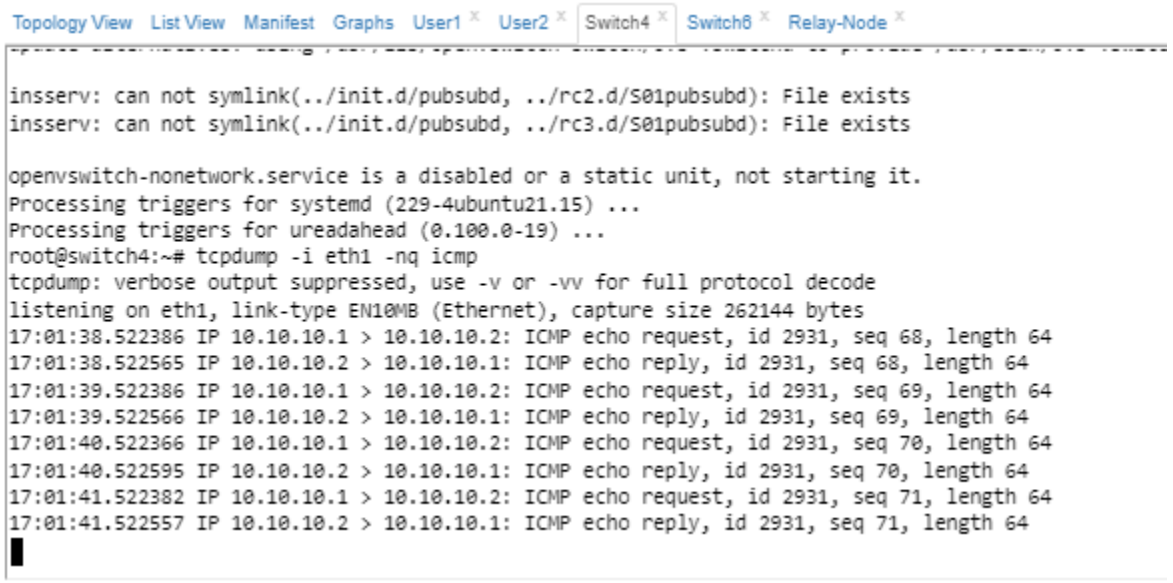

Fig 40: Packets are flowing through Switch4.

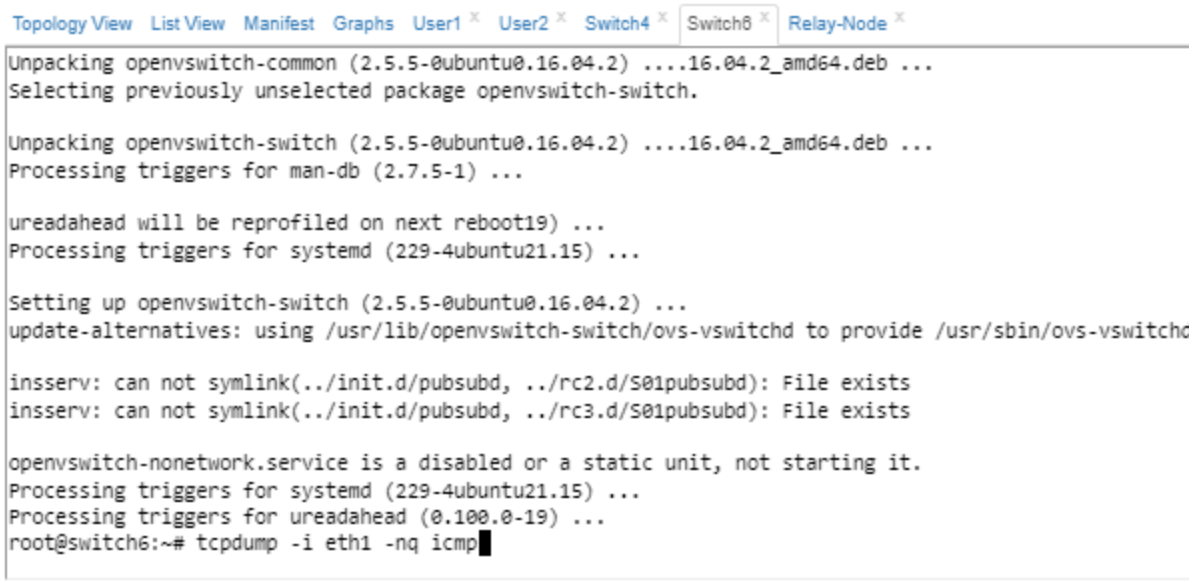

Fig 41: Using "tcpdump -i eth1 -nq icmp" to view the packets through Switch6.

Fig 42: Packets are flowing through Switch6.

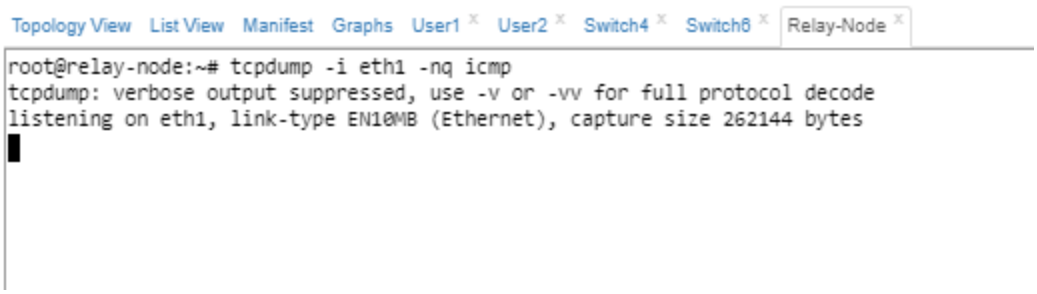

Fig 43: When "tcpdump -i eth1 -nq icmp" is used for Relay_Node, it is observed that no packet is flowing through it.

Step 4 – Conduct Man-In-Middle-Attack.

Figures starting from Fig 44 to Fig 53 show the flow of traffic in case of an MITM attack.

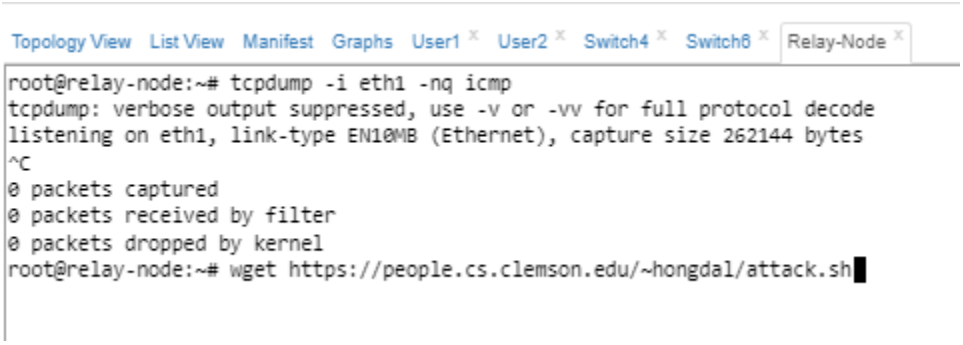

Fig 44: Downloading attack script using "wget https://people.cs.clemson.edu/~hongdal/attack.sh".

Fig 45: Script is successfully downloaded.

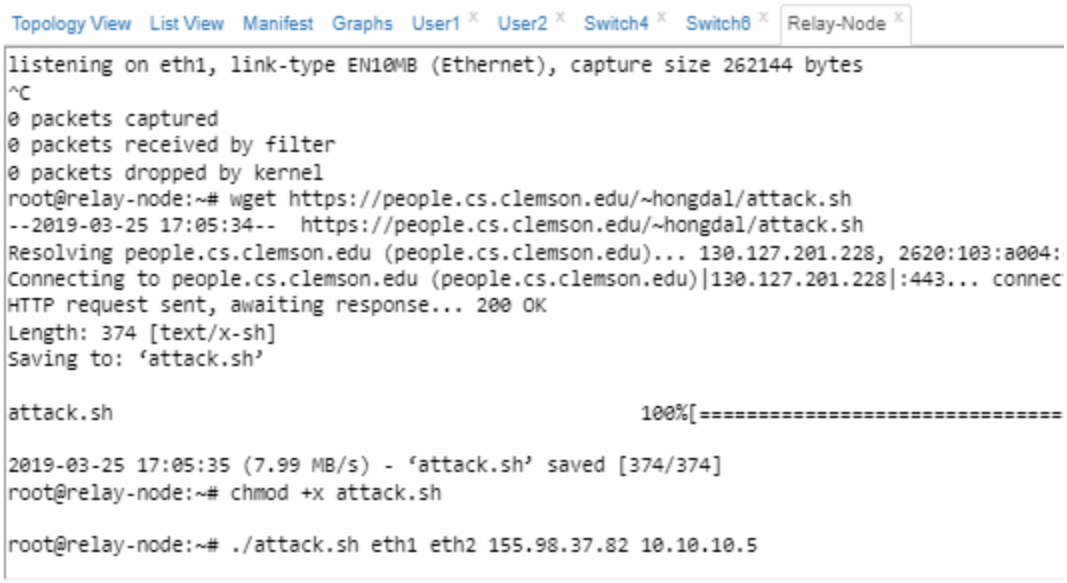

Fig 46: Changing the permissions using " chmod +x attack.sh" and starting the attack by running the command " ./ attack.sh eth1 eth2 155.98.37.82 10.10.10.5". Here 155.98.37.82 denotes the controller IP and 10.10.10.5 denotes the Relay-Node IP.

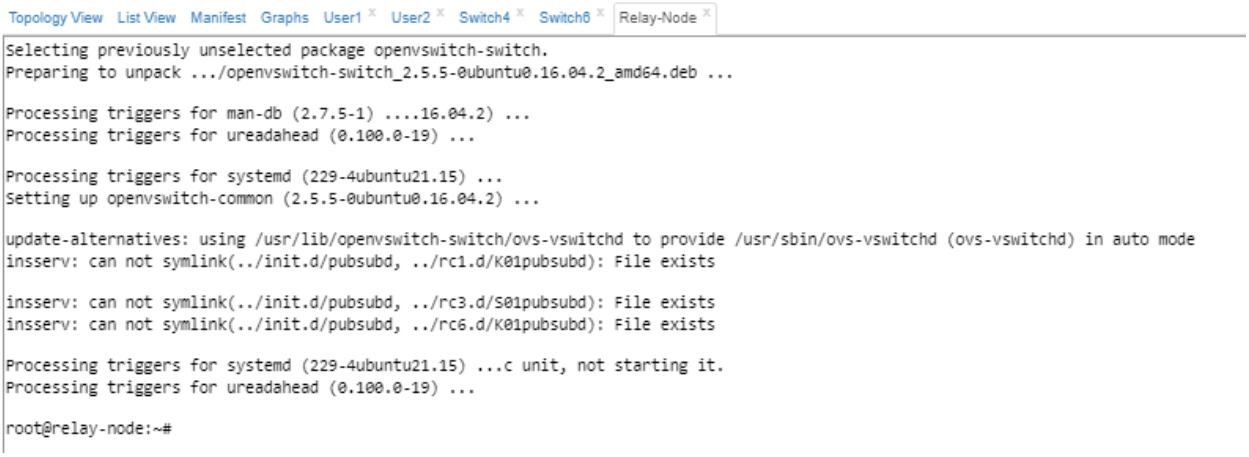

Fig 47: Attack is initiated.

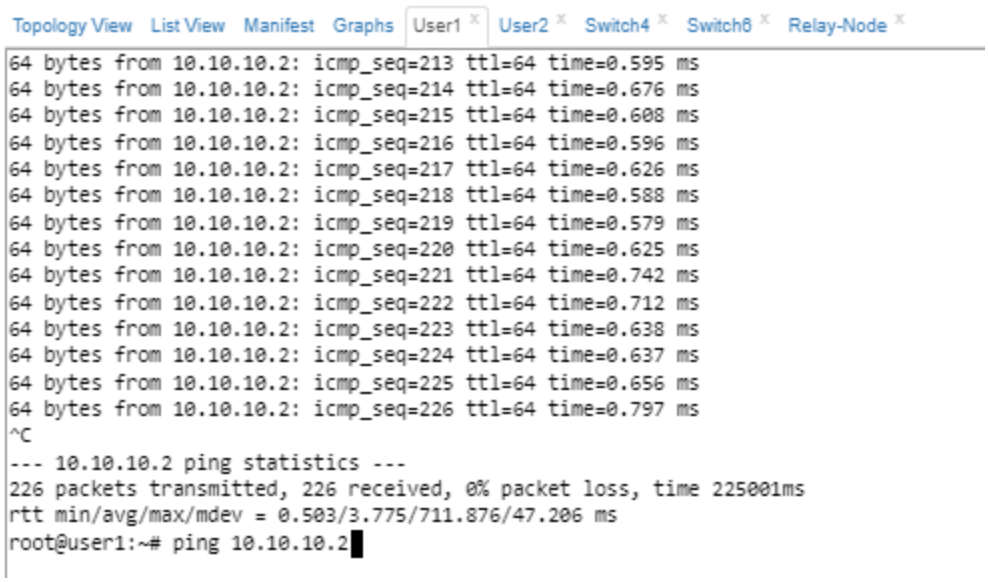

Fig 48: User2 is pinged again from User1.

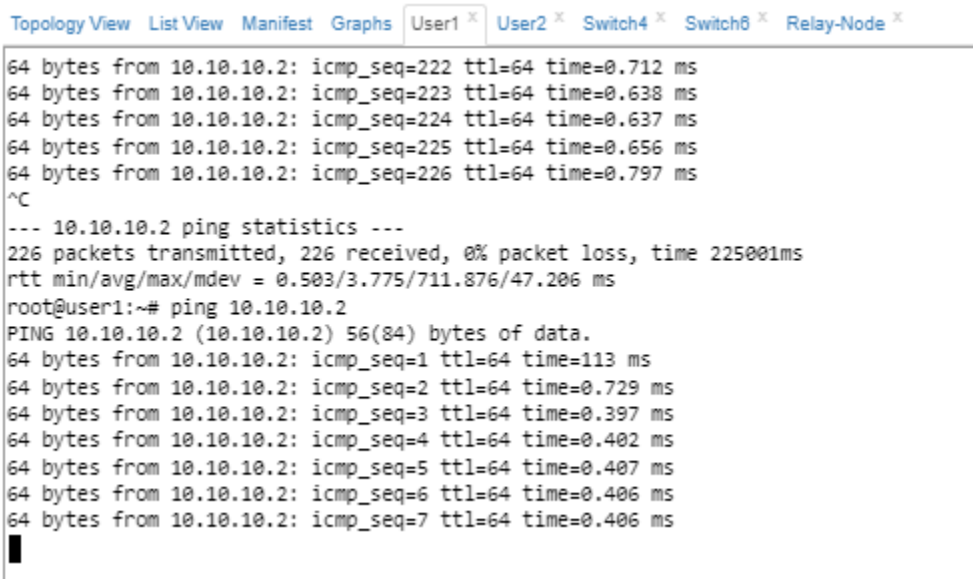

Fig 49: Ping is successful.

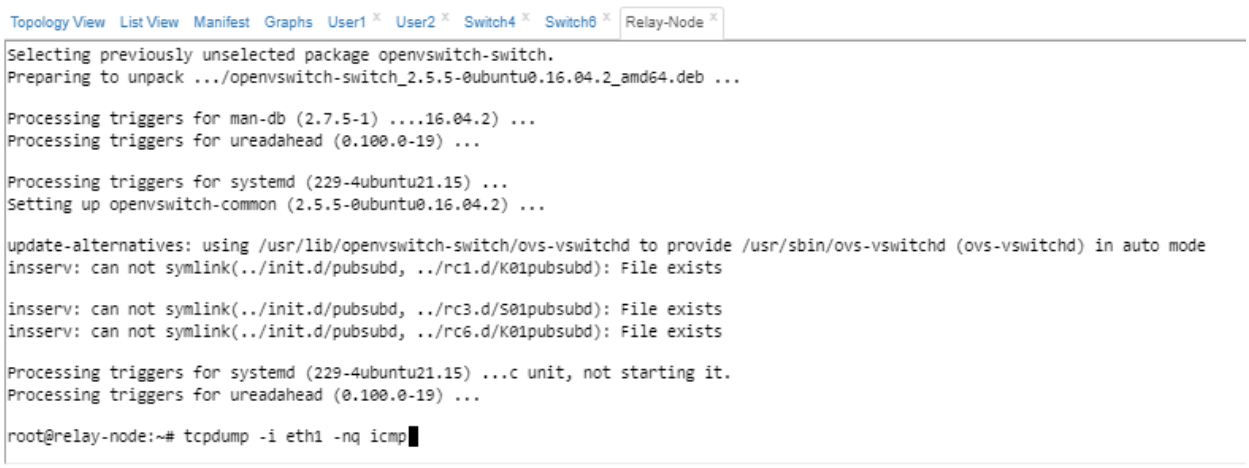

Fig 50: "tcpdump -i eth1 -nq icmp" is the command used to observe the flow of packets in Relay-Node.

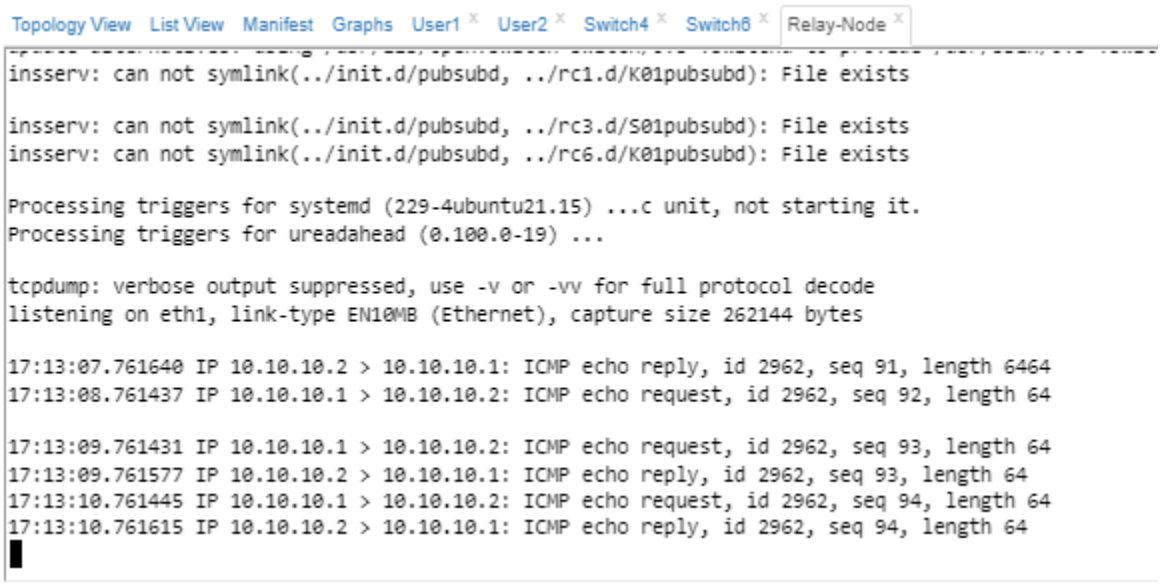

Fig 51: We see that the packets, unlike the previous case, have started flowing now.

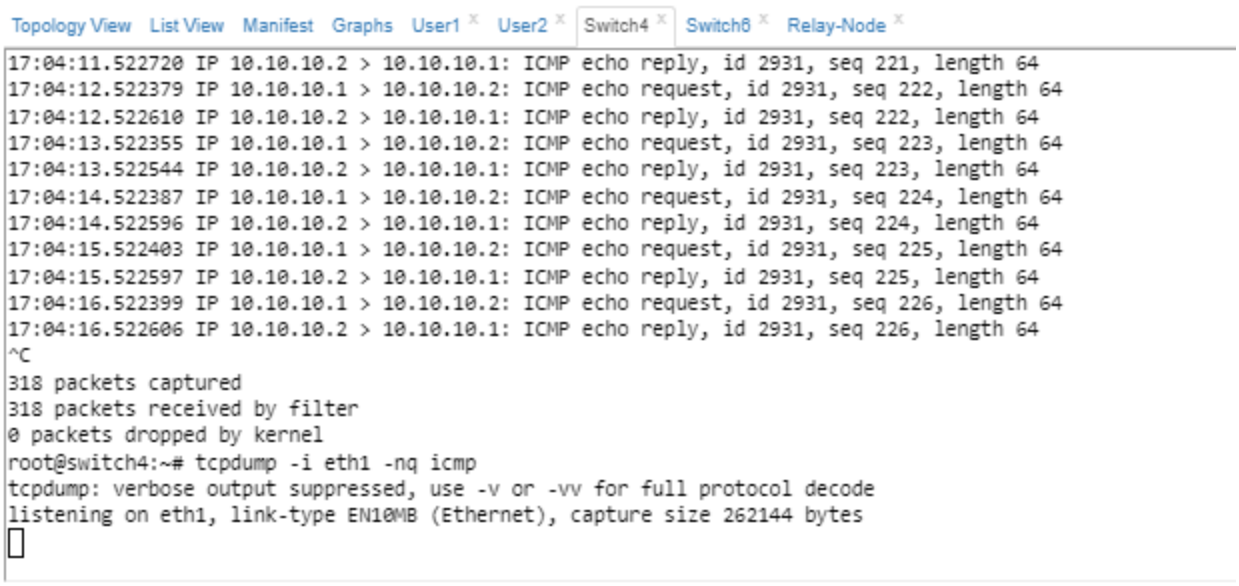

Fig 52: To see packet flow in Switch4 “tcpdump -i eth1 -nq icmp” is used. We see that no packet is flowing through this.

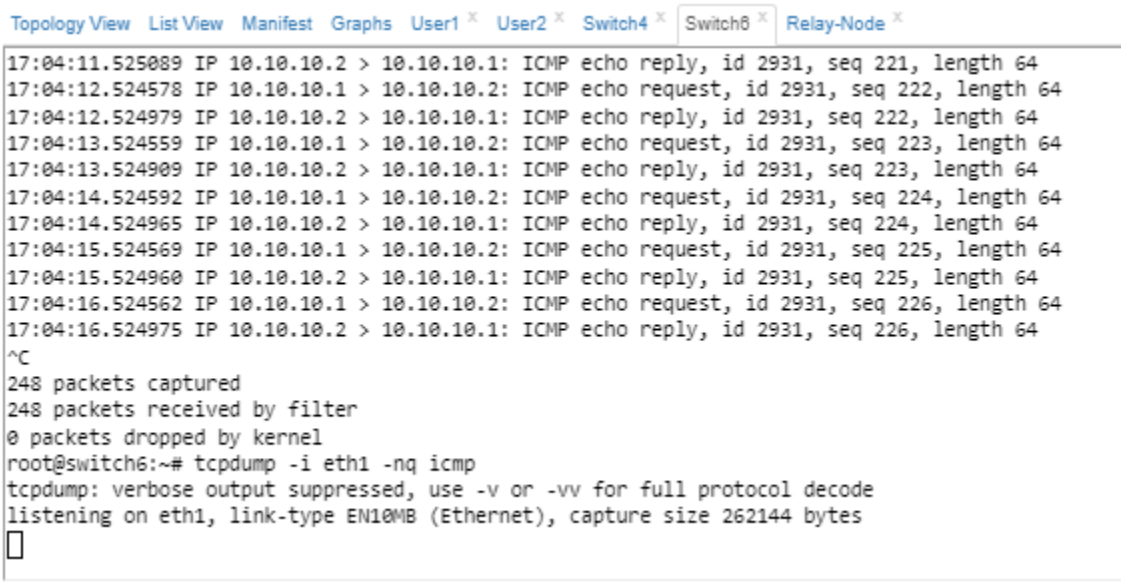

Fig 53: To see packet flow in Switch6 “tcpdump -i eth1 -nq icmp” is used. We see that no packet is flowing through this.

The above screenshots successfully demonstrate the Man-In-Middle-Attack process. In the first scenario, before the initiation of stack, we see that packets are flowing in Switch4 and Switch6 but not in Relay-Node. But, when there is attack happening, we see that there is no flow of packets in Switch4 and Switch6 whereas they are flowing in Relay-Node. We can successfully demonstrate the Man-In-Middle-Attack in the Relay-Node. Thus, the data gets stolen using this process by an attacker.