Final Project Report

Open Flow

EE 555

Gourp member
Huang, hui (huih@usc.edu)
Zhou, wenyan (wenyanzh@usc.edu)
Lu, wanyun (wanyunlu@usc.edu)

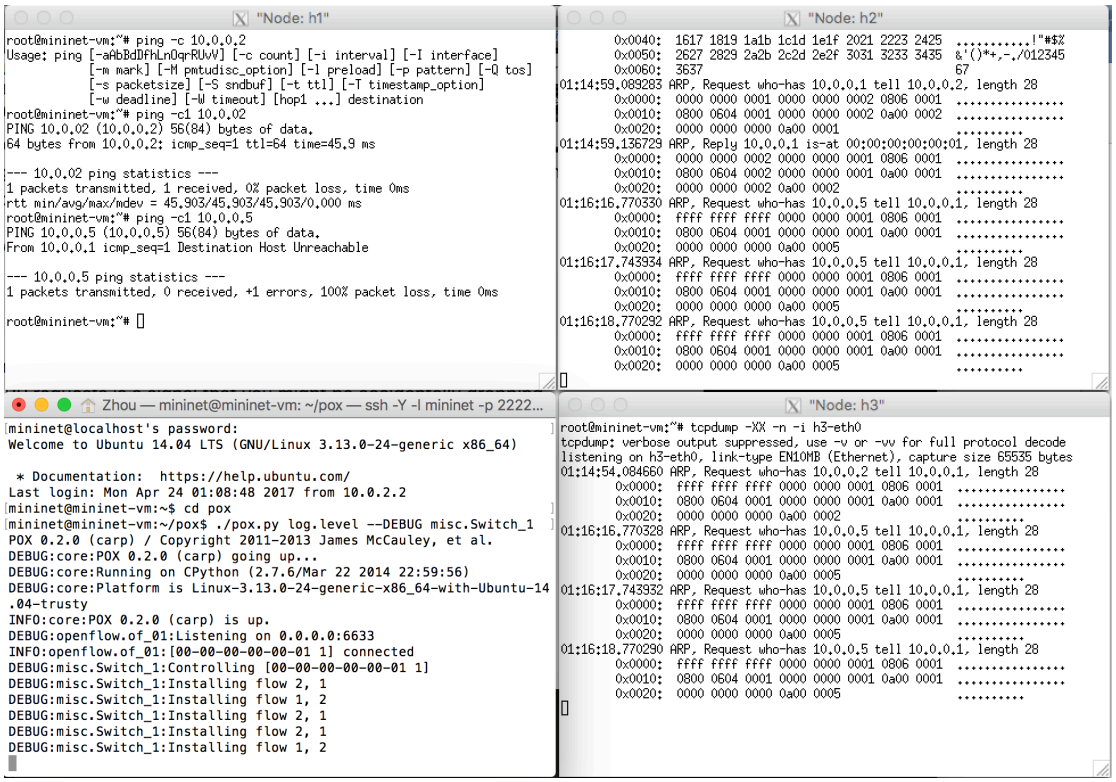
Abstract:
Open flow is an advance communication protocol, which can use the forwarding plane of a network switch or router. In this project, our group follow the instruction of open flow wiki to get our virtual network. Our group choose pox which is a python based SDN controller platform to modify our network. From this project, all of us get a better understanding of ARP and ICMP protocol.

Part 1:

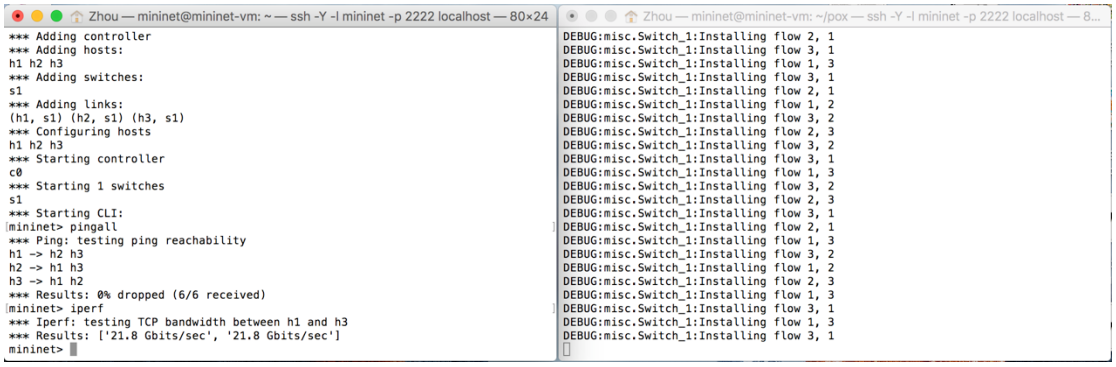
1. Learning switch
   It is the important basic knowledge for whole project. We use the tutorial material to create a learning switch.
   a. Create a mac-to-port table
   b. Switch receive a packet and check its destination mac address in table
      Controller will record the source mac address and the port number that packet comes in.
   c. If the table doesn't have a port number to match this mac address, then controller will flood this packet.
   d. If the table have the matching, controller will send this packet to destined port.

   Following the instruction, we use the ofp_flow_mod OpenFlow message to improve the performance of switch.

Testing controller:

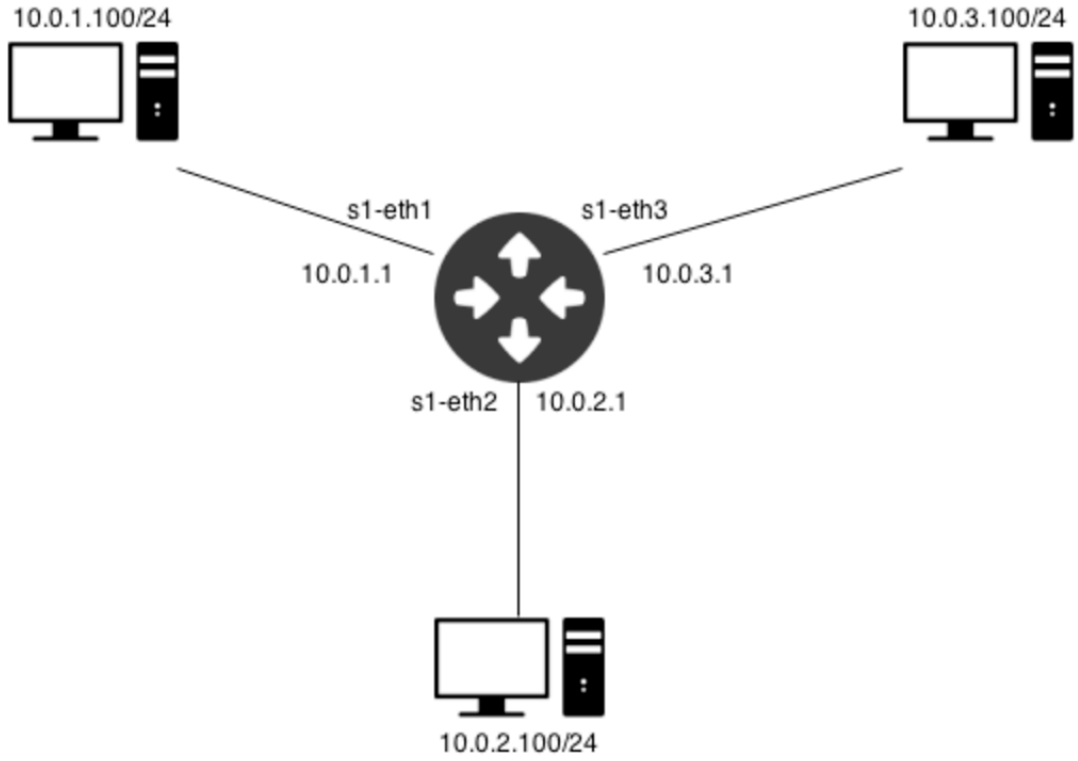


According to the figure, we can clearly see the ARP request and Rely message. At first try, we use h1 to ping h3. Due to the empty table, all the ports except input port will receive the packet, but only h3 will reply.



According the figure, we use the “pingall” to test all the hosts and the result shows every host can get the packet. After that, we use “iperf” to test the bandwidth and we can see it is 21.8 Gbits/sec.

2. Router Exercise
   First of all, we build the topology which contains one router and 3 hosts.

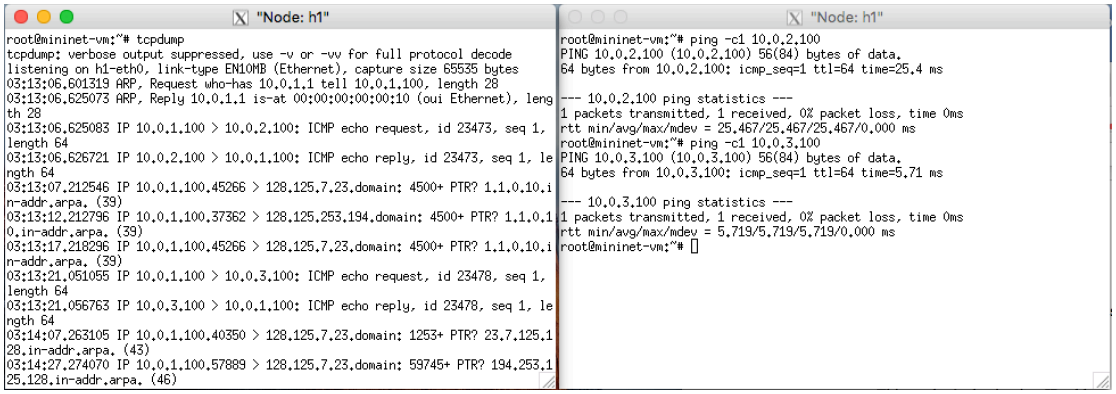


   a. Router gets a pocket
   b. If this pocket is an ARP packet
      We get two type of ARP packets
      1. This is an ARP request packet
         Router will send back an ARP reply packet to tell the source that router is the next node when it need to send a packet.
         The source will get the router's mac address.
      2. This is an ARP reply packet
         Router will forward this packet to its destination.
   c. If this is an ICMP packet
      We get two type of ICMP packet
      1. This is an ICMP echo request
         This packet's destination IP address is this router. The router will send back an echo packet to source.
         This packet's destination IP address is a host. The router will create a new frame and forward this frame to its destination.
         This packet's destination IP address is as unknown address. The router will send back a ICMP unreachable packet.
      2. This is an ICMP reply packet

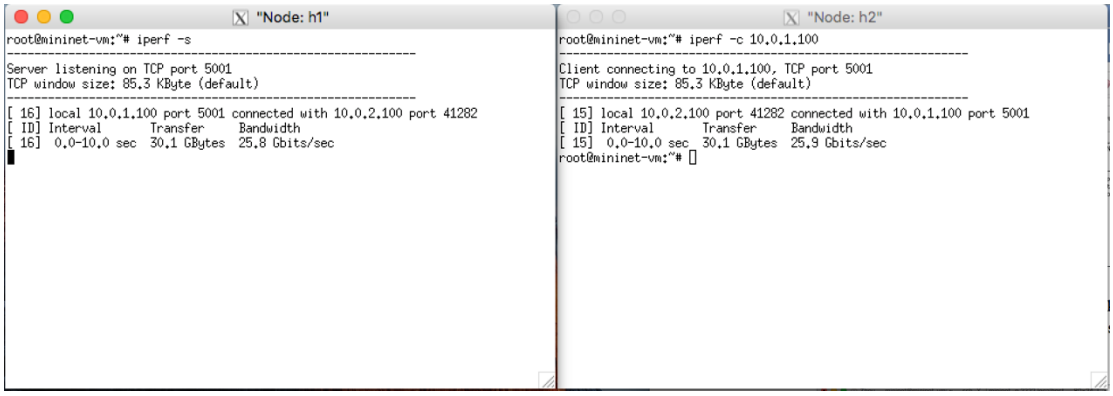
Router just send this packet to the destination.

d. If this is a normal IP packet
   Router will just send it to its destination IP address.

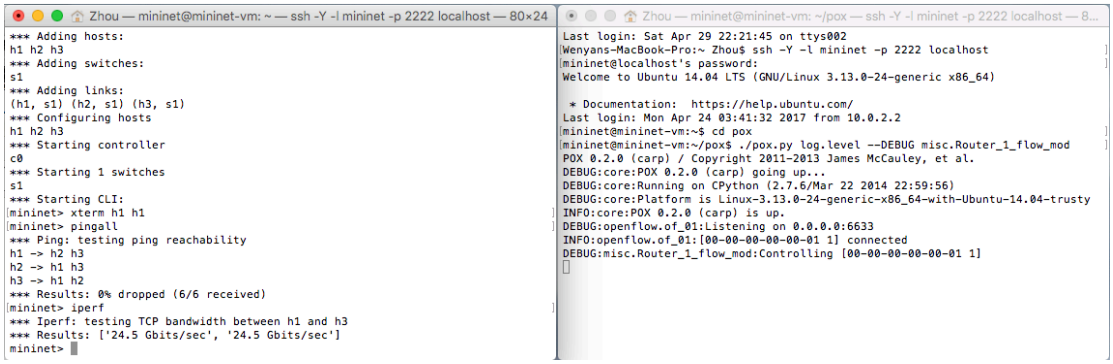
Testing Controller:



We first send packet to “10.0.2.100”.
According to figure, host 1 want to send packet to “10.0.2.100”. it needs know router’s mac address, so there are ARP request and ARP reply message. H1also need to know that if the router can reach the “10.0.2.100” or not, so there exist ICMP echo request and reply.



According to figure, we use the host 1 as sever and host 2 as client.
This is an example for normal IP packet. From the figure, the bandwidth and port number show very clear.

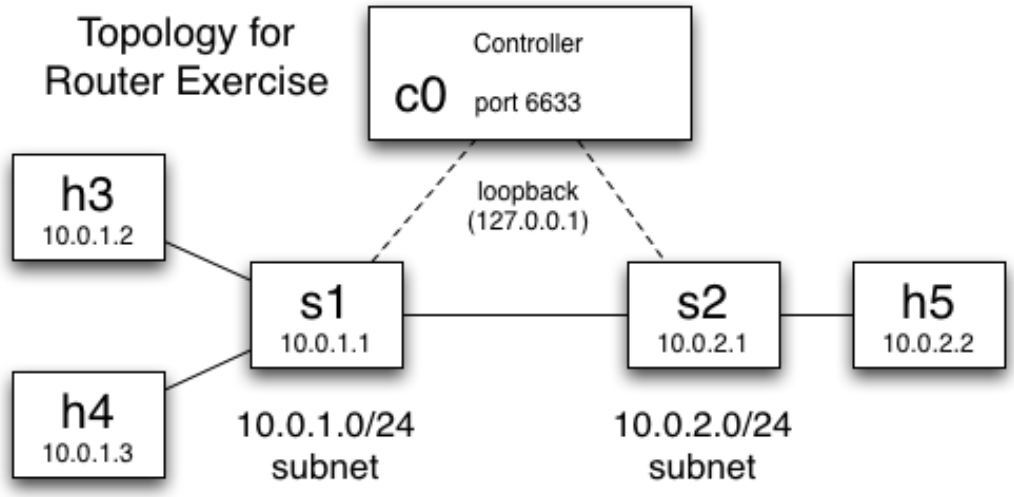


According to figure, we can check that the router is pingable. With the setting of the flow mod, the bandwidth will be increased to 24.5 Gbits/sec.

Part 2:

1. Advance Topology

   First of all, we build the topology which contains two routers and 3 hosts.



   This part is very similar with the router exercise, but we encounter with some problem in this part.

a. We need to find the correct port number for two routers

   At the beginning of programming, our group think the router's port number can be changed by ourselves. We just set the port number like "1, 2, 3, 4, 5", but we can't get a feasible connection. Then we use "log.dug" method to help us print the correct port number which is "1, 2, 3, 1, 2".

b. We need to specify each router's behavior

   1. ARP part

      If host 3 want host 4's mac address, it is very simple. The router 1 just forward this request packet to host 4.

If destination IP address of the ARP request packet is router's address, router will send back a reply packet and tell them router's mac address.

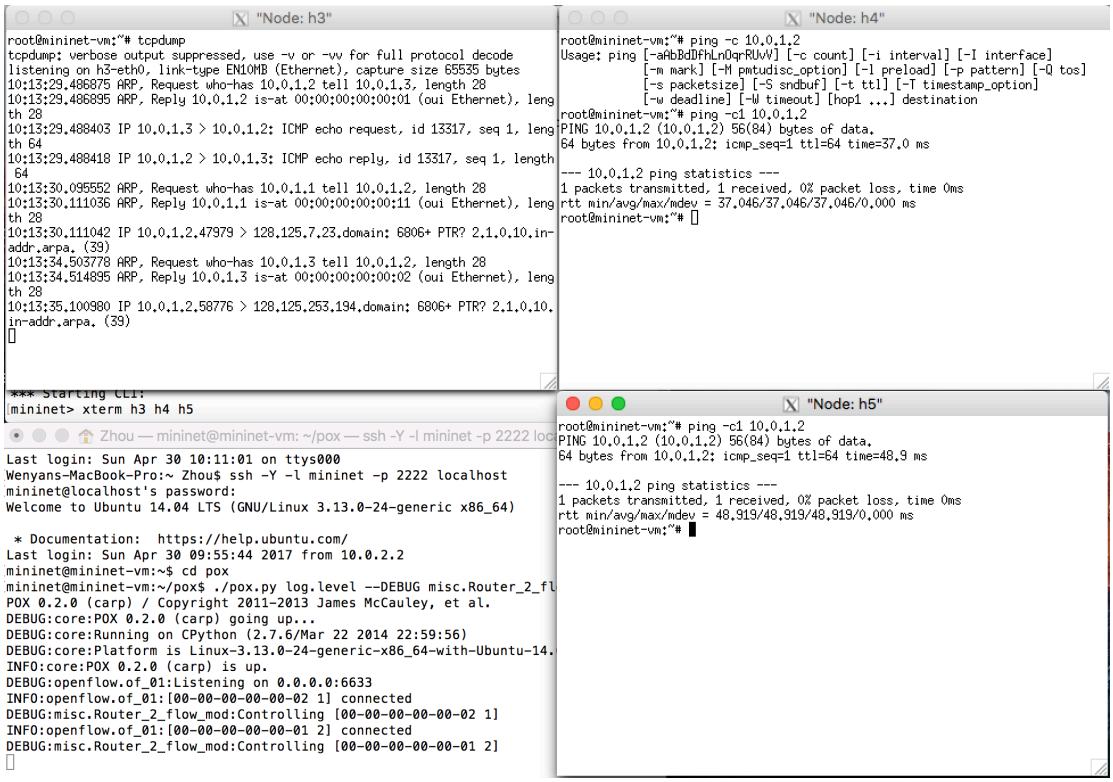
ARP reply packet is same with part 1.

2. IP part

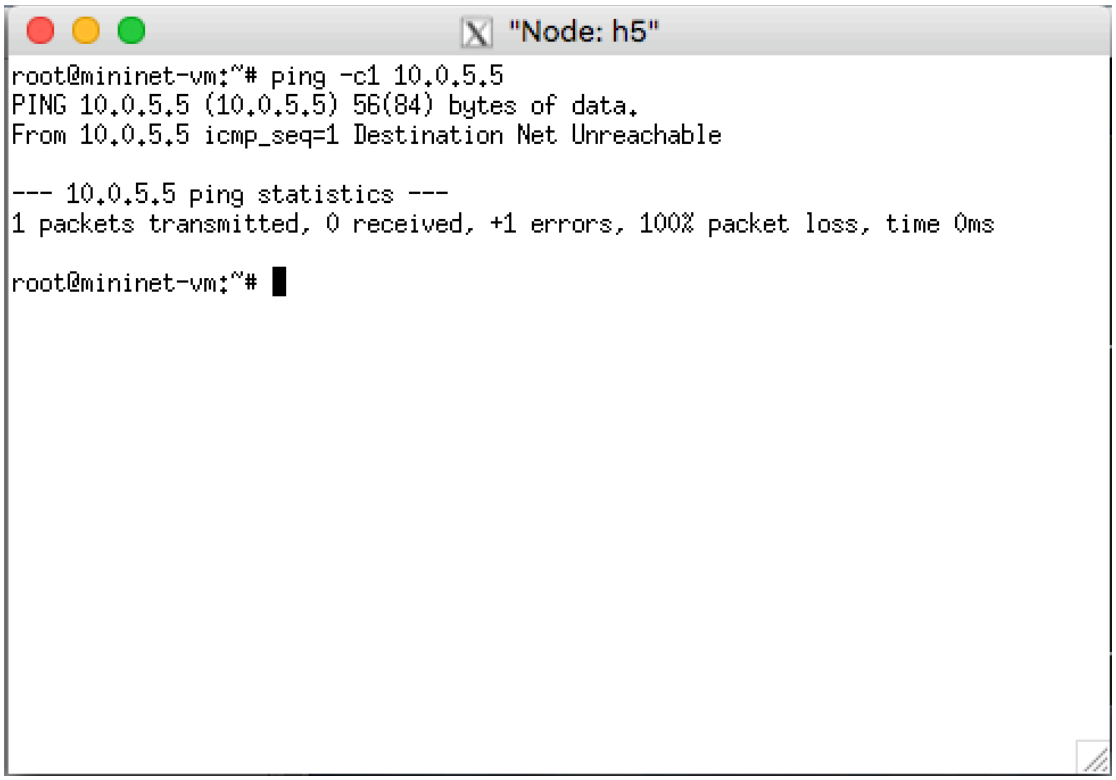
   ICMP request packet. If the packet's destination mac address is router1 and destination IP address is router 1, router 1 will send a reply packet. If the destination IP address is in subnet 1, then send it to the distinct port. If the destination IP address is in subnet 2, then send it to router 2.

   ICMP reply and Normal IP packet should also follow the same way that has mentioned above.

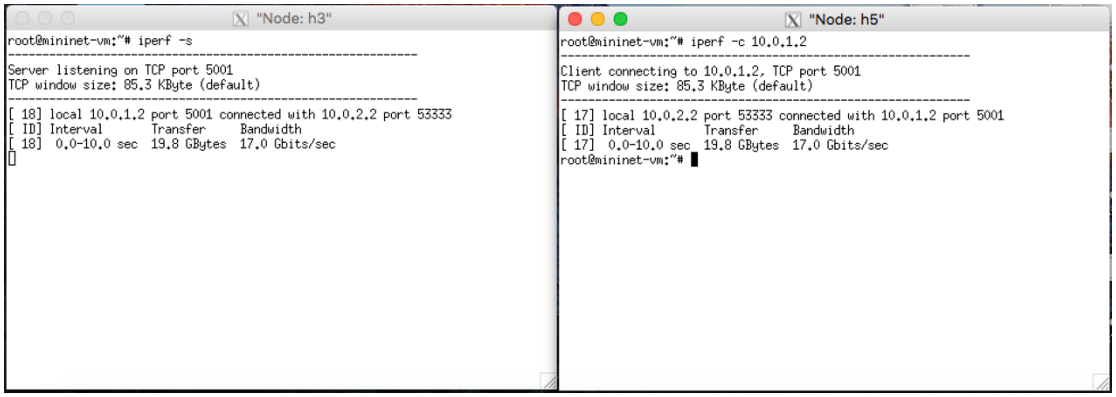
Testing Controller:



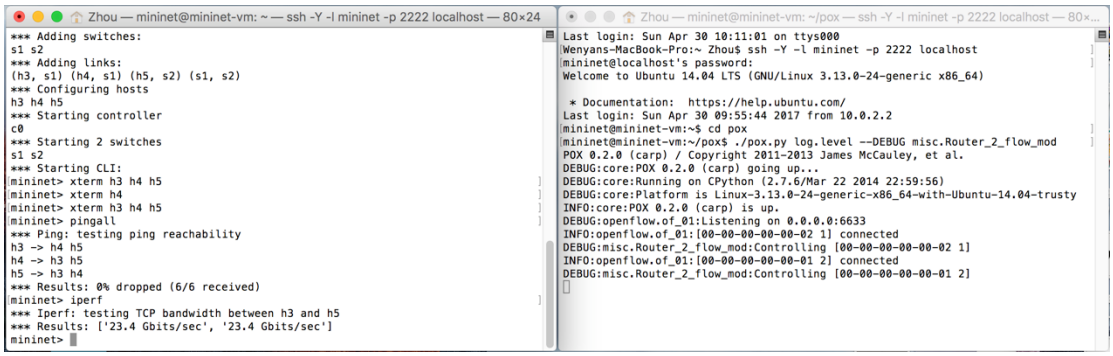
To the text our router, we make host 3 to listen the network. In host 4 and host 5 which can represent the same and different network, we type the "ping 10.0.1.2". That is to say, host 4 and 5 send packet to host 3. According to the figure, we can clear see the ARP request and reply packet. All the packets are sent successfully.



In this figure, we make host 5 to a unknown IP address and the packet was lost.



In this figure, we test the bandwidth between host 3 and host 5.

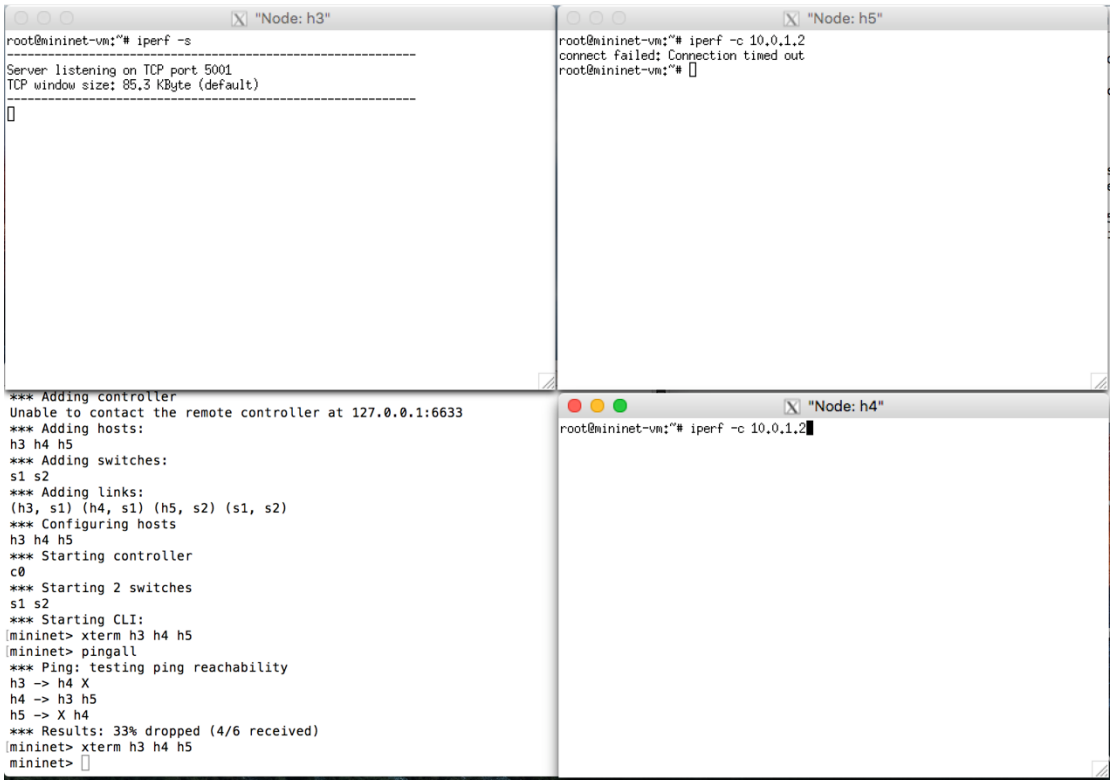


According to this figure, it shows all the result for our work. Both of two routers are pingable. With the help of flow mod, we can send large amount of packet at the same time, which give our network a 23.4 Gbits/sec bandwidth.

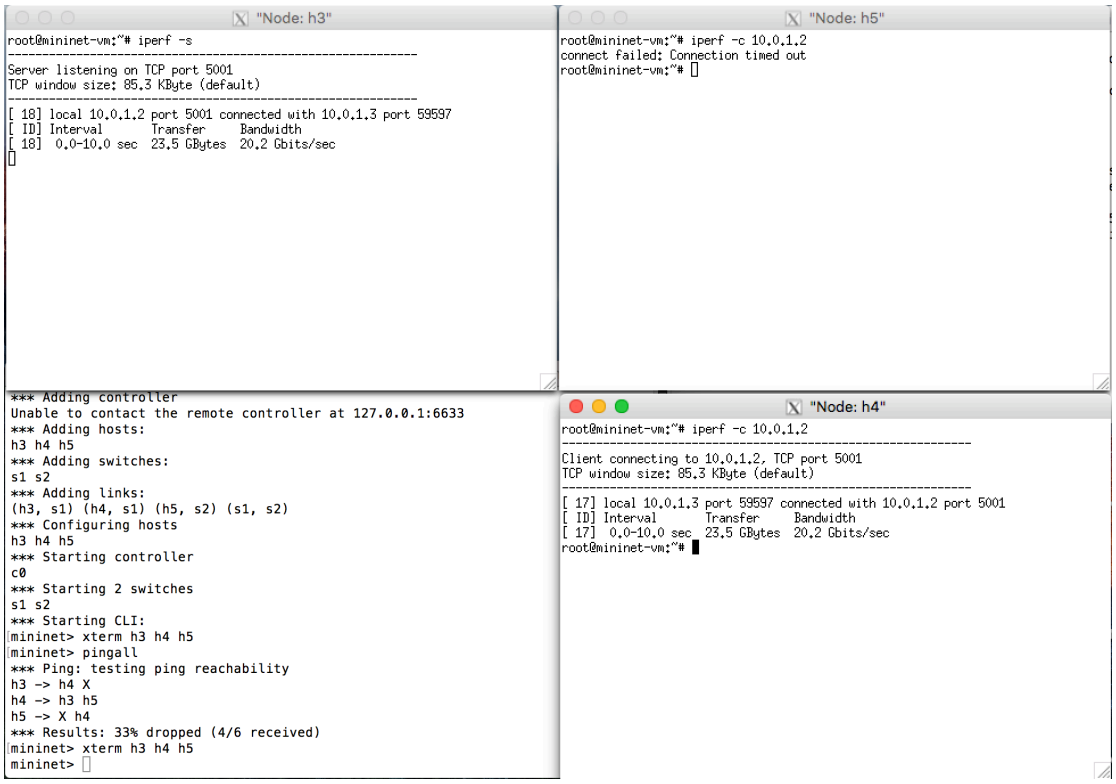
2. Firewall

   To realize the function of firewall, we created an array to store the information of the source mac address and destination mac address which should be cut off in the communication. In this experiment, in the self.deny array, we recorded the mac address of h3 and the mac address of the default router of h3. As a result, we can cut of the communication between the host3 and the router1. Hosts from the outside network can no longer send message to h3 because the packet must go through the default router. In this situation, we also cut off the ICMP reply packet destined to host 3. So, h3 can not send packets out of the network either. Because, communication between h3 and h4 does not need to the help of router, so this cut of does not influent the packets transmission between h3 and h4. In the terminal, we can see that only h3 and h5 line has lost their packets.

Testing Controller:



According to this figure, we first make the host 3 as a server. Then at host 5, we need to get connection to host 3. However, we waited for a minute, it shows that connection time out.



In this figure, we can see that host 4 can get connection to host 3, because they are in the same network.

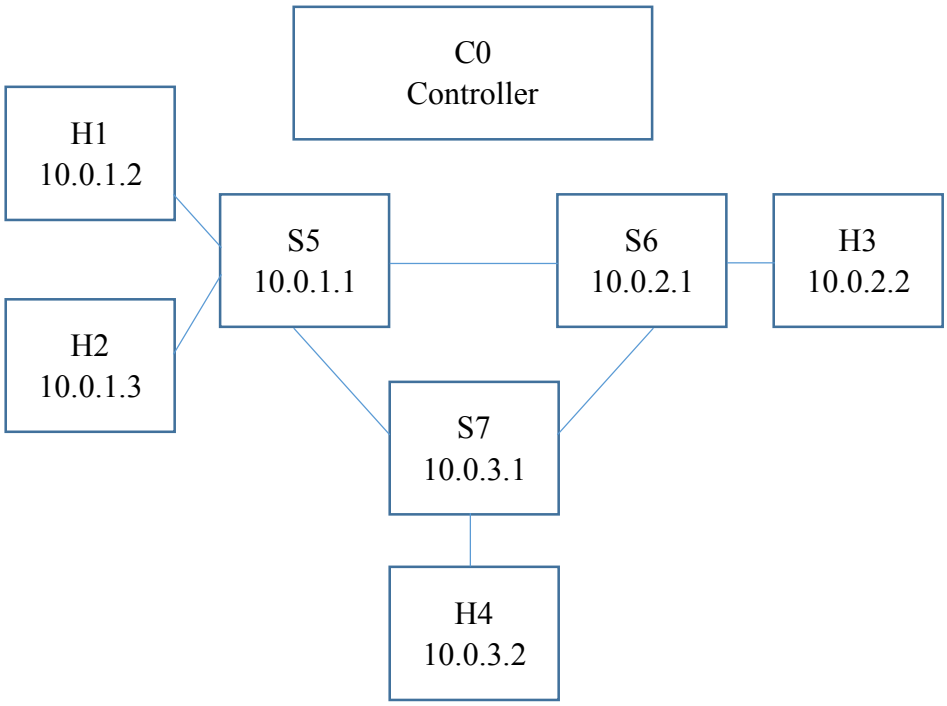
Both of two figures, we can see that host 3 has build a fire wall to deny all the packet which comes from other network.

We can use “pingall” to show the result directly. Host 5 can’t connect to host 3.

Part 3:

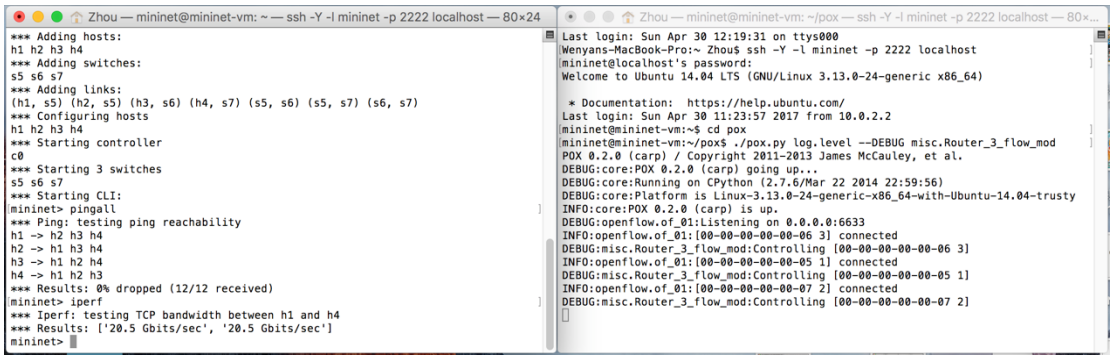
1. Bonus

   First of all, we build the topology which contains 3 routers and 4 hosts.



   This part is very similar with the advance topology and we only need to do a little modify.

   Testing Controller:



   According to this figure, we can see that our topology can work successfully and has a 20.5 Gbits/sec bandwidth.

Conclusion:

From this project, according to the simulation of SDN, we have a deeper understanding of the principle and mechanism of Software Defined Network. Inside the controller, from the detailed information of routing table, we need to react to the different information from the router. This kind of mechanism can reduce the load in the router and increase the efficiency of the routing mechanism. Plus, in the project, we know more about the ARP, ICMP messages. We know how these messages work and some details about the transmission. This experiment can help us to establish the solid basic knowledge about the SDN and benefit us a lot.