Creating a production level network using SDN

Archit Taneja, Prateek Lohchubh, Sukrit Kalra, Tarun Verma

April 28, 2016

Abstract

The project includes creating a production level network using a given set topology which involves the use of different nodes like a load balancer, RYU controller, firewall, vSwitch instances and multiple clients

1 Topology

The topology being used in the above network is given in Figure 1 and roles corresponding to the nodes are given in figure 2.

2 LoadBalancer

We have a working loadbalancer, which fires requests to the hosts coming from the client in a round robin fashion. For each new TCP connection, the load balancer selects one of the specified hosts in round robin order of the connection requests. The load balancer maintains a mapping of active connections—identified by the client's IP and TCP port—to the assigned hosts. For all packets sent from clients to the load balancer, the load balancer rewrites the destination IP and MAC addresses to the IP and MAC addresses of the selected host. The mapping information stored by the load balancer is used to determine the appropriate host IP and MAC addresses that should be written into a packet arriving from a client. For all packets sent from servers to clients, the load balancer rewrites the source IP and MAC addresses to the IP and MAC addresses of the load balancer.

3 Firewall

We have a working firewall which filters packets come from the client, based on the src MAC address. The firewall checks the MAC address, and checks with the current table to see if the packet has to be dropped, and drops the packet accordingly.

How to Run

1. Run the topology file using `sudo python <topologyfile.py>`
2. Run the controller program file using `ryu-manager <controllerfile.py>`

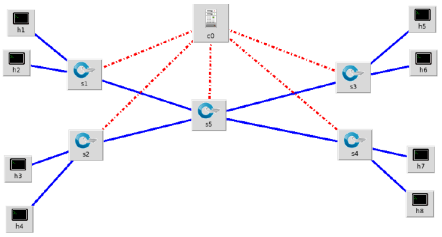

Figure 1: Topology.

Node name	Role (and functions)
c0	the SDN (Ryu) controller
s1	Open vSwitch instance
s2	Open vSwitch instance
s3	Open vSwitch instance Load balancer-balances load between h5 and h6
s4	Open vSwitch instance Load balancer-balances load between h7 and h8
s5	Open vSwitch instance
h1	*HTTPS* traffic requesting client
h2	*HTTP* traffic requesting client
h3	*HTTPS* traffic requesting client
h4	*HTTP* traffic requesting client
h5	*HTTP/HTTPS* Server instance
h6	*HTTP/HTTPS* Server instance
h7	*HTTP/HTTPS* Server instance
h8	*HTTP/HTTPS* Server instance

Figure 2: Roles