SDN – Openflow Firewall Assignment

Student: Thai Kha Le

StudentID: 20061112

Implemented features

- Layer 2,3 (IPv4 only), and 4 Firewall
- Bidirectional blocking for each rule
- Firewall allows by default
- The Firewall code is called from inside the `act_like_switch` function

Firewall rule format

- TYPE (`ip` OR `mac`),SRC_ADDRESS,DST_ADDRESS,PORT
- ADDRESS can either be MAC address, IP address or `*`
- PORT can either be port number, `*`, or left empty which means the same as `*`

What the script does

When it loads:

- Read the firewall rules (only those that have the correct format described above)

When a packet comes in:

- Learn the port-to-mac mapping (switch functionality)
- Parse the packet for layers 2,3,4 data into an layers234_data dictionary, including:
  - Source MAC address (src_mac)
  - Destination MAC address (dst_mac)
  - Source IP address (src_ip) - if available
  - Destination IP address (dst_ip) - if available
  - Destination port (dst_port) - if available
- layers234_data is then checked against each firewall rule:
  - If it is a layer 2 rule (`mac` id), and `src_mac` matches SRC_ADDRESS and `dst_mac` matches DST_ADDRESS, or `src_mac` matches DST_ADDRESS and `dst_mac` matches SRC_ADDRESS: packet is dropped
  - If it is a layer 3 rule (`ip` id), and `src_ip` matches SRC_ADDRESS and `dst_ip` matches DST_ADDRESS, or `src_ip` matches DST_ADDRESS and `dst_ip` matches SRC_ADDRESS: packet is dropped
  - If it is a layer 4 rule (`ip` id with a specific PORT number), and `src_ip` matches SRC_ADDRESS and `dst_ip` matches DST_ADDRESS, or `src_ip` matches DST_ADDRESS and `dst_ip` matches SRC_ADDRESS, and `dst_port` matches PORT: packet is dropped
- If layers234_data does not match any rule, the firewall allows it and let the rest of the "switch" code handles the packet

Testing

Quick note

I use markdown to pdf plugin in Atom which messes up MAC address. So □ means `: 0 0 :` (ignore the spaces)

Hosts' addresses

When I ran the tests, the hosts were as follows:

- h1:
  - IP address: 10.0.0.1
  - MAC address: `00□00□00:01`
- h2:
  - IP address: 10.0.0.2
  - MAC address: `00□00□00:02`
- h3:
  - IP address: 10.0.0.3
  - MAC address: `00□00□00:03`

- h4:
  - IP address: 10.0.0.4
  - MAC address: `00:00:00:00:00:04`

Tested rules

```
ip,10.0.0.1,10.0.0.2,
mac,00:00:00:00:00:02,00:00:00:00:00:03
ip,10.0.0.3,10.0.0.4,80
ip,*,*,8080
```

These rules will be referred to later on respectively by rule 1,2,3,4

Commands

Command used to start the firewall script (this should be run firstly): `./pox.py log.level --DEBUG misc.firewall`

Command used to start the topology in another terminal: `sudo killall controller || sudo mn -c && sudo mn --topo single,4 --mac --switch ovsk --controller remote`

Pingall

`pingall` 's output should be:

```
h1 -> X h3 h4
h2 -> X X h4
h3 -> h1 X h4
h4 -> h1 h2 h3
*** Results: 33% dropped (8/12 received)
```

Explain:

- h1 cannot ping h2 due to 1 (layer 3)
- h2 cannot ping h1 due to rule 1 (layer 3)
- h2 cannot ping h3 due to rule 2 (layer 2)
- h3 cannot ping h2 due to rule 2 (layer 2)

HTTP

First, HTTP servers are started by the following commands:

```
h4 python -m SimpleHTTPServer 80 &
h3 python -m SimpleHTTPServer 80 &
h2 python -m SimpleHTTPServer 80 &
h1 python -m SimpleHTTPServer 80 &
```

Then, all of the following wget commands should fail:

```
h1 wget -O - h2
h2 wget -O - h1
h3 wget -O - h4
h4 wget -O - h3
h1 wget -O - 10.0.0.3:8080
h3 wget -O - 10.0.0.1:8080
```

Explain:

- h1 cannot wget h2 due to rule 1 (layer 3)
- h2 cannot wget h1 due to rule 1 (layer 3)
- h3 cannot wget h4 due to rule 3 (layer 4)
- h4 cannot wget h3 due to rule 3 (layer 4)
- h1 cannot wget h3 on port 8080 due to rule 4 (layer 4)
- h3 cannot wget h1 on port 8080 due to rule 4 (layer 4)

Then, the following wget commands should pass:

```
h1 wget -O - 10.0.0.3:80
h3 wget -O - 10.0.0.1:80
```

Explain:

- There is no rule that blocks layer 2 and 3 traffic from h1 to h3 and vice versa
- There is a rule that blocks all layer 4 traffic going to port 8080 but port 80 is used wget so the traffic is not blocked