

SDN  
Total Solution  
Provider



COD  
(Customer Optimized Datacenter)  
Platform



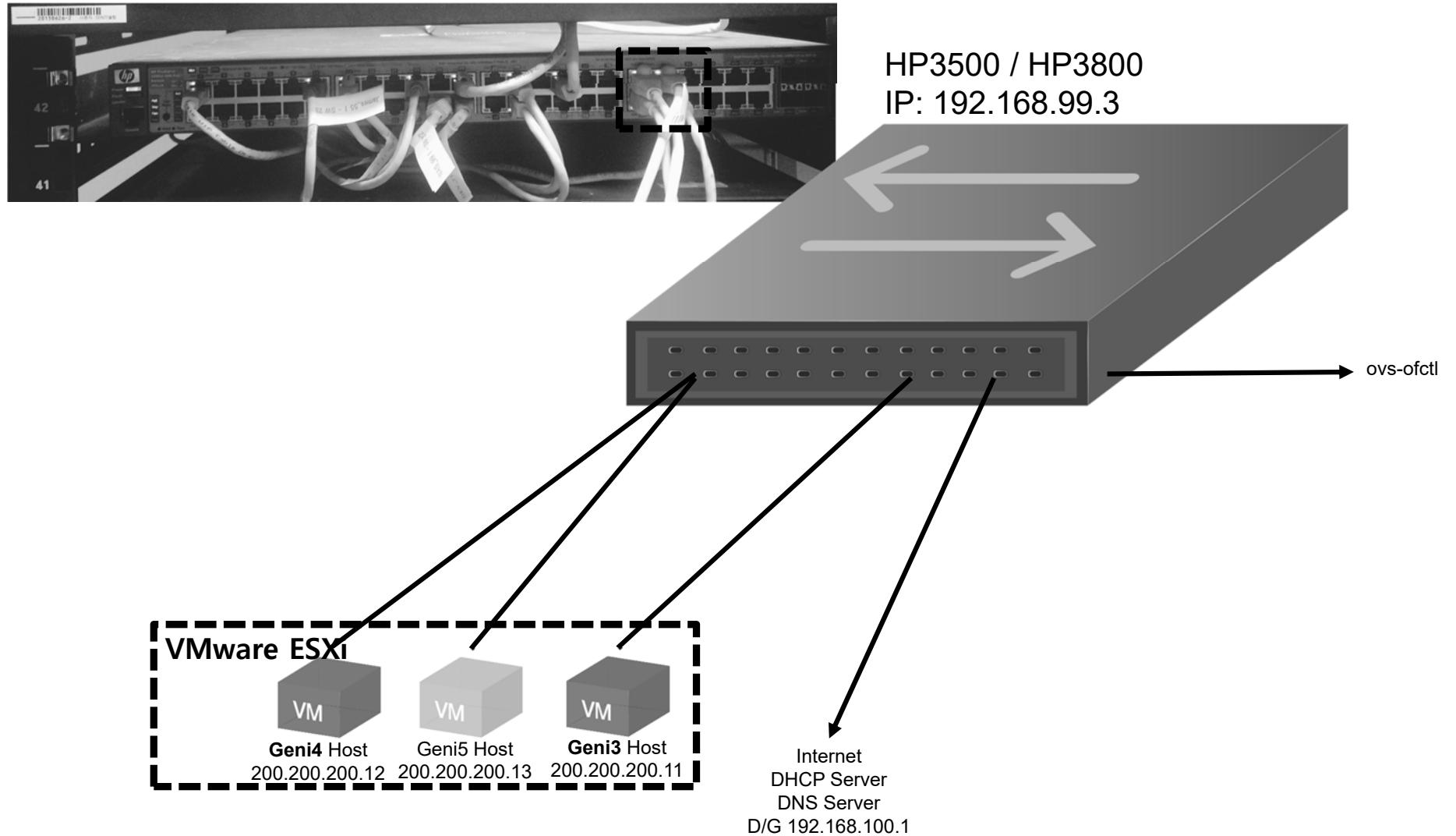
## SDN Switch Test Examples

June 2016

안종석  
NAIM Networks, Inc.



# 테스트 구성





# HP3500 OpenFlow 구성 정보

- HP3500 CLI 명령어 : show openflow instance test

```
HP-3500yl-48G-PoEP(config)# show openflow instance test

Configured OF Version      : 1.0
Egress Only Ports          : None
Instance Name               : test
Data-path Description       : test
Admin. Status              : Enabled
Member List                 : VLAN 888
Pipeline Model              : Standard Match
Listen Port                : 6633
Oper. Status                : Up
Oper. Status Reason         : NA
Datapath ID                 : 0378e4115b748fc0
Mode                        : Active
Flow Location               : Hardware and Software
No. of Hw Flows             : 1
No. of Sw Flows             : 0
Hw. Rate Limit              : 0 kbps
Sw. Rate Limit              : 100 pps
Conn. Interrupt Mode        : Fail-Secure
Maximum Backoff Interval   : 60 seconds
Probe Interval              : 10 seconds
Hw. Table Miss Count       : 1542888
No. of Sw Flow Tables       : NA
Table Model                 : Single Table

Controller Id Connection Status Connection State Secure Role
----- ----- ----- ----- -----
1           Disconnected     Connecting    No      Equal

HP-3500yl-48G-PoEP(config) #
```



# HP3500 config 정보

- HP3500 CLI 명령어 : show running-config
- OVS 명령어 : ovs-ofctl show tcp:192.168.99.3

```
HP-3500yl-48G-PoEP(openflow)# show running-config

Running configuration:

; J9311A Configuration Editor; Created on release #K.15.17.0007
; Ver #08:02.ff.f7.fc.7f.ff.3f.ef:ae
hostname "HP-3500yl-48G-PoEP"
module 1 type j93yya
module 2 type j93xxa
module 3 type j9312a
ip default-gateway 192.168.99.1
snmp-server community "public" unrestricted
openflow
    controller-id 1 ip 192.168.99.100 controller-interface vlan 999
    instance "test"
        listen-port
        member vlan 888
        controller-id 1
        enable
        exit
    enable
    exit
vlan 1
    name "DEFAULT_VLAN"
    no untagged 1-48
    untagged A1-A4
    ip address dhcp-bootp
    exit
vlan 888
    name "VLAN888"
    untagged 37-40
    ip address 192.168.88.3 255.255.255.0
    exit
vlan 999
    name "VLAN999"
    untagged 1-36,41-48
    ip address 192.168.99.3 255.255.255.0
    exit
```

```
* Documentation: https://help.ubuntu.com/
Last login: Wed Sep 16 14:00:33 2015
mininet@mininet-vm:~$ ovs-ofctl show tcp:192.168.99.3
OFPT_FEATURES_REPLY (xid=0x2): dpid:0378e4115b748fc0
n_tables:16, n_buffers:0
capabilities: FLOW_STATS TABLE_STATS PORT_STATS QUEUE_STATS ARP_MATCH_IP
actions: OUTPUT SET VLAN_VID SET VLAN_PCP STRIP_VLAN SET_DL_SRC SET_DL_DST SET_NW_SRC SET_NW_DST SET_NW_TOS SET_TP_SRC SET_TP_DST
37(37): addr:e4:11:5b:74:8f:db
    config: 0
    state: STP_FORWARD
    current: 1GB-FD AUTO_NEG
    supported: 10MB-HD 10MB-FD 100MB-HD 100MB-FD 1GB-FD AUTO_NEG
    speed: 1000 Mbps now, 1000 Mbps max
38(38): addr:e4:11:5b:74:8f:da
    config: 0
    state: STP_FORWARD
    current: 1GB-FD AUTO_NEG
    supported: 10MB-HD 10MB-FD 100MB-HD 100MB-FD 1GB-FD AUTO_NEG
    speed: 1000 Mbps now, 1000 Mbps max
39(39): addr:e4:11:5b:74:8f:d9
    config: 0
    state: STP_FORWARD
    current: 1GB-FD AUTO_NEG
    supported: 10MB-HD 10MB-FD 100MB-HD 100MB-FD 1GB-FD AUTO_NEG
    speed: 1000 Mbps now, 1000 Mbps max
40(40): addr:e4:11:5b:74:8f:d8
    config: 0
    state: STP_FORWARD
    current: 1GB-FD AUTO_NEG
    supported: 10MB-HD 10MB-FD 100MB-HD 100MB-FD 1GB-FD AUTO_NEG
    speed: 1000 Mbps now, 1000 Mbps max
LOCAL(local): addr:e4:11:5b:74:8f:c0
    config: 0
    state: 0
    speed: 0 Mbps now, 0 Mbps max
OFPT_GET_CONFIG_REPLY (xid=0x4): frags=normal miss_send_len=0
mininet@mininet-vm:~$
```



# 테스트 단말기 화면 - OF 1.0

- **ovs-ofctl 명령 실행 창**

```
naim@dev-server:~$ ovs-ofctl del-flows $testovs
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
naim@dev-server:~$
```

- **geni3 = sender = 200.200.200.11**

```
ubuntu@ubuntu:~$ nmap 200.200.200.12 -p 80
Starting Nmap 6.40 ( http://nmap.org ) at 2015-09-16 16:30 KST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.03 seconds
ubuntu@ubuntu:~$
```

- **geni4 = receiver= 200.200.200.12**

```
ubuntu@ubuntu:~$ !T
-bash: !T: event not found
ubuntu@ubuntu:~$ sudo tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
```



# Rule for ARP - OF 1.0

---

- 기본적인 arp 통신을 위한 룰을 셋팅함
  - \$testovs = tcp:192.168.99.3:6633
- 룰
  - ovs-ofctl add-flow \$testovs  
arp,nw\_dst=200.200.200.0/24,actions=normal
  - ovs-ofctl add-flow \$testovs  
arp,nw\_src=200.200.200.0/24,actions=normal



# Version Check - OF 1.0

---

- Version 확인: 1.0

```
2015-09-16T05:32:26Z|00023|ofctl|DBG|connecting to tcp:192.168.99.3:6633
2015-09-16T05:32:26Z|00024|poll_loop|DBG|wakeup due to [POLLOUT] on fd 7 (192.168.99.77:49048<->192.168.99.3:6633) at ../lib/stream-fd.c:120
2015-09-16T05:32:26Z|00025|poll_loop|DBG|wakeup due to [POLLOUT] on fd 7 (192.168.99.77:49048<->192.168.99.3:6633) at ../lib/stream-fd.c:120
2015-09-16T05:32:26Z|00026|vconn|DBG|tcp:192.168.99.3:6633: sent (Success): OFPT_HELLO (xid=0x3):
version bitmap: 0x01
2015-09-16T05:32:26Z|00027|poll_loop|DBG|wakeup due to [POLLIN] on fd 7 (192.168.99.77:49048<->192.168.99.3:6633) at ../lib/stream-fd.c:124
2015-09-16T05:32:26Z|00028|poll_loop|DBG|wakeup due to [POLLIN] on fd 7 (192.168.99.77:49048<->192.168.99.3:6633) at ../lib/stream-fd.c:124
2015-09-16T05:32:26Z|00029|vconn|DBG|tcp:192.168.99.3:6633: received: OFPT_HELLO (xid=0x131):
version bitmap: 0x01
2015-09-16T05:32:26Z|00030|vconn|DBG|tcp:192.168.99.3:6633: negotiated OpenFlow version 0x01 (we support version 0x01, peer supports version 0x01)
2015-09-16T05:32:26Z|00031|vconn|DBG|tcp:192.168.99.3:6633: sent (Success): OFPT_GET_CONFIG_REQUEST (xid=0x4):
2015-09-16T05:32:26Z|00032|poll_loop|DBG|wakeup due to [POLLIN] on fd 7 (192.168.99.77:49048<->192.168.99.3:6633) at ../lib/stream-fd.c:124
2015-09-16T05:32:26Z|00033|vconn|DBG|tcp:192.168.99.3:6633: received: OFPT_GET_CONFIG_REPLY (xid=0x4): frags=normal miss_send_len=0
SET_GENEVE_REPLY (idle 0ms) from channel 192.168.99.3:6633
```



# Allow TCP 80 - OF 1.0

---

## Allow TCP 80

```
ovs-ofctl add-flow tcp:192.168.99.3:6633  
priority=37000,tcp,nw_src=200.200.200.1,tp_dst=80,actions=normal  
ovs-ofctl add-flow tcp:192.168.99.3:6633  
priority=37000,tcp,nw_dst=200.200.200.1,tp_src=80,actions=normal  
ovs-ofctl dump-flows tcp:192.168.99.3:6633
```



# Allow TCP 80 - OF 1.0

- ovs-ofctl, Geni3, Geni4 사용

[DevServer]

```
naim@dev-server:~$ ovs-ofctl del-flows $testovs
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
naim@dev-server:~$
```

[Eddie]Geni3 (1)

```
ubuntu@ubuntu:~$ nmap 200.200.200.12 -p 80
Starting Nmap 6.40 ( http://nmap.org ) at 2015-09-16 16:30 KST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.03 seconds
ubuntu@ubuntu:~$
```

[Eddie]Geni4

```
ubuntu@ubuntu:~$ !T
-bash: !T: event not found
ubuntu@ubuntu:~$ sudo tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
```



# Allow TCP 80 - OF 1.0

```
[Eddie]Geni3 (1)
Starting Nmap 6.40 ( http://nmap.org ) at 2015-09-16 16:49 KST
Nmap scan report for 200.200.200.12
Host is up (0.00062s latency).
PORT      STATE    SERVICE
80/tcp    closed   http
81/tcp    filtered hosts2-ns
82/tcp    filtered xfer
83/tcp    filtered mit-m1-dev
84/tcp    filtered ctf
85/tcp    filtered mit-m1-dev
86/tcp    filtered mfcobol
87/tcp    filtered priv-term-1
88/tcp    filtered kerberos-sec
89/tcp    filtered su-mit-tg
90/tcp    filtered dnsix

Nmap done: 1 IP address (1 host up) scanned in 1.27 seconds
ubuntu@ubuntu:~$
```

```
[DevServer]
naim@dev-server:~$ ovs-ofctl del-flows $testovs
naim@dev-server:~$ ovs-ofctl add-flow $testovs arp,nw_dst=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs arp,nw_src=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
  cookie=0x0, duration=1.362s, table=2, n_packets=0, n_bytes=0, idle_age=1, arp,arp_spa=200.200.200.0/24 actions=NORMAL
  cookie=0x0, duration=5.364s, table=2, n_packets=0, n_bytes=0, idle_age=5, arp,arp_tpa=200.200.200.0/24 actions=NORMAL
naim@dev-server:~$ ovs-ofctl add-flow $testovs priority=37000,tcp,nw_src=200.200.200.11,tp_dst=80,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs priority=37000,tcp,nw_dst=200.200.200.11,tp_src=80,actions=normal
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
  cookie=0x0, duration=16.01s, table=0, n_packets=0, n_bytes=0, idle_age=16, priority=37000,tcp,nw_src=200.200.200.11,tp_dst=80 actions=NORMAL
  cookie=0x0, duration=95.993s, table=2, n_packets=0, n_bytes=0, idle_age=95, arp,arp_spa=200.200.200.0/24 actions=NORMAL
  cookie=0x0, duration=99.995s, table=2, n_packets=0, n_bytes=0, idle_age=99, arp,arp_tpa=200.200.200.0/24 actions=NORMAL
  cookie=0x0, duration=9.006s, table=0, n_packets=0, n_bytes=0, idle_age=9, priority=37000,tcp,nw_dst=200.200.200.11,tp_src=80 actions=NORMAL
naim@dev-server:~$
```

```
[Eddie]Geni4
ubuntu@ubuntu:~$ sudo tcpdump -i eth0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
16:49:49.971183 IP 200.200.200.11.54012 > 200.200.200.12.http: Flags [S], seq 2637895428, win 29200, options [mss 1460,sackOK,ts val 933568461 ecr 0,nop,wscale 7], length 0
16:49:49.971245 IP 200.200.200.12.http > 200.200.200.11.54012: Flags [R.], seq 0, ack 2637895429, win 0, length 0
16:49:54.984856 ARP, Request who-has 200.200.200.11 tell 200.200.200.12, length 28
16:49:54.985948 ARP, Reply 200.200.200.11 is-at 00:50:56:ad:a2:2e (oui Unknown), length 46
```

- add-flow 후 TCP Port 80~90 까지 패킷을 보내본 결과 TCP Port 80에 대한 패킷만 전달 되는 것을 확인



# Allow All - OF 1.0

---

## Allow All

```
ovs-ofctl add-flow tcp:192.168.99.3:6633 ip,nw_src=200.200.200.11,actions=normal  
ovs-ofctl add-flow tcp:192.168.99.3:6633 ip,nw_dst=200.200.200.12,actions=normal  
ovs-ofctl dump-flows tcp:192.168.99.3:6633
```



# Allow All - OF 1.0

- ovs-ofctl, Geni3, Geni4 사용

The image displays three separate terminal windows, each with a title bar and a scrollable text area.

- [DevServer]**: Shows the output of the command `ovs-ofctl dump-flows $testovs`. The output lists two flow entries (NXST\_FLOW reply) with xid=0x4. Both flows have cookie=0x0, duration=12.007s and 30.584s respectively, table=2, n\_packets=0, n\_bytes=0, idle\_age=12 and 30, arp,arp\_tpa=200.200.200.0/24 actions=NORMAL.
- [Eddie]Geni3 (1)**: Shows the output of the Nmap scan report for host 200.200.200.12. The host is up. It lists various ports (75-85/tcp) as filtered, with services like priv-dial, deos, priv-rje, finger, http, hosts2-ns, xfer, mit-ml-dev, and ctf. The scan took 5.41 seconds.
- [Eddie]Geni4**: Shows the output of the tcpdump session. It starts with a failed attempt to capture on eth0 due to permission denied. After a sudo command, it successfully starts capturing on eth0, showing verbose output suppressed and a link-type of EN10MB (Ethernet) with a capture size of 65535 bytes.



# Allow All - OF 1.0

```
[Eddie]Geni3 (1)
Starting Nmap 6.40 ( http://nmap.org ) at 2015-09-16 16:58 KST
Nmap scan report for 200.200.200.12
Host is up.
PORT      STATE    SERVICE
75/tcp    filtered priv-dial
76/tcp    filtered deos
77/tcp    filtered priv-rje
78/tcp    filtered unknown
79/tcp    filtered finger
80/tcp    filtered http
81/tcp    filtered hosts2-ns
82/tcp    filtered xfer
83/tcp    filtered mit-m1-dev
84/tcp    filtered ctf
85/tcp    filtered mit-m1-dev

Nmap done: 1 IP address (1 host up) scanned in 5.10 seconds
ubuntu@ubuntu:~$
```

```
[DevServer]
naim@dev-server:~$ naim@dev-server:~$ ovs-ofctl del-flows $testovs
naim@dev-server:~$ ovs-ofctl add-flow $testovs arp,nw_dst=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs arp,nw_src=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs ip,nw_src=200.200.200.11,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs ip,nw_dst=200.200.200.12,actions=normal
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
cookie=0x0, duration=5.005s, table=0, n_packets=0, n_bytes=0, idle_age=5, ip,nw_dst=200.200.200.12 actions=NORMAL
cookie=0x0, duration=18.013s, table=2, n_packets=0, n_bytes=0, idle_age=18, arp,arp_spa=200.200.200.0/24 actions=NORMAL
cookie=0x0, duration=31.18s, table=2, n_packets=0, n_bytes=0, idle_age=31, arp,arp_tpa=200.200.200.0/24 actions=NORMAL
cookie=0x0, duration=9.009s, table=0, n_packets=0, n_bytes=0, idle_age=9, ip,nw_src=200.200.200.11 actions=NORMAL
naim@dev-server:~$
```





# Allow All - OF 1.0

---

- ovs-ofctl del-flows \$testovs
- ovs-ofctl add-flow \$testovs arp,nw\_dst=200.200.200.0/24,actions=normal
- ovs-ofctl add-flow \$testovs arp,nw\_src=200.200.200.0/24,actions=normal
- ovs-ofctl add-flow \$testovs ip,nw\_src=200.200.200.11,nw\_dst=200.200.200.12,actions=normal
- ovs-ofctl add-flow \$testovs ip,nw\_src=200.200.200.12,nw\_dst=200.200.200.11,actions=normal
- ovs-ofctl dump-flows \$testovs

-- or --

- ovs-ofctl del-flows \$testovs
- ovs-ofctl add-flow \$testovs arp,nw\_dst=200.200.200.0/24,actions=normal
- ovs-ofctl add-flow \$testovs arp,nw\_src=200.200.200.0/24,actions=normal
- ovs-ofctl add-flow \$testovs ip,nw\_src=200.200.200.12,actions=normal
- ovs-ofctl add-flow \$testovs ip,nw\_dst=200.200.200.12,actions=normal
- ovs-ofctl dump-flows \$testovs

위의 룰 적용 시 정상적으로 양방향 통신이 가능



# Allow All - OF 1.0

```
[Eddie]Geni3 (1)
Starting Nmap 6.40 ( http://nmap.org ) at 2015-09-16 17:15 KST
Nmap scan report for 200.200.200.12
Host is up (0.00024s latency).

PORT      STATE SERVICE
75/tcp    closed priv-dial
76/tcp    closed deos
77/tcp    closed priv-rje
78/tcp    closed unknown
79/tcp    closed finger
80/tcp    closed http
81/tcp    closed hosts2-ns
82/tcp    closed xfer
83/tcp    closed mit-m1-dev
84/tcp    closed ctf
85/tcp    closed mit-m1-dev

Nmap done: 1 IP address (1 host up)

      □ ovs-ofctl del-flows $testovs
      □ ovs-ofctl add-flow $testovs arp,nw_dst=200.200.200.0/24,actions=normal
      □ ovs-ofctl add-flow $testovs arp,nw_src=200.200.200.0/24,actions=normal
      □ ovs-ofctl add-flow $testovs ip,nw_src=200.200.200.11,nw_dst=200.200.200.12,actions=normal
      □ ovs-ofctl add-flow $testovs ip,nw_src=200.200.200.12,nw_dst=200.200.200.11,actions=normal
      □ ovs-ofctl dump-flows $testovs
-- or --
      □ ovs-ofctl del-flows $testovs
      □ ovs-ofctl add-flow $testovs arp,nw_dst=200.200.200.0/24,actions=normal
      □ ovs-ofctl add-flow $testovs arp,nw_src=200.200.200.0/24,actions=normal
      □ ovs-ofctl add-flow $testovs ip,nw_src=200.200.200.12,actions=normal
      □ ovs-ofctl add-flow $testovs ip,nw_dst=200.200.200.12,actions=normal
      □ ovs-ofctl dump-flows $testovs

위의 를 적용 후 정상적으로 양방향 통신 가능 테스트
```

```
[DevServer]
naim@dev-server:~$ ovs-ofctl del-flows $testovs
ovs-ofctl add-flow $testovs arp,nw_dst=200.200.200.0/24,actions=normal
ovs-ofctl add-flow $testovs arp,nw_src=200.200.200.0/24,actions=normal
ovs-ofctl add-flow $testovs ip,nw_src=200.200.200.11,nw_dst=200.200.200.12,actions=normal
ovs-ofctl add-flow $testovs ip,nw_src=200.200.200.12,nw_dst=200.200.200.11,actions=normal
ovs-ofctl dump-flows $testovs
naim@dev-server:~$ ovs-ofctl add-flow $testovs arp,nw_dst=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs arp,nw_src=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs ip,nw_src=200.200.200.11,nw_dst=200.200.200.12,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs ip,nw_src=200.200.200.12,nw_dst=200.200.200.11,actions=normal
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
  cookie=0x0, duration=7.008s, table=0, n_packets=0, n_bytes=0, idle_age=7, ip,nw_src=200.200.200.11,nw_dst=200.200.200.12 actions=NORMAL
  cookie=0x0, duration=4.004s, table=0, n_packets=0, n_bytes=0, idle_age=4, ip,nw_src=200.200.200.12,nw_dst=200.200.200.11 actions=NORMAL
  cookie=0x0, duration=10.01s, table=2, n_packets=0, n_bytes=0, idle_age=10, arp,arp_spa=200.200.200.0/24 actions=NORMAL
  cookie=0x0, duration=13.013s, table=2, n_packets=0, n_bytes=0, idle_age=13, arp,arp_tpa=200.200.200.0/24 actions=NORMAL
naim@dev-server:~$
```





# ARP - OF 1.0

---

## ARP

```
ovs-ofctl add-flow tcp:192.168.99.3:6633 dl_type=0x0806,actions=normal  
ovs-ofctl dump-flows tcp:192.168.99.3:6633
```



# ARP - OF 1.0

- **ovs-ofctl add-flow \$testovs dltype=0x0806,actions=normal**
- 위 룰 적용 결과 sender에서 ping을 전송 시 arp만 전송되고 그 후 icmp 패킷은 전달되지 않음.
  
- **ovs-ofctl add-flow \$testovs arp,actions=normal**
- 위 룰도 같은 결과를 볼 수 있다.

[DevServer]

```
naim@dev-server:~$ ovs-ofctl del-flows $testovs
naim@dev-server:~$ ovs-ofctl add-flow $testovs dltype=0x0806,actions=normal
ovs-ofctl: unknown keyword dltype
naim@dev-server:~$ ovs-ofctl add-flow $testovs dl_type=0x0806,actions=normal
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
  cookie=0x0, duration=15.82s, table=2, n_packets=29, n_bytes=1740, idle_age=0, arp actions=NORMAL
naim@dev-server:~$
```

[Eddie]Geni3 (1)

```
ubuntu@ubuntu:~$ sudo arp -d 200.200.200.12
ubuntu@ubuntu:~$ ping 200.200.200.12
PING 200.200.200.12 (200.200.200.12) 56(84) bytes of data.
^C
--- 200.200.200.12 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4031ms
ubuntu@ubuntu:~$
```



# ARP - OF 1.0

- ARP 테스트 결과: ping을 전송 시 arp 만 전송되고 그 후 icmp 패킷은 전달되지 않음.

```
[Eddie]Geni4
17:21:51.441631 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:21:51.573654 ARP, Request who-has 192.168.100.132 tell 192.168.100.162, length 46
17:21:52.440903 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:21:52.572582 ARP, Request who-has 192.168.100.132 tell 192.168.100.162, length 46
17:21:53.441654 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:21:53.525633 ARP, Request who-has 192.168.100.183 tell 192.168.100.153, length 50
17:21:54.440780 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:21:54.548909 ARP, Request who-has 192.168.100.153 tell 192.168.100.129, length 46
17:21:55.441073 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:21:56.441333 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:21:56.506138 ARP, Request who-has 192.168.100.183 (00:1c:20:00:04:c0) tell 192.168.100.153, length 46
17:21:56.510031 ARP, Request who-has 192.168.100.183 (00:1c:20:00:04:c0) tell 192.168.100.153, length 46
17:21:56.510959 ARP, Request who-has 192.168.100.183 (00:1c:20:00:04:c0) tell 192.168.100.153, length 46
17:21:56.518226 ARP, Request who-has 192.168.100.183 (00:1c:20:00:04:c0) tell 192.168.100.153, length 46
17:21:56.518321 ARP, Request who-has 192.168.100.183 (00:1c:20:00:04:c0) tell 192.168.100.153, length 46
17:21:57.441543 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:21:58.440495 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:21:58.949584 ARP, Request who-has 192.168.100.134 tell 192.168.100.2, length 46
^C
125 packets captured
127 packets received by filter
0 packets dropped by kernel
ubuntu@ubuntu:~$ sudo tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
17:22:16.440645 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:16.580308 ARP, Request who-has 192.168.100.132 tell 192.168.100.162, length 46
17:22:17.440563 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:17.576608 ARP, Request who-has 192.168.100.132 tell 192.168.100.162, length 46
17:22:18.440151 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:18.572143 ARP, Request who-has 192.168.100.132 tell 192.168.100.162, length 46
17:22:19.440702 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:19.902640 ARP, Request who-has 200.200.200.12 tell 200.200.200.11, length 46
17:22:19.902658 ARP, Reply 200.200.200.12 is-at 00:50:56:ad:de:bd, length 28
17:22:20.440677 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:21.440421 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:22.440024 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:23.441049 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:24.439877 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
17:22:25.440021 ARP, Request who-has 192.168.100.132 tell 192.168.100.128, length 46
^C
15 packets captured
16 packets received by filter
0 packets dropped by kernel
ubuntu@ubuntu:~$
```



# Block TCP 80

---

## Block TCP 80

```
#ovs-ofctl add-flow tcp:192.168.99.3:6633 tcp,nw_src=200.200.200.0,tp_dst=80,actions=TABLE
#ovs-ofctl add-flow tcp:192.168.99.3:6633 tcp,nw_src=200.200.200.0/24,tp_dst=80,actions=mod_nw_dst:200.200.200.12,mod_dl_dst:d8:50:e6:57:b8:ba
#ovs-ofctl add-flow tcp:192.168.99.3:6633 priority=45000,tcp,tp_dst=80,actions=mod_nw_dst:200.200.200.12,mod_dl_dst:ac:22:0b:85:5b:63,output:17
#ovs-ofctl add-flow tcp:192.168.99.3:6633 priority=45000,tcp,tp_dst=80,actions=mod_dl_dst:cc:cc:cc:cc:cc:cc,output:17

ovs-ofctl add-flow tcp:192.168.99.3:6633 "tcp,tp_dst=80,actions=mod_dl_dst:ac:22:0b:85:5b:63"

#ovs-ofctl add-group tcp:192.168.99.3:6633 group_id=1,type=indirect,bucket=mod_dl_dst:33:33:33:33:33:33

#ovs-ofctl add-flow tcp:192.168.99.3:6633 priority=45000,tcp,tp_dst=80,actions=output:17
#ovs-ofctl add-flow tcp:192.168.99.3:6633 priority=45000,tcp,in_port=17,actions=NORMAL
#ovs-ofctl add-flow tcp:192.168.99.3:6633 tcp,nw_src=200.200.200.0,tp_dst=80,actions=LOCAL
ovs-ofctl dump-flows tcp:192.168.99.3:6633
```



# Block TCP 80 - OF 1.0

- 제목이 block인데 내용은 modify 이어서 drop 으로 변경 테스트

- ovs-ofctl del-flows \$testovs
- ovs-ofctl add-flow \$testovs arp,nw\_dst=200.200.200.0/24,actions=normal
- ovs-ofctl add-flow \$testovs arp,nw\_src=200.200.200.0/24,actions=normal
- ovs-ofctl add-flow \$testovs priority=3000,ip,nw\_src=200.200.200.0/24,actions=normal
- ovs-ofctl add-flow \$testovs priority=3000,ip,nw\_dst=200.200.200.0/24,actions=normal
- ovs-ofctl add-flow \$testovs priority=3001,tcp,tp\_dst=80,actions=drop
- ovs-ofctl dump-flows \$testovs

- drop률의 priority가 allow의 priority 보다 높아야 정상 동작.

```
[DevServer]
naim@dev-server:~$ ovs-ofctl del-flows $testovs
ovs-ofctl add-flow $testovs arp,nw_src=200.200.200.0/24,actions=normal
ovs-ofctl add-flow $testovs priority=3000,ip,nw_src=200.200.200.0/24,actions=normal
ovs-ofctl add-flow $testovs priority=3000,ip,nw_dst=200.200.200.0/24,actions=normal
ovs-ofctl add-flow $testovs priority=3001,tcp,tp_dst=80,actions=drop
ovs-ofctl dump-flows $testovs
naim@dev-server:~$ ovs-ofctl add-flow $testovs arp,nw_dst=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs arp,nw_src=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs priority=3000,ip,nw_src=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs priority=3000,ip,nw_dst=200.200.200.0/24,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs priority=3001,tcp,tp_dst=80,actions=drop
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
  cookie=0x0, duration=4.004s, table=0, n_packets=0, n_bytes=0, idle_age=4, priority=3001,tcp,tp_dst=80 actions=drop
  cookie=0x0, duration=10.011s, table=0, n_packets=0, n_bytes=0, idle_age=10, priority=3000,ip,nw_src=200.200.200.0/24 actions=NORMAL
  cookie=0x0, duration=13.014s, table=2, n_packets=0, n_bytes=0, idle_age=13, arp,arp_spa=200.200.200.0/24 actions=NORMAL
  cookie=0x0, duration=16.016s, table=2, n_packets=0, n_bytes=0, idle_age=16, arp,arp_tpa=200.200.200.0/24 actions=NORMAL
  cookie=0x0, duration=7.008s, table=0, n_packets=0, n_bytes=0, idle_age=7, priority=3000,ip,nw_dst=200.200.200.0/24 actions=NORMAL
naim@dev-server:~$
```





# Block Basic - OF 1.0

---

## Block Basic

```
#ovs-ofctl add-flow tcp:192.168.99.3:6633 ip,nw_src=200.200.200.0/24,actions=drop
ovs-ofctl add-flow tcp:192.168.99.3:6633 priority=30000,dl_type=0x0800,actions=drop
ovs-ofctl dump-flows tcp:192.168.99.3:6633
```



# Block Basic - OF 1.0

- arp 패킷은 전달 되나, ip 프로토콜에 속하는 icmp 패킷은 전달 되지 않음.
- 패킷을 전달하는 룰보다 priority가 높아야함.

The image displays three terminal windows illustrating the configuration and testing of OpenFlow 1.0 rules on a DevServer and two Ubuntu hosts (Eddie and Geni).

- [DevServer]**: Shows the configuration of flow rules. The user runs `ovs-ofctl del-flows $testovs`, followed by `add-flow $testovs priority=29999,actions=normal`, `add-flow $testovs priority=30000,dl_type=0x800,actions=drop`, and `dump-flows $testovs`. The output includes details about the NXST\_FLOW reply (xid=0x4) for both priority levels.
- [Eddie]Geni3 (1)**: Shows the user attempting to delete ARP entries with `arp -d 200.200.200.12` and failing with "Operation not permitted". Then, they ping the target IP and receive a 100% packet loss response. They then use sudo to ping and get a successful response.
- [Eddie]Geni4**: Shows the user using `tcpdump -i eth0 src host 200.200.200.11` to capture ARP requests. The output shows three ARP requests from the target IP (200.200.200.12) to the source IP (200.200.200.11), each with a length of 46 bytes.



# Block ICMP - OF 1.0

---

## Block ICMP

```
ovs-ofctl add-flow tcp:192.168.99.3:6633 icmp,nw_src=200.200.200.0/24,actions=drop  
ovs-ofctl dump-flows tcp:192.168.99.3:6633
```



# Block ICMP - OF 1.0

- arp, tcp의 다른패킷은 전달되나 icmp패킷은 전달되지 않음.

```
[Eddie]Geni3 (1)
--- 200.200.200.12 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9072ms
ubuntu@ubuntu:~$ nmap 200.200.200.12 -p 75-85 -PO
Starting Nmap 6.40 ( http://nmap.org ) at 2015-09-16 18:13 KST
Nmap scan report for 200.200.200.12
Host is up (0.00039s latency).
PORT      STATE SERVICE
75/tcp    closed  priv-dial
76/tcp    closed  deos
77/tcp    closed  priv-rje
78/tcp    closed  unknown
79/tcp    closed  finger
80/tcp    closed  http
81/tcp    closed  hosts2-ns
82/tcp    closed  xfer
83/tcp    closed  mit-m1-dev
84/tcp    closed  ctf
85/tcp    closed  mit-m1-dev

Nmap done: 1 IP address (1 host up) scanned in 0.42 seconds
ubuntu@ubuntu:~$ ping 200.200.200.12
PING 200.200.200.12 (200.200.200.12) 56(84) bytes of data.
^C
--- 200.200.200.12 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2015ms
ubuntu@ubuntu:~$
```

```
[DevServer]
naim@dev-server:~$ ovs-ofctl del-flows $testovs
ovs-ofctl add-flow $testovs priority=29999,actions=normal
ovs-ofctl add-flow $testovs priority=30000,icmp,nw_src=200.200.200.0/24,actions=drop
ovs-ofctl dump-flows $testovs

naim@dev-server:~$ ovs-ofctl add-flow $testovs priority=29999,actions=normal
naim@dev-server:~$ ovs-ofctl add-flow $testovs priority=30000,icmp,nw_src=200.200.200.0/24,actions=drop
naim@dev-server:~$ ovs-ofctl dump-flows $testovs
NXST_FLOW reply (xid=0x4):
  cookie=0x0, duration=4.004s, table=0, n_packets=0, n_bytes=0, idle_age=4, priority=30000,icmp,nw_src=200.200.200.0/24 actions=drop
  cookie=0x0, duration=7.007s, table=0, n_packets=1, n_bytes=0, idle_age=7, priority=29999 actions=NORMAL
naim@dev-server:~$
```



# Block ICMP - OF 1.0

- arp, tcp의 다른패킷은 전달되나 icmp패킷은 전달되지 않음.

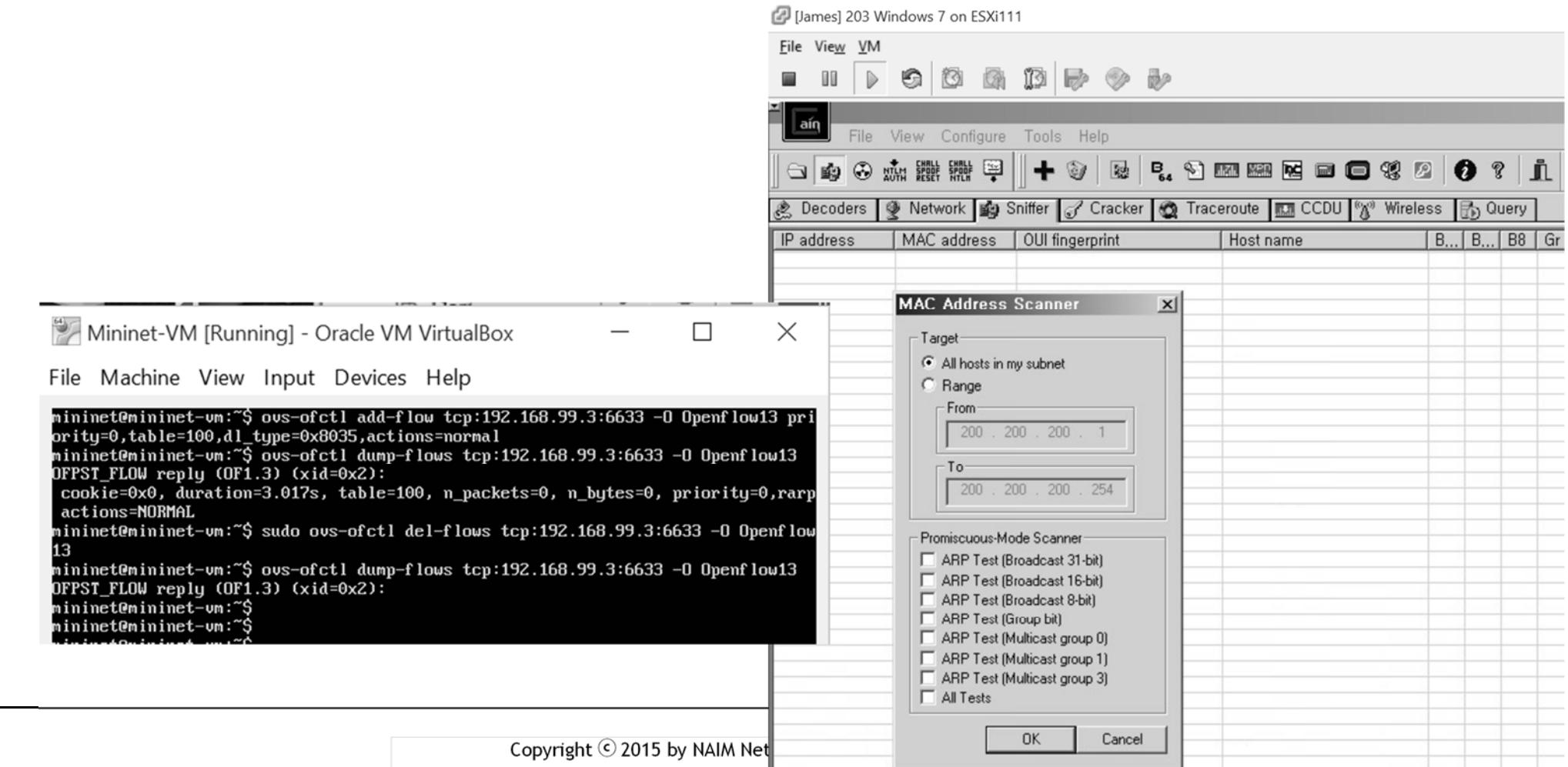
```
[Eddie]Geni4
ubuntu@ubuntu:~$ sudo tcpdump -i eth0 src host 200.200.200.11
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
18:12:55.963184 ARP, Request who-has 200.200.200.12 tell 200.200.200.11, length 46
18:13:16.840092 IP 200.200.200.11.44411 > 200.200.200.12.78: Flags [S], seq 3297172188, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840150 IP 200.200.200.11.43018 > 200.200.200.12.84: Flags [S], seq 2567331215, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840163 IP 200.200.200.11.56342 > 200.200.200.12.81: Flags [S], seq 441325966, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840209 IP 200.200.200.11.42506 > 200.200.200.12.82: Flags [S], seq 3966133440, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840331 IP 200.200.200.11.55970 > 200.200.200.12.75: Flags [S], seq 3061828456, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840458 IP 200.200.200.11.54671 > 200.200.200.12.http: Flags [S], seq 1283995516, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840495 IP 200.200.200.11.54801 > 200.200.200.12.76: Flags [S], seq 1599372403, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840504 IP 200.200.200.11.48289 > 200.200.200.12.83: Flags [S], seq 1224264102, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840511 IP 200.200.200.11.36110 > 200.200.200.12.rje: Flags [S], seq 3470576419, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840524 IP 200.200.200.11.42690 > 200.200.200.12.85: Flags [S], seq 1351485438, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:16.840544 IP 200.200.200.11.51189 > 200.200.200.12.finger: Flags [S], seq 1079904666, win 29200, options [mss 1460,sackOK,TS val 934820179 ecr 0,nop,wscale 7], length 0
18:13:21.849029 ARP, Reply 200.200.200.11 is-at 00:50:56:ad:a2:2e (oui Unknown), length 46
```



# RARP

- RARP 허용하지 않으면 Security Tool (Cain&Abel) 동작 하지 않음

- `./ovs-ofctl dump-flows tcp:192.168.99.3:6633 -O Openflow13`

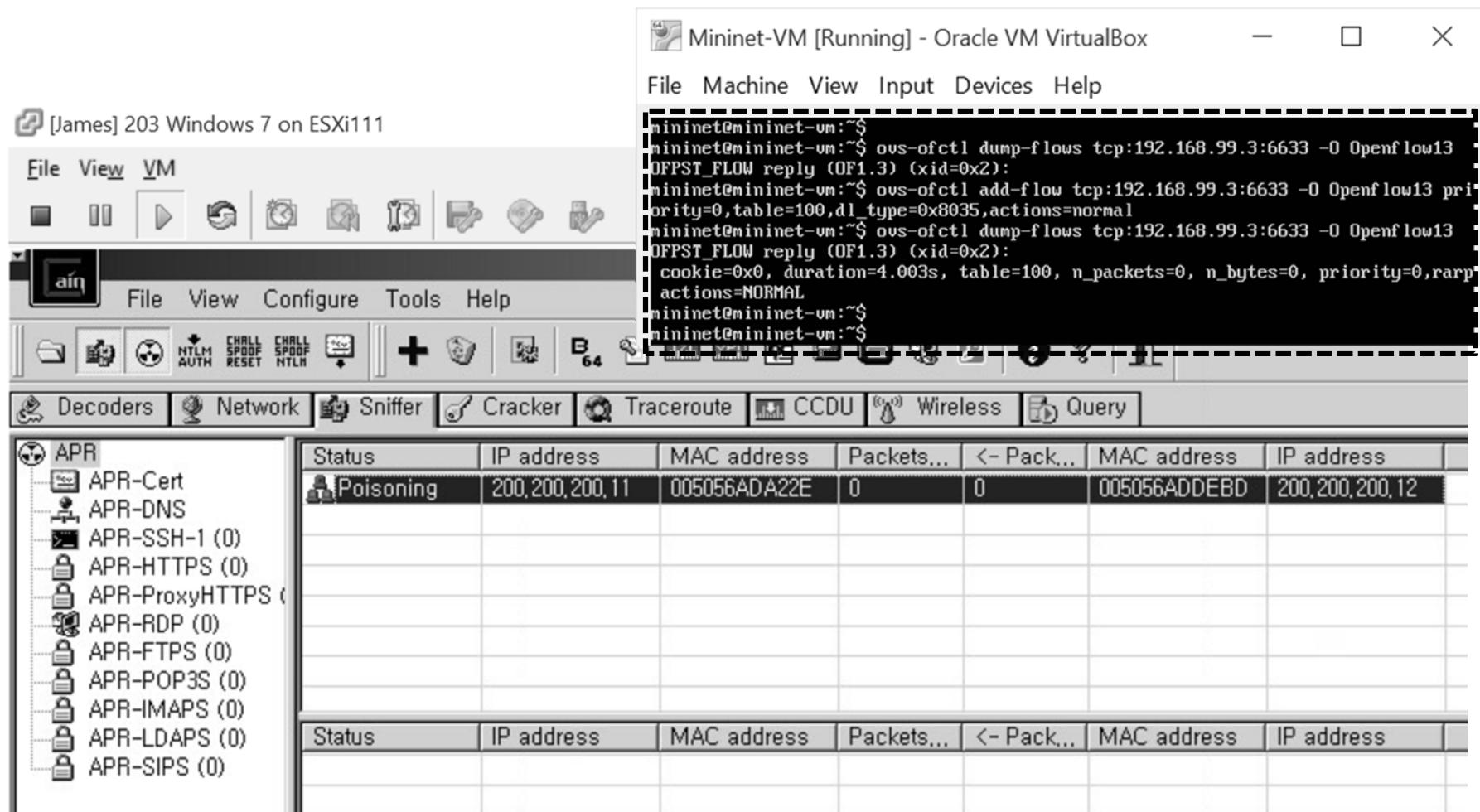




# RARP

- RARP 허용하면 Security Tool (Cain&Abel) 동작함

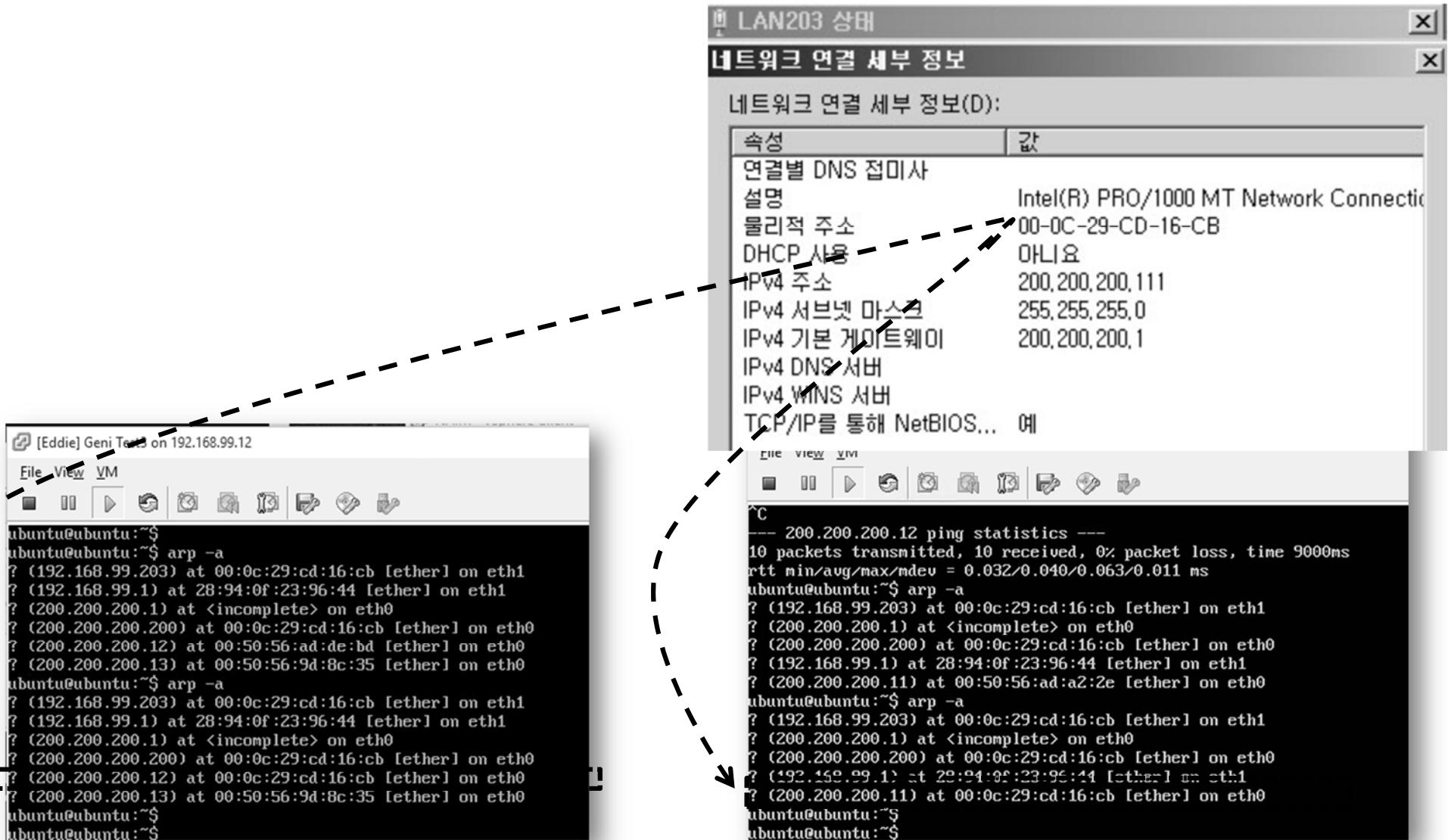
- `./ovs-ofctl dump-flows tcp:192.168.99.3:6633 -O Openflow13`





# RARP

- RARP 허용하면 Security Tool (Cain&Abel)에 의해 ARP 변경 가능





# Thank you very much

