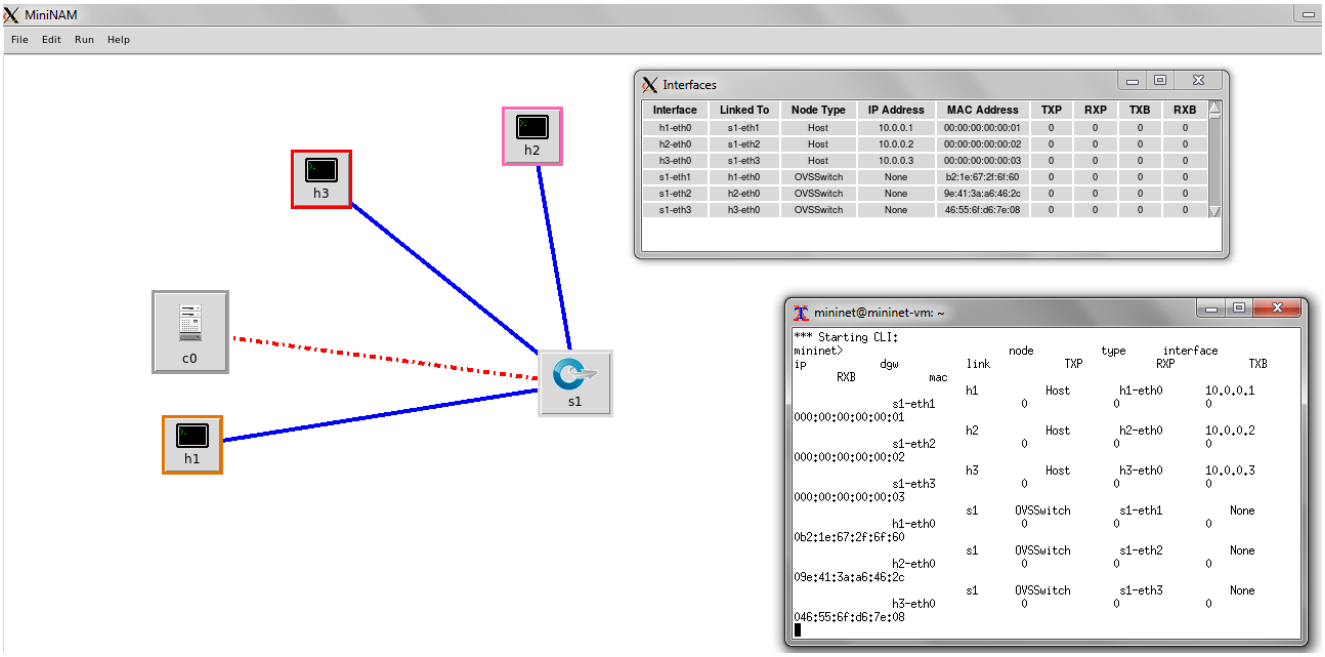
Desi Tambunan **161402005**
Sonia Elisa Telaumbanua **161402045**
Administrasi & Desain Jaringan **Novalina Simbolon** **161402077**
Chyntia Claudia **161402101**
OPENFLOW **Tirza Priskila Kinanti Sibuea** **161402110**

OpenFlow adalah sebuah protokol yang memungkinkan pengaturan penjaluran dan pengiriman paket ketika melalui sebuah *switch*. Dalam sebuah jaringan konvensional, setiap *switch* hanya berfungsi meneruskan paket yang lewat ke port yang sesuai tanpa dapat membedakan tipe protokol data yang dikirimkan, misalnya *elastic* atau *inelastic traffic*.

Berikut contoh tahap-tahapannya;

1. mininet@mininet~vm:~$ sudo mn --topo single,3 --mac --switch ovsk --controller remote

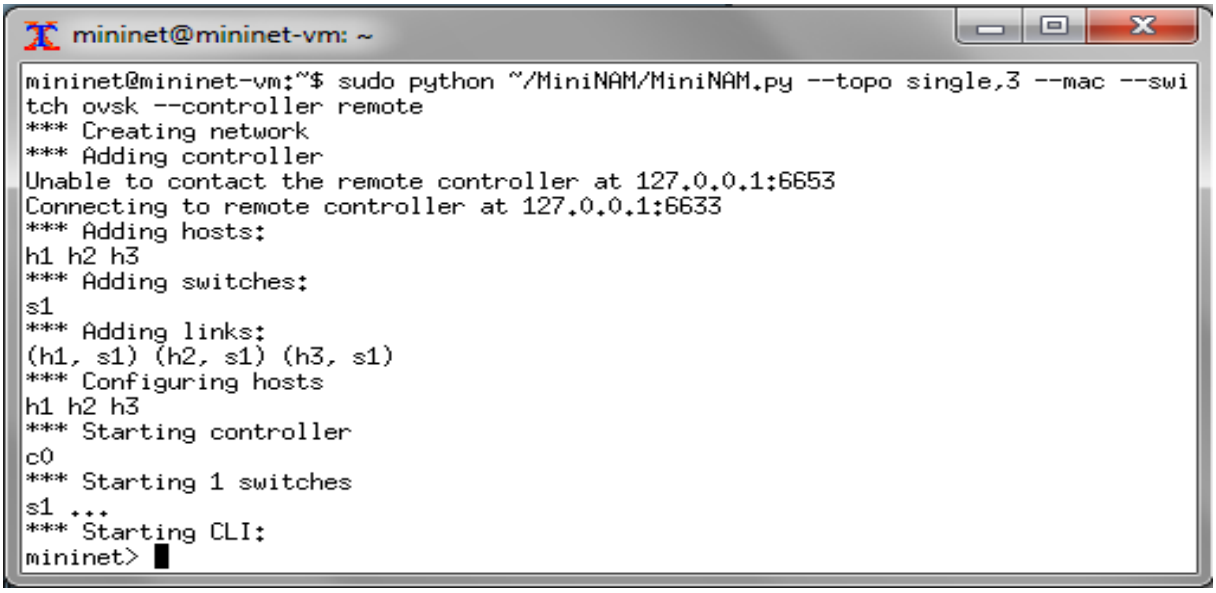
   Membuat 3 *host* dengan 1 *switch*

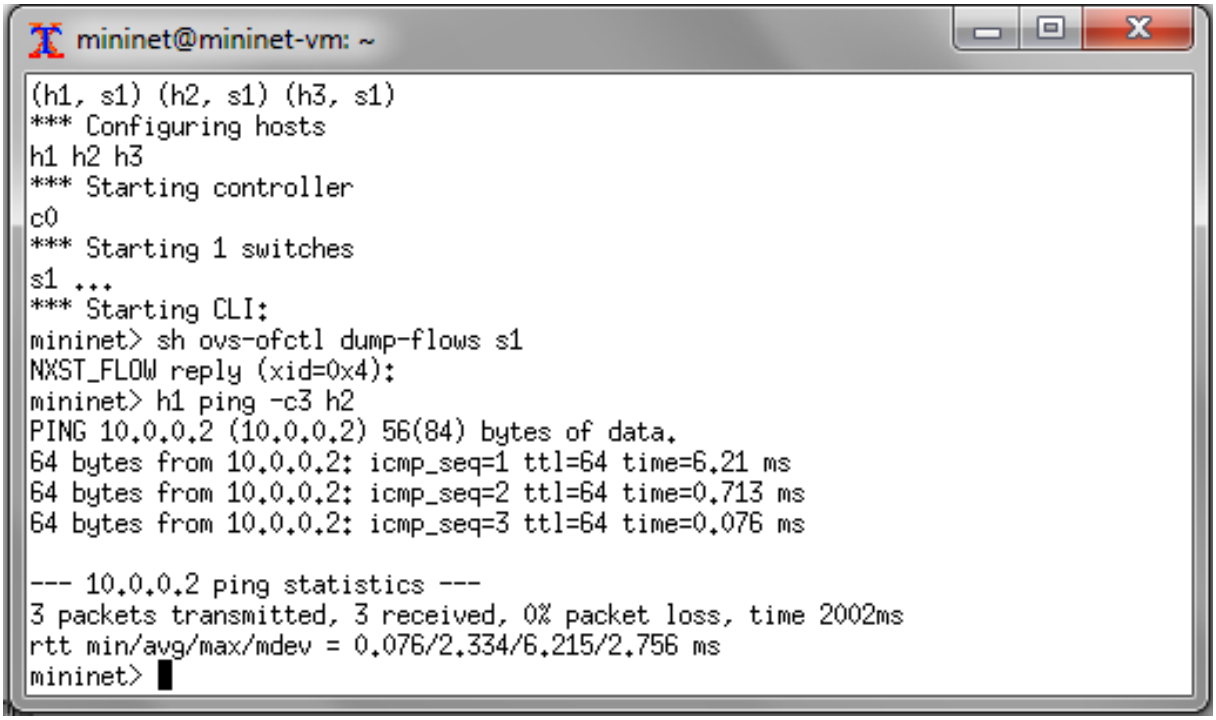


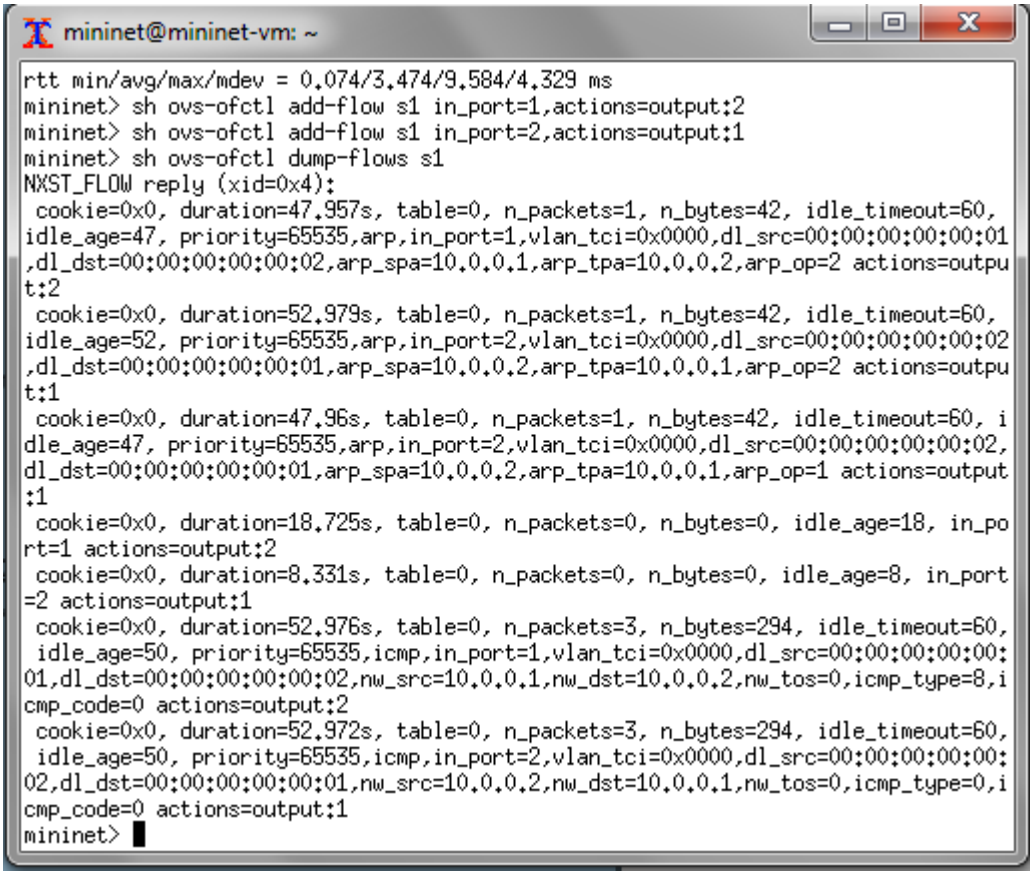
2. Manual Flow Entry

   mininet@mininet~vm:~$ sudo python ~/MiniNAM/MiniNAM.py --topo single,3 --mac --switch ovsk --controller remote

   mininet> sh ovs-ofctl dump-flows s1

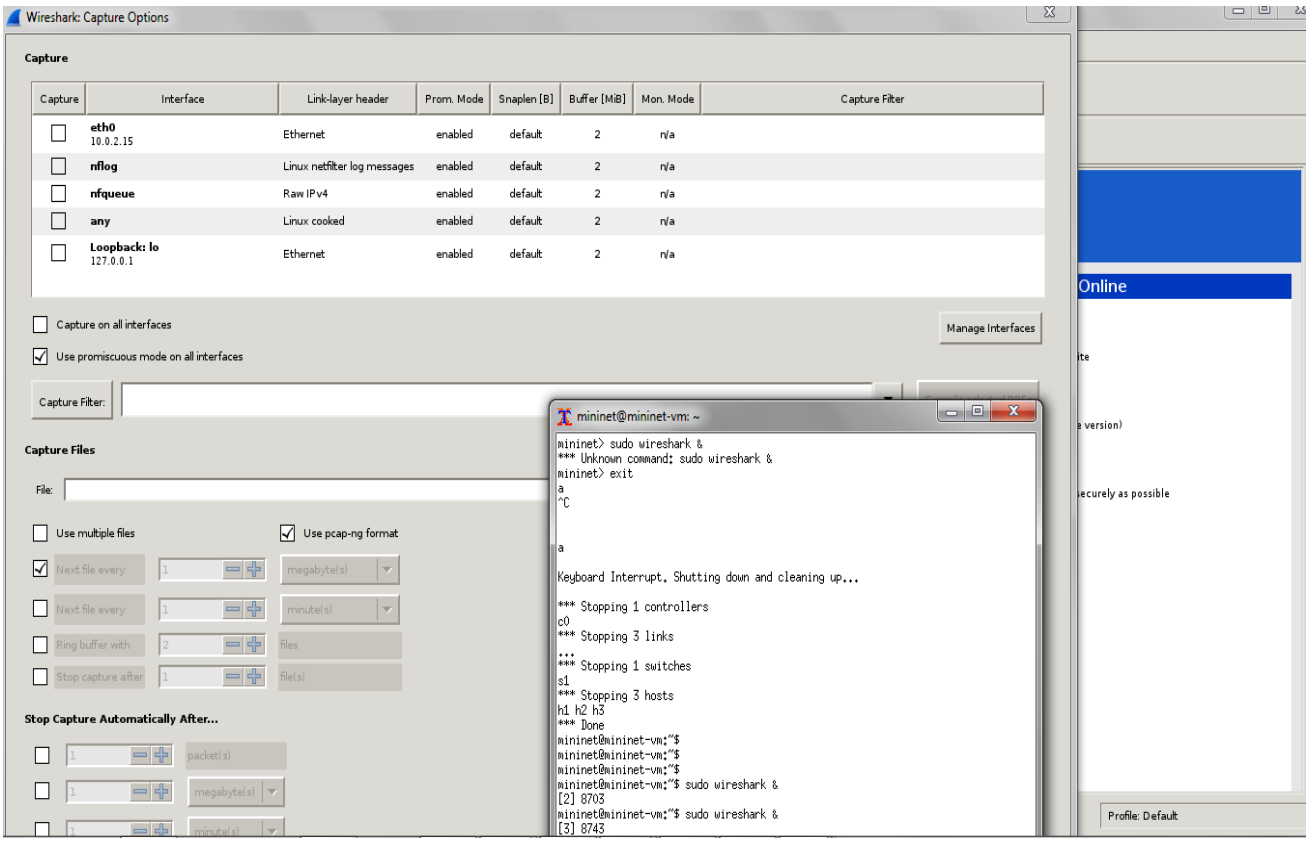

mininet@mininet-vm: ~
mininet@mininet-vm:~$ sudo python ~/MiniNAM/MiniNAM.py --topo single,3 --mac --swi
tch ovsk --controller remote
*** Creating network
*** Adding controller
Unable to contact the remote controller at 127.0.0.1:6653
Connecting to remote controller at 127.0.0.1:6633
*** Adding hosts:
h1 h2 h3
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1) (h3, s1)
*** Configuring hosts
h1 h2 h3
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet>


mininet@mininet-vm: ~
(h1, s1) (h2, s1) (h3, s1)
*** Configuring hosts
h1 h2 h3
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> sh ovs-ofctl dump-flows s1
NXST_FLOW reply (xid=0x4):
mininet> h1 ping -c3 h2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=6.21 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.713 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.076 ms
--- 10.0.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.076/2.334/6.215/2.756 ms
mininet>

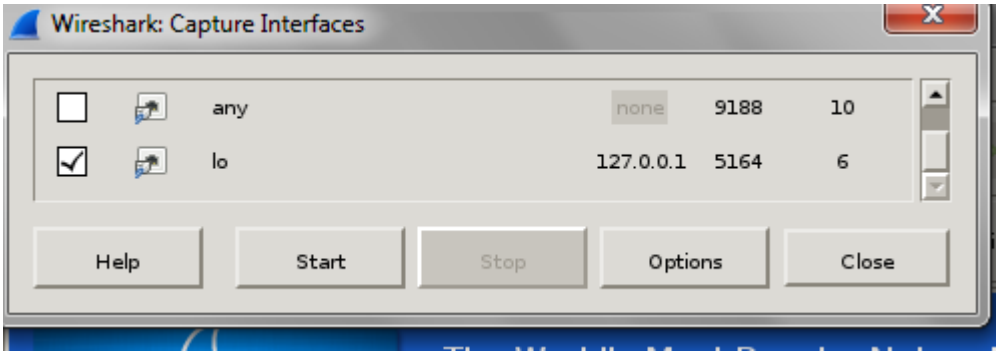


3. Wireshark

mininet@mininet~vm:~$ sudo wireshark &



Klik ‘Capture’ lalu pada *menubar* di atas lalu pilih ‘Capture Interfaces’ dan pilih (centang) ‘lo’ kemudian mulai (‘Start’).



4. Starting Controller (OF Reference Controller)

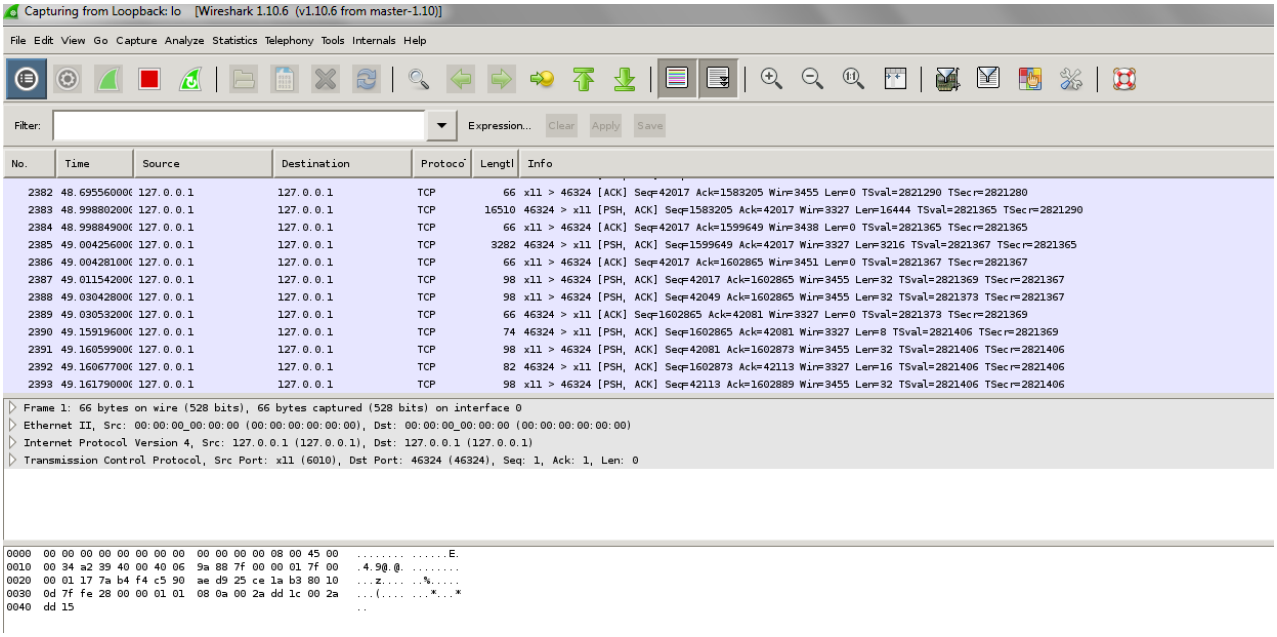
mininet@mininet~vm:~$ controller ptcp: &

```
mininet@mininet-vm:~$ controller ptcp: &
[4] 9119
mininet@mininet-vm:~$ Apr 07 03:45:20|00001|vconn_tcp|ERR|ptcp:: bind: Address alr
eady in use
Apr 07 03:45:20|00002|controller|ERR|ptcp:: connect: Address already in use
mininet@mininet-vm:~$ sudo mn --topo single,3 --mac --switch ovsk --controller rem
ote
*** Creating network
*** Adding controller
Unable to contact the remote controller at 127.0.0.1:6653
Connecting to remote controller at 127.0.0.1:6633
*** Adding hosts:
h1 h2 h3
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1) (h3, s1)
*** Configuring hosts
h1 h2 h3
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet>
```

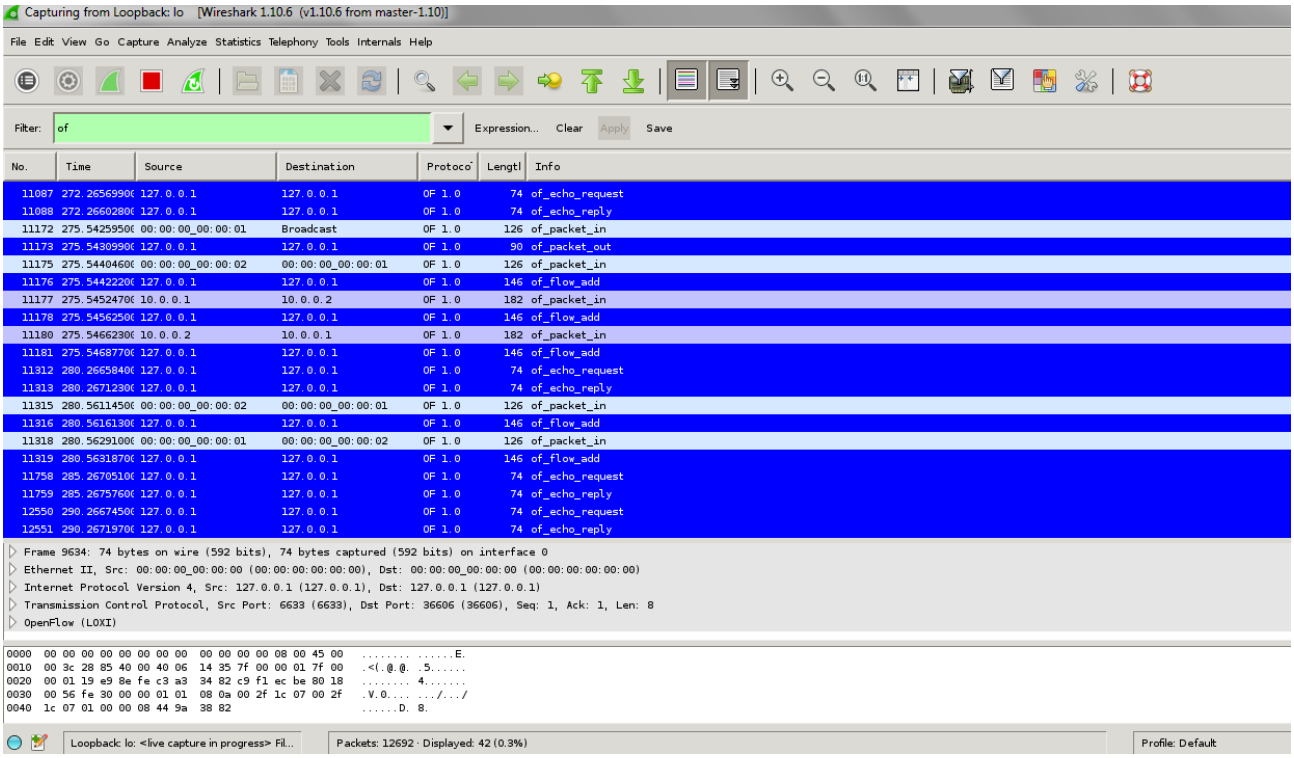
```
mininet@mininet-vm: ~
mininet> h1 ip -s -s neigh flush all
Nothing to flush.
mininet> h2 ip -s -s neigh flush all
Nothing to flush.
mininet> sh ovs-ofctl del-flows s1
mininet> h1 ping -c1 h2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=6.39 ms

--- 10.0.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 6.393/6.393/6.393/0.000 ms
mininet>
```

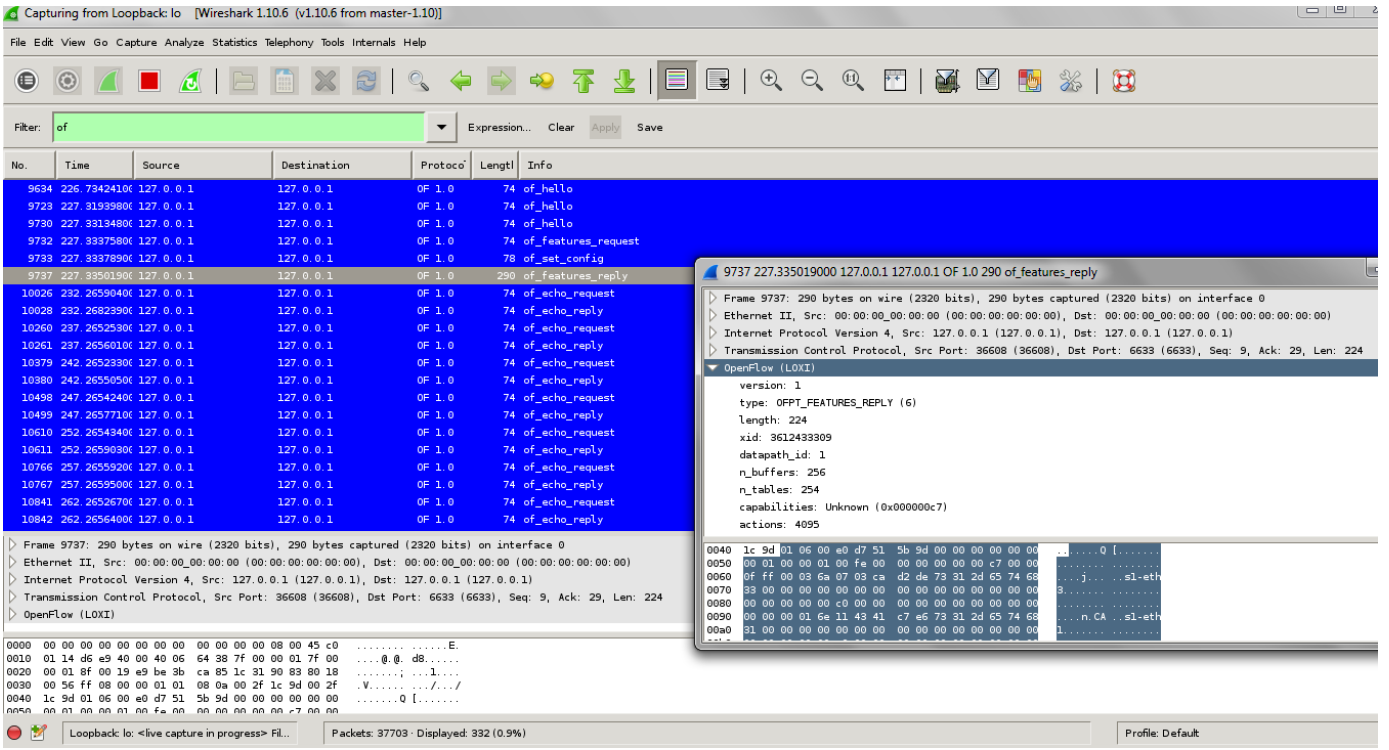
Berikut tampilan pada wireshark;



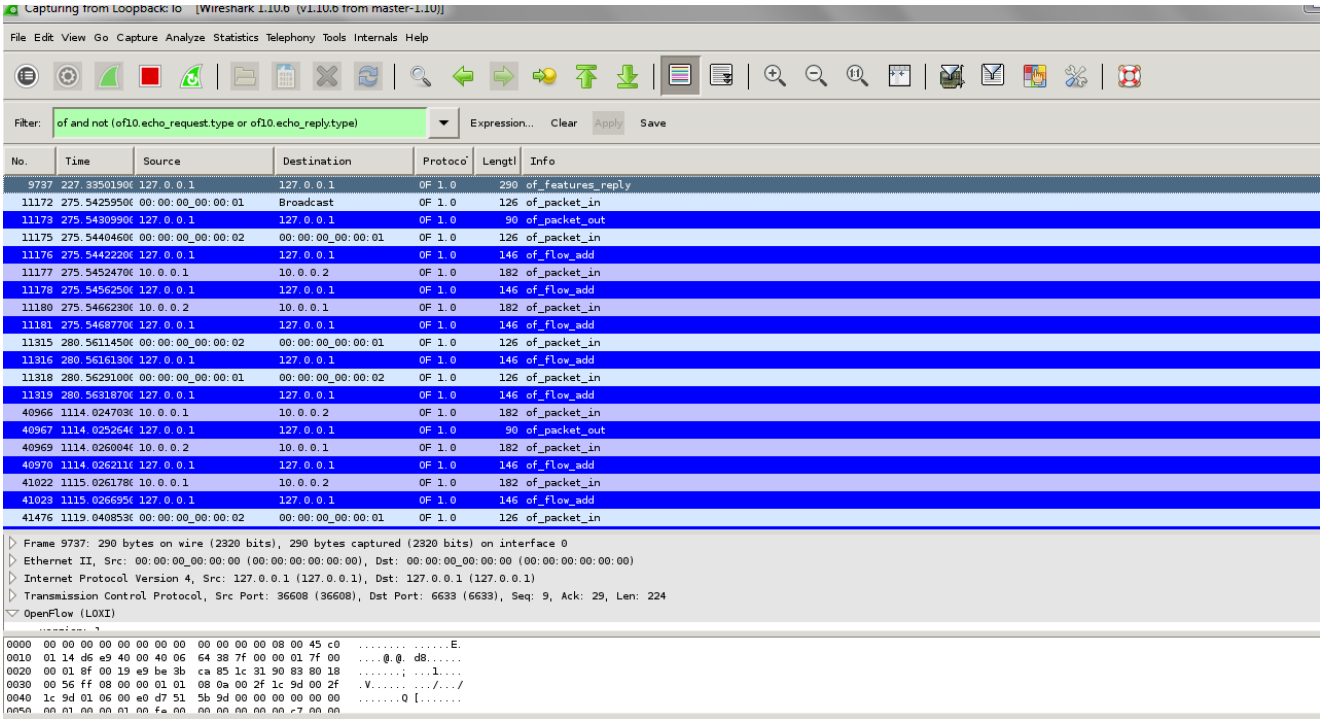
5. Lakukan Filter ‘OF’ pada Wireshark



6. Inspect Packet



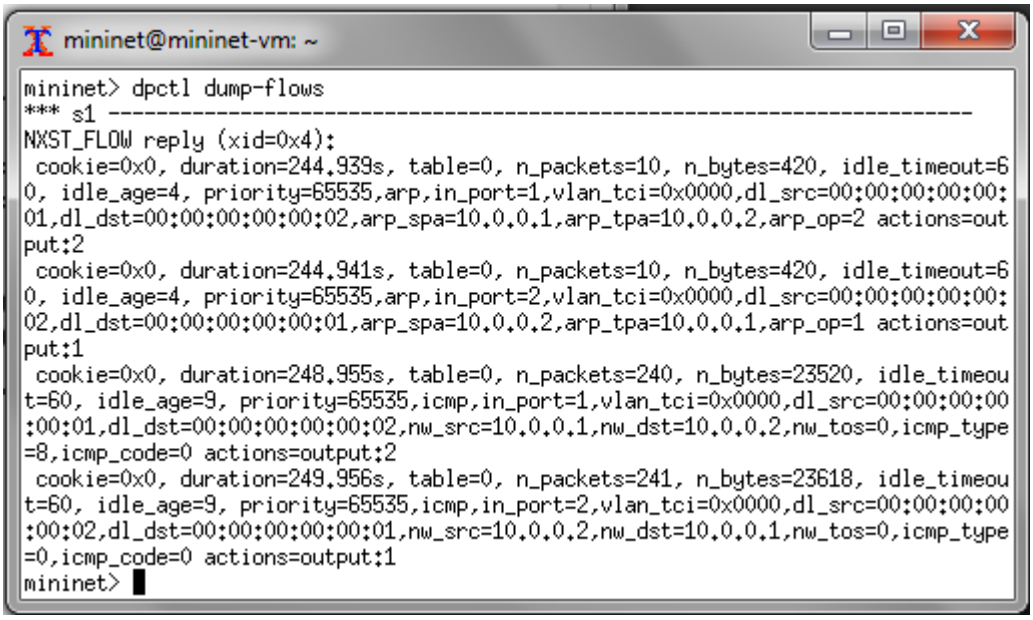
7. Filter ‘of and not (of10.echo_request.type or of10.echo_reply.type)’ pada tampilan WireShark setelah ‘h1 ping –c1 h2’.



8. Flow Entries

mininet> dpctl dump-flows

mininet@mininet~vm:~$ sudo ovs-ofctl dump-flows s1



9. Benchmark kernel- vs. user-space