

Administrasi dan Design Jaringan

Mininet, MiniNAM, OpenFlow, dan PoxController



Oleh :

Dewi Safrida (161402008)

Khairunnisa Sitanggang (161402014)

Bora Sejati Siboro (161402020)

Haryati (161402047)

Mayang Dyah Azurah (161402050)

Fakultas Ilmu Komputer dan Teknologi Informasi

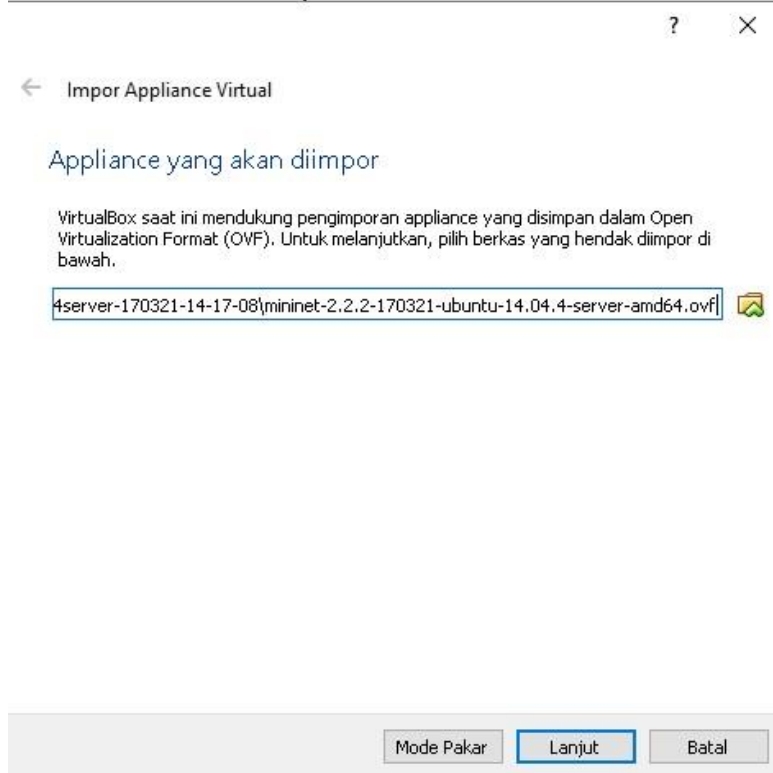
Universitas Sumatera Utara

2018

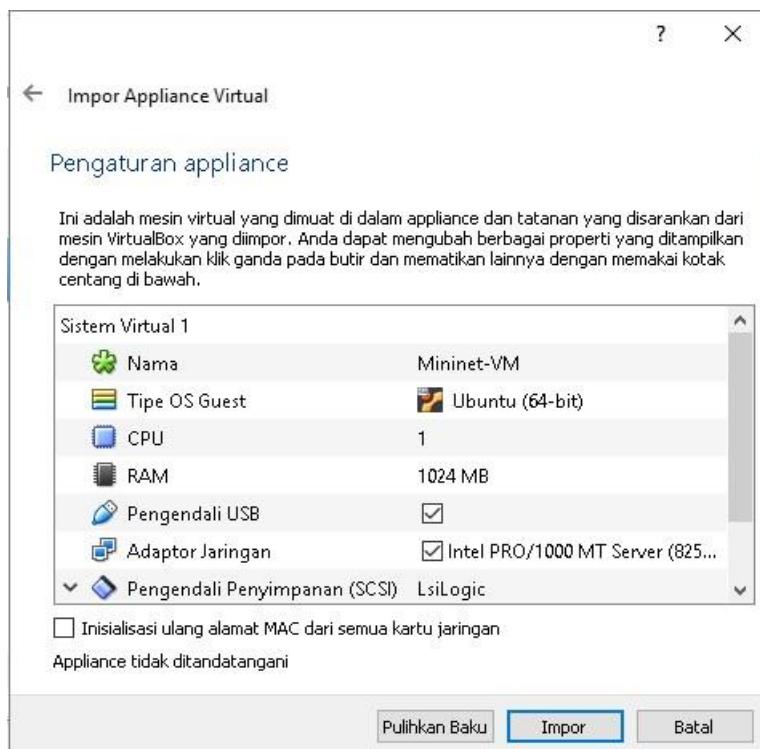
Mininet

Mininet adalah emulator berbasis CLI yang digunakan untuk membuat sebuah topologi jaringan pada Software Defined Network. Pada Mininet sudah terdapat beberapa topologi bawaan yang dapat langsung digunakan dengan menggunakan perintah (command) tertentu. Tahap instalasi mininet:

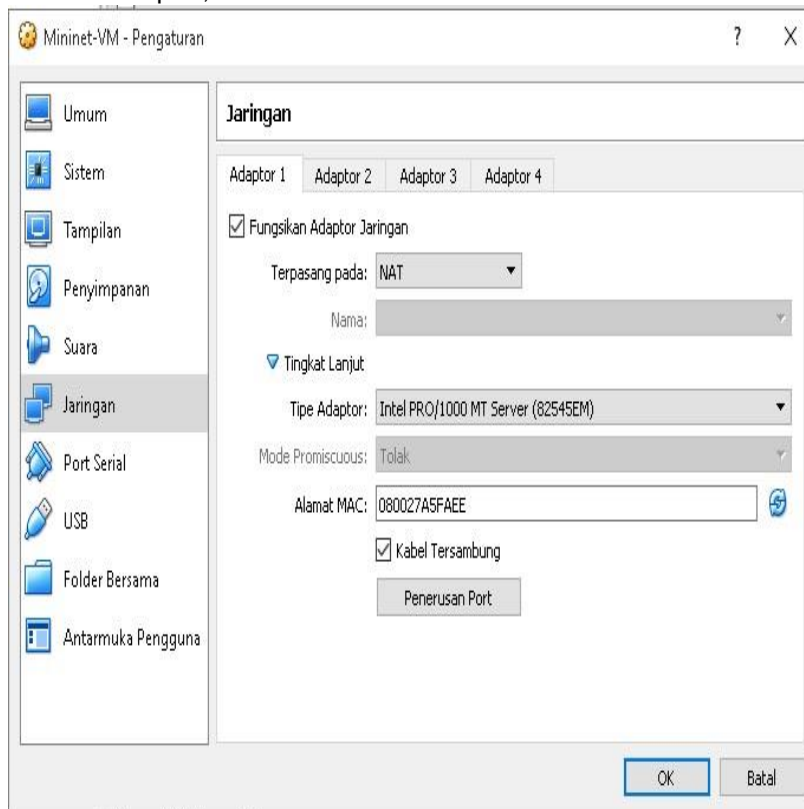
1. Buka virtual box, setelah itu pilih Impor Appliance Virtual kemudian pilih mininetnya



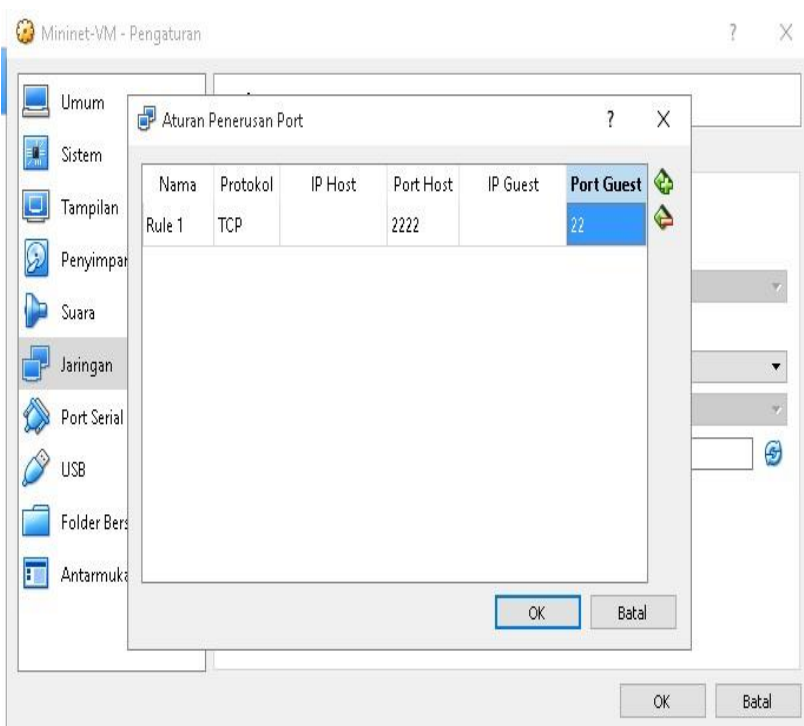
2. Kemudian, impor installer mininetnya



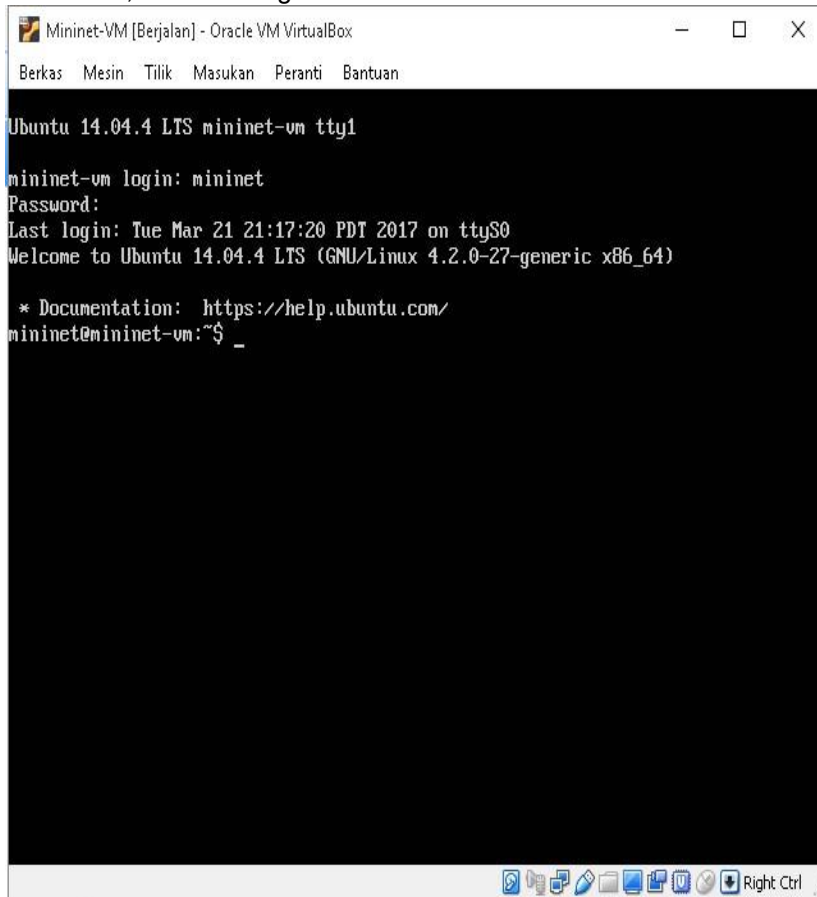
3. setelah di impor , klik ok



4. Kemudian isi table **Port Host** dan **Port Guest**



5. setelah itu, **LOGIN** dengan Username “mininet” dan Password “mininet”



miniNAM

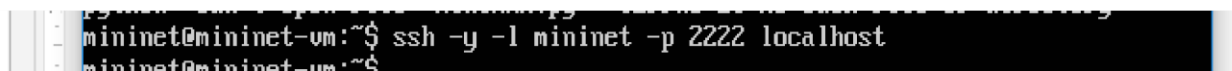
Mininam merupakan alat yang sangat kuat untuk debug protokol jaringan dimana ia berfungsi sebagai penyedia antarmuka pengguna grafis yang memungkinkan modifikasi dinamis preferensi dan filter paket yang sangat membantu pada proses memahami konsep jaringan. Tahap instalasi miniNAM:

Setelah penginstalan Mininet di Virtual Box selesai, kita akan melanjutkan untuk menginstal Mininam. Supaya lebih baik, jangan menutup mininet yang tadi sudah kita buat. Tetap lanjutkan penginstalan mininam pada halaman kerja yang sama.

1. Pertama – tama ketikkan sebagai berikut :

ssh -y -l mininet -p 2222 localhost

maka, akan muncul tampilan seperti gambar dibawah



2. Setelah berhasil, lanjutkan dengan mengetik

sudo apt-get install git python-imaging python-imaging-tk

```
mininet@mininet-vm:~$ sudo apt-get install git python-imaging python-imaging-tk
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package python-imaging
E: Unable to locate package python-imaging-tk
mininet@mininet-vm:~$
```

3. Kemudian kita akan melakukan clone dengan mengetikkan perintah sebagai berikut

git clone <https://github.com/uccmis/MiniNAM.git>

pada screenshot an kami pada perintah ini, terjadi fatal yang mengatakan “MiniNAM” already exists dikarenakan kami sudah pernah melakukan perintah ini sebelumnya dan belum melakukan screenshot makanya terjadi hal fatal tersebut.

```
mininet@mininet-vm:~$ git clone https://github.com/uccmis/MiniNAM.git
fatal: destination path 'MiniNAM' already exists and is not an empty directory.
mininet@mininet-vm:~$
```

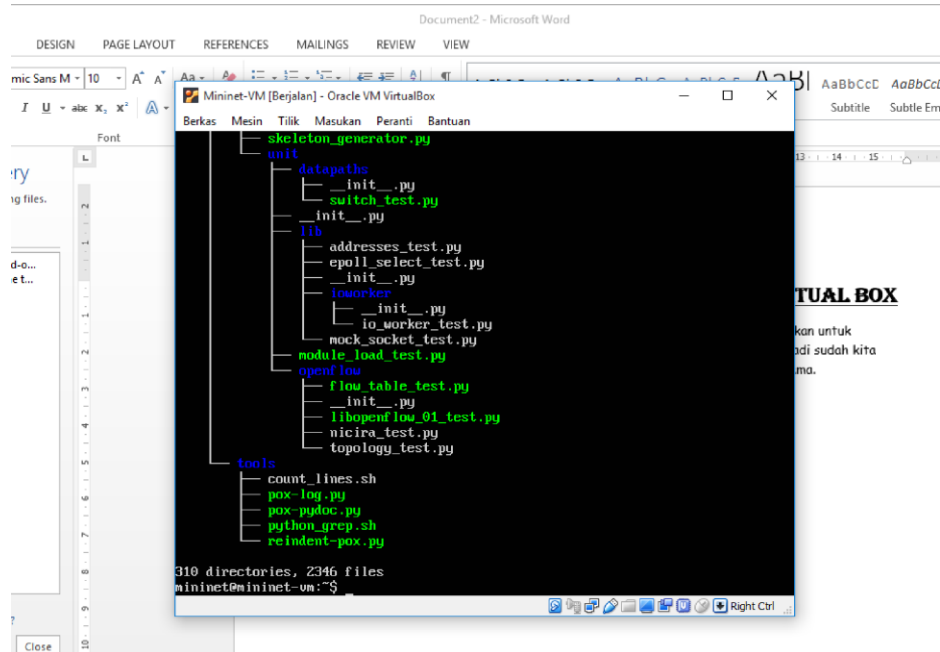
4. Kemudian kita ketikkan perintah selanjutnya seperti ini

sudo apt-get install tree

kegunaan dari perintah tersebut adalah untuk memulai melakukan penginstalan MiniNAM dengan apt-get install tree

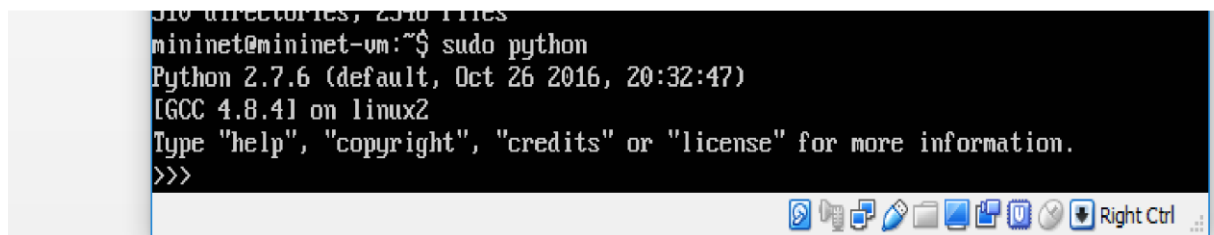
```
mininet@mininet-vm:~$ sudo apt-get install tree
Reading package lists... Done
Building dependency tree
Reading state information... Done
tree is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 98 not upgraded.
mininet@mininet-vm:~$
```

5. Kemudian setelah apt-get install tree berhasil, maka akan muncul tulisan **“tree is already the newest version”** dengan maksud sudah terinstal dengan versi terbaru sesuai pada gambar nomor 4. Kemudian kita ketikkan **“tree”** untuk melakukan pengecekan. Jika sudah berhasil, maka akan muncul tampilan seperti skema tree seperti pada gambar dibawah

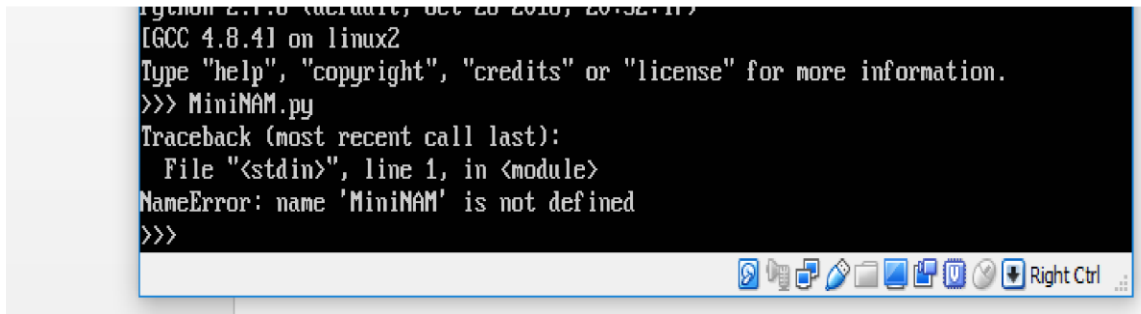


6. Setelah itu ketikkan lagi perintah sebagai berikut

sudo python



7. Yang terakhir, disamping tanda >>> yang juga terdapat pada langkah no. 6 ketikkan **"MiniNAM.py"**



```
Python 2.7.6 (default, Oct 26 2016, 20:32:11)
[GCC 4.8.4] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> MiniNAM.py
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'MiniNAM' is not defined
>>>
```

8. Penginstalan mininam pun sudah selesai .

PutTY dan Xming

PutTY adalah sebuah aplikasi open-source memanfaatkan protokol jaringan seperti SSH dan Telnet. **PutTY** memanfaatkan protokol tersebut untuk mengaktifkan sesi remote pada komputer. ... **PutTY** is an SSH and telnet client, developed originally by Simon Tatham for the Windows platform.

Ini adalah tools yang kebanyakan dipakai oleh para pemilik server untuk berkomunikasi dengan servernya menggunakan command teks guna menjalankan perintah tertentu.

Tujuan utama dari PutTY adalah menjadi aplikasi multi-platform yang mampu mengeksekusi dalam sebuah sistem operasi. Hal ini juga disebut terminal xterm.

Jendela utama dari PutTY memiliki sesi yang berjalan pada komputer remote dan dan dapat mengirim perintah langsung ke komputer remote.

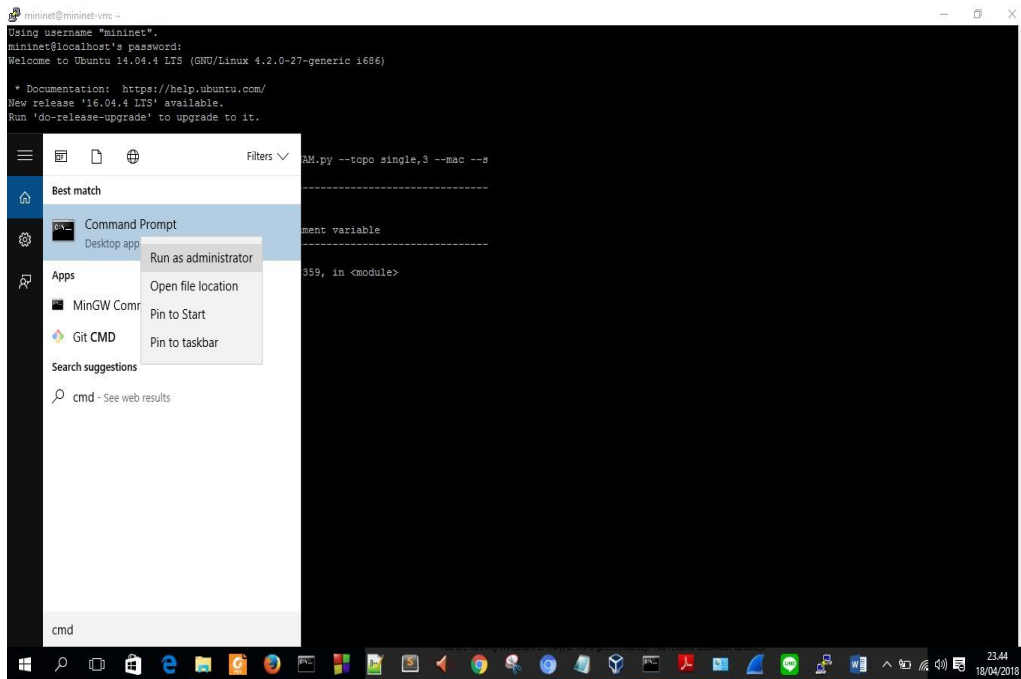
PutTY memberikan beberapa keuntungan yang berbeda, terutama ketika bekerja dari jarak jauh. Hal ini dirasa lebih memudahkan untuk mengkonfigurasi.

XMING adalah software yang mengimplementasikan X Server pada sistem operasi Windows. ... Untuk mengakses SSH, **XMING** memerlukan software lain yang berperan sebagai alat untuk melakukan transfer data pada jalur SSH, seperti Putty misalnya.

Langkah langkah:

1. Download putty.exe (installer melalui internet atau bisa juga share softcopy jika sudah ada teman yang memiliki installer tersebut sebelumnya)

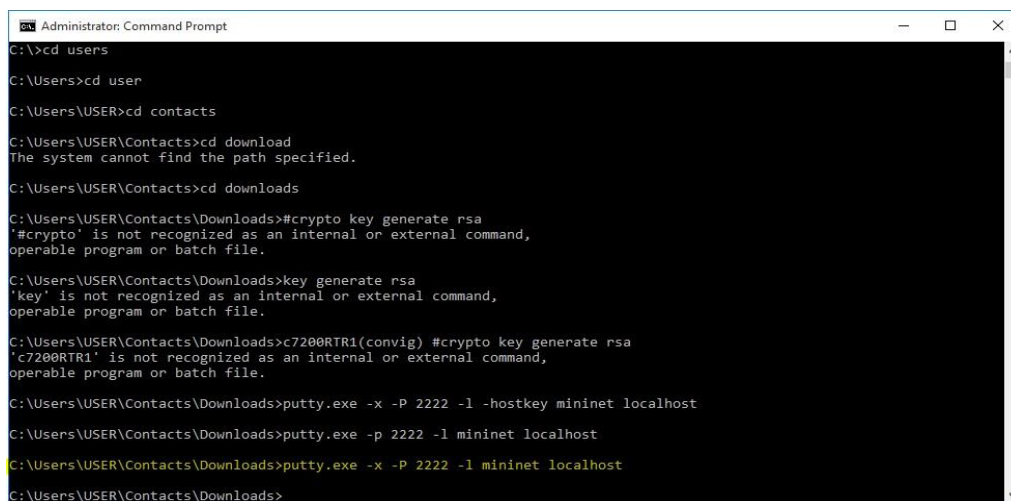
2. Buka cmd (sebaiknya sebelum dibuka ada baiknya di klik kanan – pilih run as administrator) karena banyaknya keluhan yang bermunculan karena putty ini terkadang tidak bisa berjalan disemua pc dengan mudah



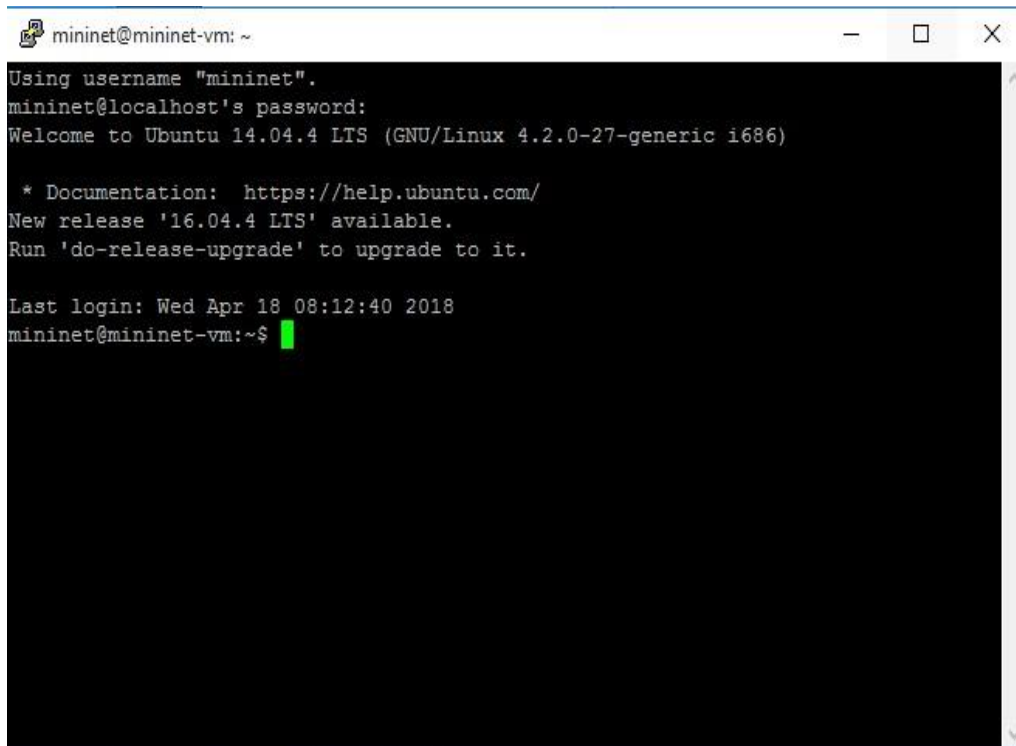
3. Setelah itu ketikkan perintah sebagai berikut

C:\Users\USER\Contacts\Downloads>putty.exe -x -P 2222 -l mininet localhost

Seperti yang teman-teman lihat, saya mengetikkan perintah “putty.exe -X -P 2222 -l mininet localhost” pada folder C:\Users\USER\Contacts\Downloads. Jadi itu tergantung teman-teman dimana menyimpan file installer putty tersebt. Dan akan muncul tampilan seperti dibawah ini (detailnya seperti tanda bercetak kuning)



4. Setelah itu akan muncul laman login seperti dibawah ini



```
mininet@mininet-vm: ~  
Using username "mininet".  
mininet@localhost's password:  
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 4.2.0-27-generic i686)  
  
 * Documentation:  https://help.ubuntu.com/  
New release '16.04.4 LTS' available.  
Run 'do-release-upgrade' to upgrade to it.  
  
Last login: Wed Apr 18 08:12:40 2018  
mininet@mininet-vm:~$
```

Masukkan username : mininet dan Password : mininet seperti tampilan diatas. Jika sudah sesuai dengan tampilan diatas, maka putty anda sudah terpasang dengan sempurna.

Openflow

OpenFlow adalah sebuah protokol yang memungkinkan pengaturan penjaluran dan pengiriman packet ketika melalui sebuah switch (see : demo). ... Hal ini dapat dilakukan dengan membuat algoritma dan forwarding rules-nya di controller server kemudian aturan tersebut didistribusikan ke switch yang ada di jaringan.

LANGKAH – LANGKAH

Manual Flow Entry

1. Mengetikkan perintah sebagai berikut, seperti gambar dibawah ini

Sudo ovs-ofctl dump-flows s1 Kemudian : **Controller ptcp: &** dan
mininet@mininet-vm:~\$ sudo mn --topo single,3 --mac -- switch ovsk --
controller remote

```
Mininet-VM [Berjalan] - Oracle VM VirtualBox
Berkas  Mesin  Tilik  Masukan  Peranti  Bantuan

Mesin Virtual melaporkan bahwa OS guest tidak mendukung integrasi pointer mouse dalam mode video saat ini.

File "/home/mininet/MiniNAM/MiniNAM.py", line 2359, in <module>
  app.stop()
NameError: name 'app' is not defined
mininet@mininet-vm:~$ sudo ovs-ofctl dump-flows s1
ovs-ofctl: s1 is not a bridge or a socket
mininet@mininet-vm:~$ controller ptcp: &
[1] 9892
mininet@mininet-vm:~$ sudo mn --topo single,3 --mac --switch 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>
```

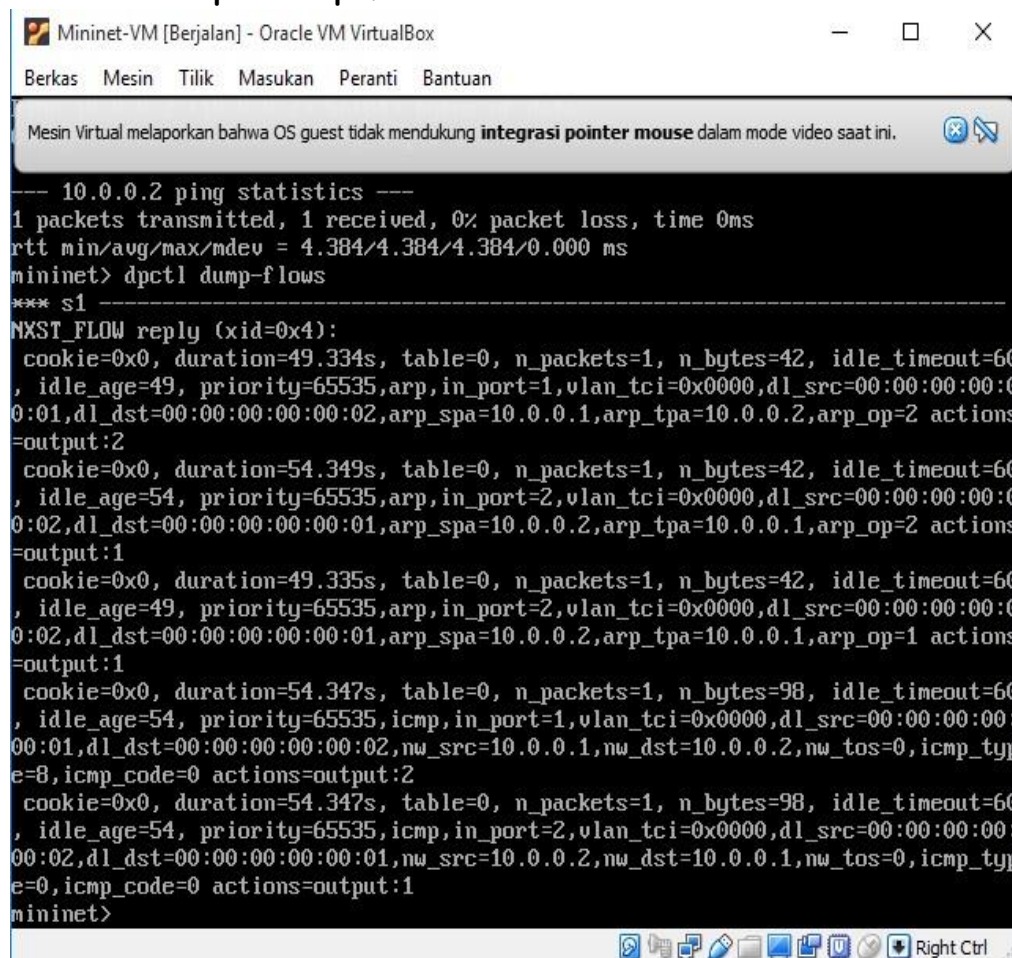
2. Kemudian mengetikkan
mininet> h1
ping -c1 h2

```
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=4.38 ms

--- 10.0.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 4.384/4.384/4.384/0.000 ms
mininet>
```

3. Kemudian ketikkan perintah dibawah ini hingga muncul hasil seperti gambar dibawah

Mininet> Dpctl dump-flows



```
Mininet-VM [Berjalan] - Oracle VM VirtualBox
Berkas Mesin Tilik Masukan Peranti Bantuan

Mesin Virtual melaporkan bahwa OS guest tidak mendukung integrasi pointer mouse dalam mode video saat ini.

--- 10.0.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 4.384/4.384/4.384/0.000 ms
mininet> dpctl dump-flows
*** s1 ***
NXST_FLOW reply (xid=0x4):
  cookie=0x0, duration=49.334s, table=0, n_packets=1, n_bytes=42, idle_timeout=60,
  idle_age=49, priority=65535, arp, in_port=1, vlan_tci=0x0000, dl_src=00:00:00:00:00:01,
  dl_dst=00:00:00:00:00:02, arp_spa=10.0.0.1, arp_tpa=10.0.0.2, arp_op=2 actions
  =output:2
  cookie=0x0, duration=54.349s, table=0, n_packets=1, n_bytes=42, idle_timeout=60,
  idle_age=54, priority=65535, arp, in_port=2, vlan_tci=0x0000, dl_src=00:00:00:00:00:02,
  dl_dst=00:00:00:00:00:01, arp_spa=10.0.0.2, arp_tpa=10.0.0.1, arp_op=2 actions
  =output:1
  cookie=0x0, duration=49.335s, table=0, n_packets=1, n_bytes=42, idle_timeout=60,
  idle_age=49, priority=65535, arp, in_port=2, vlan_tci=0x0000, dl_src=00:00:00:00:00:02,
  dl_dst=00:00:00:00:00:01, arp_spa=10.0.0.2, arp_tpa=10.0.0.1, arp_op=1 actions
  =output:1
  cookie=0x0, duration=54.347s, table=0, n_packets=1, n_bytes=98, idle_timeout=60,
  idle_age=54, priority=65535, icmp, in_port=1, vlan_tci=0x0000, dl_src=00:00:00:00:00:01,
  dl_dst=00:00:00:00:00:02, nw_src=10.0.0.1, nw_dst=10.0.0.2, nw_tos=0, icmp_type=8,
  icmp_code=0 actions=output:2
  cookie=0x0, duration=54.347s, table=0, n_packets=1, n_bytes=98, idle_timeout=60,
  idle_age=54, priority=65535, icmp, in_port=2, vlan_tci=0x0000, dl_src=00:00:00:00:00:02,
  dl_dst=00:00:00:00:00:01, nw_src=10.0.0.2, nw_dst=10.0.0.1, nw_tos=0, icmp_type=0,
  icmp_code=0 actions=output:1
mininet>
```

4. Dilanjutkan dengan mengetikkan perintah sebagai berikut

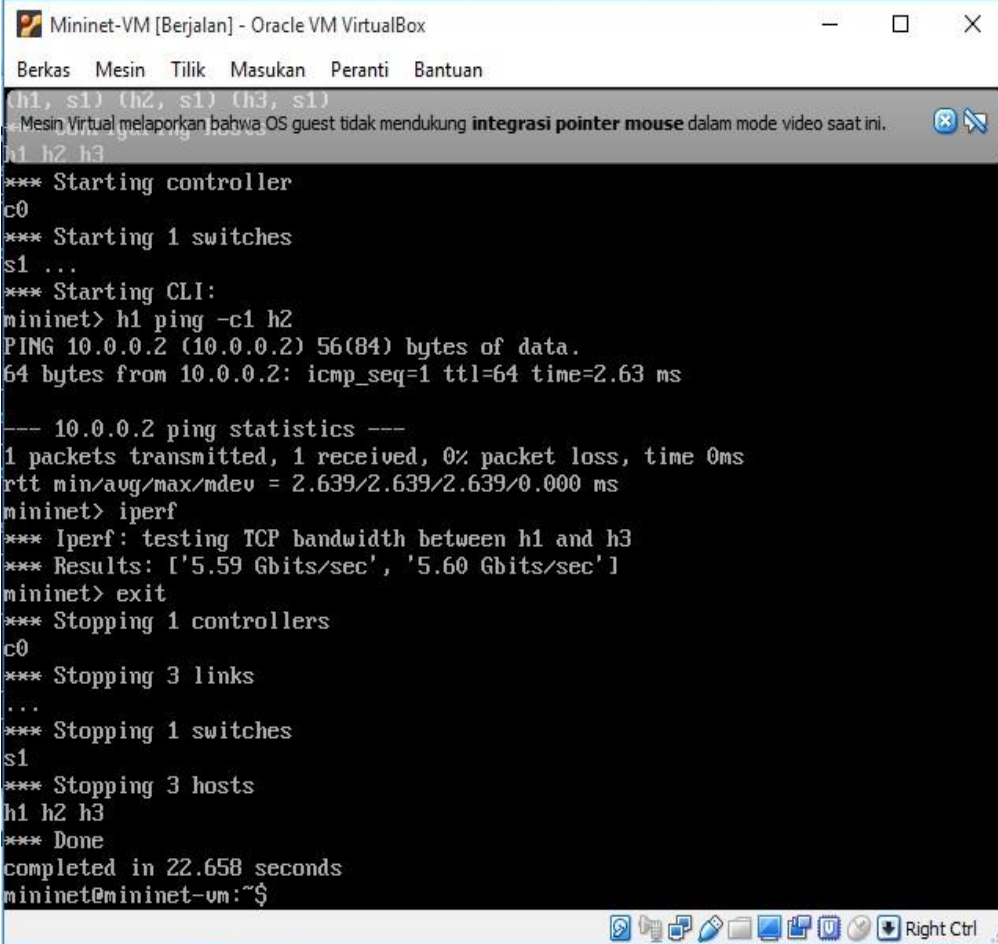
Mininet> h1 ping -c1 h2

Kemudian

Mininet> iperf (seperti pada gambar dibawah ini)

Lalu

Mininet> exit



```
Mininet-VM [Berjalan] - Oracle VM VirtualBox
Berkas Mesin Tilik Masukan Peranti Bantuan
(h1, s1) (h2, s1) (h3, s1)
Mesin Virtual melaporkan bahwa OS guest tidak mendukung integrasi pointer mouse dalam mode video saat ini.
h1 h2 h3
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
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=2.63 ms

--- 10.0.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 2.639/2.639/2.639/0.000 ms
mininet> iperf
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['5.59 Gbits/sec', '5.60 Gbits/sec']
mininet> exit
*** Stopping 1 controllers
c0
*** Stopping 3 links
...
*** Stopping 1 switches
s1
*** Stopping 3 hosts
h1 h2 h3
*** Done
completed in 22.658 seconds
mininet@mininet-vm:~$
```

5. Kemudian melakukan lagi perintah ke 3 pada langkah no 1, kemudian dilanjutkan dengan mengetikkan perintah seperti dibawah ini

Mininet> iperf

Dan yang terakhir

Mininet> sh ovs-ofctl dump-flows s1 seperti gambar dibawah ini

```
mininet> iperf
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['334 Mbits/sec', '334 Mbits/sec']
mininet> sh ovs-ofctl dump-flows s1
ovs-ofctl: s1 is not a bridge or a socket
mininet>
```



Dan openflow pun sudah siap digunakan.

POX Controller

POX controller adalah salah satu controller SDN yang mendukung protokol OpenFlow. POX adalah controller yang berbasis bahasa Python. Tahapannya sebagai berikut.

1. Melakukan POX Controller files pada Mininet

```
mininet@mininet-vm: ~/pox/pox/forwarding
Using username "mininet".
mininet@localhost's password:
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 4.2.0-27-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
Last login: Tue Apr 17 23:17:03 2018 from 10.0.2.2
mininet@mininet-vm:~$ cd pox
mininet@mininet-vm:~/pox$ cd pox
mininet@mininet-vm:~/pox/pox$ cd misc
mininet@mininet-vm:~/pox/pox/misc$ ll
total 76
drwxrwxr-x  3 mininet mininet  4096 Mar 21  2017 ./
drwxrwxr-x 15 mininet mininet  4096 Mar 21  2017 ../
-rw-rw-r--  1 mininet mininet  1240 Mar 21  2017 cbench.py
-rw-rw-r--  1 mininet mininet  1079 Mar 21  2017 full_payload.py
-rw-rw-r--  1 mininet mininet  5214 Mar 21  2017 gephi_topo.py
-rw-rw-r--  1 mininet mininet   689 Mar 21  2017 __init__.py
-rw-rw-r--  1 mininet mininet 10251 Mar 21  2017 ip_loadBalancer.py
-rw-rw-r--  1 mininet mininet   3794 Mar 21  2017 mac_blocker.py
-rw-rw-r--  1 mininet mininet 14375 Mar 21  2017 nat.py
-rw-rw-r--  1 mininet mininet   4582 Mar 21  2017 of_tutorial.py
-rw-rw-r--  1 mininet mininet   2096 Mar 21  2017 pidfile.py
drwxrwxr-x  2 mininet mininet  4096 Mar 21  2017 telnetd/
mininet@mininet-vm:~/pox/pox/misc$ cd
mininet@mininet-vm:~$ cd pox/pox/forwarding
mininet@mininet-vm:~/pox/pox/forwarding$ ll
total 96
drwxrwxr-x  2 mininet mininet  4096 Mar 21  2017 ./
drwxrwxr-x 15 mininet mininet  4096 Mar 21  2017 ../
-rw-rw-r--  1 mininet mininet  1092 Mar 21  2017 hub.py
-rw-rw-r--  1 mininet mininet   651 Mar 21  2017 __init__.py
-rw-rw-r--  1 mininet mininet   4426 Mar 21  2017 l2_flowvisor.py
-rw-rw-r--  1 mininet mininet   6692 Mar 21  2017 l2_learning.py
-rw-rw-r--  1 mininet mininet 15558 Mar 21  2017 l2_multi.py
-rw-rw-r--  1 mininet mininet   4324 Mar 21  2017 l2_nx.py
-rw-rw-r--  1 mininet mininet   2105 Mar 21  2017 l2_nx_self_learning.py
-rw-rw-r--  1 mininet mininet   2882 Mar 21  2017 l2_pairs.py
-rw-rw-r--  1 mininet mininet 12330 Mar 21  2017 l3_learning.py
-rw-rw-r--  1 mininet mininet 14102 Mar 21  2017 topo_proactive.py
mininet@mininet-vm:~/pox/pox/forwarding$
```

2. Memilih of_tutorial.py pada pox maupun pox dan misc

```
mininet@mininet-vm: ~/pox/pox/misc
# Copyright 2012 James McCauley
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at:
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the license for the specific language governing permissions and
# limitations under the License.
"""
This component is for use with the OpenFlow tutorial.
It acts as a simple hub, but can be modified to act like an L2
learning switch.
It's roughly similar to the one Brandon Heller did for NOX.
"""
from pox.core import core
import pox.openflow.libopenflow_01 as of
log = core.getLogger()

class Tutorial(object):
    """
    A Tutorial object is created for each switch that connects.
    A Connection object for that switch is passed to the __init__ function.
    """
    def __init__(self, connection):
        # Keep track of the connection to the switch so that we can
        # send it messages!
        self.connection = connection

        # This binds our PacketIn event listener
        connection.addListeners(self)

        # Use this table to keep track of which ethernet address is on
        # which switch port (keys are MACs, values are ports).
        self.mac_to_port = {}

    def resend_packet(self, packet_in, out_port):
        """
        Instructs the switch to resend a packet that it had sent to us.
        "packet_in" is the ofp_packet_in object the switch had sent to the
        """
```


- Memulai pox controller dengan './pox.py log.level --DEBUG misc.of_tutorial pada directory pox'. Kemudian mengetikkan lagi perintah **sudo mn --topo single,3 --mac --witch ovsk -controller remote**.
- Setelah itu, ketikkan perintah sebagai berikut **'tcpdump -XX -n -i h2-eth0 > h2.txt'** pada terminal node 2, dan **'tcpdump -XX -n -i h3-eth0 > h3.txt'** pada terminal node 3. Lalu lakukan ping pada terminal node 1 dengan perintah **'ping -c1 10.0.0.2'**

```
mininet@mininet-vm:~$
mininet@mininet-vm:~$ sudo mn -c
*** Removing excess controllers/ofprotocols/ofdatapaths/pings/noxes
killall -9 controller ofprotocol ofdatapath ping nox_core if-nox_core ovs-openflow ovs-controller udptest mininet ixe 25 /dev/null
killall -9 controller ofprotocol ofdatapath ping nox_core if-nox_core ovs-openflow ovs-controller udptest mininet ixe 25 /dev/null
rm -f /tmp/ovsk /tmp/ovsk* /tmp/*.*.out /tmp/*.*.log
*** Removing old X11 tunnels
*** Removing excess kernel datapaths
ps ax | egrep -o 'dp[0-9]*' | sed 's/dp/nli/'
*** Removing OVS datapaths
ovs-vsctl --timeout=1 list-br
ovs-vsctl --timeout=1 list-br
*** Removing all links of the pattern fop-eth*
ip link show | egrep -o '([0-9]{4})-eth[0-9]{2}' | sed 's/([0-9]{4})-eth[0-9]{2}/[0-9]{4}-eth[0-9]{2}/'
*** Killing stale mininet node processes
ip link show
*** Shutting down stale tunnels
killall -9 -f tunnelstunnel
killall -9 -f ssh/min
rm -f /tmp/ovsk/*
*** Cleanup complete
mininet@mininet-vm:~$ sudo mn --topo single,3 --mac --switch ovsk --controller remote
*** Creating network
*** Adding controller
Unable to contact the remote controller at 127.0.0.1:6633
Unable to contact the remote controller at 127.0.0.1:6633
Setting remote controller to 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> term h1 h2 h3
mininet>
```

```
Node h1
root@mininet-vm:~$ ping -c1 10.0.0.2
PING 10.0.0.2 (32.0.0.2) 56(84) bytes of data:
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.45 ms
--- 10.0.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.454/0.454/0.454/0.000 ms
root@mininet-vm:~$

Node h2
root@mininet-vm:~$ tcpdump -XX -n -i h2-eth0 > h2.txt
tcpdump: command not found
root@mininet-vm:~$ tcpdump -XX -n -i h2-eth0 > h2.txt
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on h2-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
0 packets captured
0 packets received by filter
0 packets dropped by kernel
root@mininet-vm:~$

Node h3
root@mininet-vm:~$ tcpdump -XX -n -i h3-eth0 > h3.txt
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on h3-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
0 packets captured
0 packets received by filter
0 packets dropped by kernel
root@mininet-vm:~$
```

```
mininet@mininet-vm:~$ git clone http://github.com/noxrepo/pox
Fatal: destination path 'pox' already exists and is not an empty directory.
mininet@mininet-vm:~$ git checkout beta
Fatal: Not a git repository (or any of the parent directories): .git
mininet@mininet-vm:~$ cd pox
mininet@mininet-vm:~/pox$ git checkout beta
Note: branch is up-to-date with 'origin/beta'.
mininet@mininet-vm:~/pox$ ./pox.py log.level --DEBUG misc.of_tutorial
Pox 0.1.0 (beta) / Copyright 2011-2013 James McKeown, et al.
DEBUG:core:Running on Python (2.7.6/beta) 26 Jan 2016 20:30:19
DEBUG:core:Platform is Linux-4.2.0-27-generic-x86_64-with-Ubuntu-14.04-trusty
DEBUG:core:Pox 0.1.0 (beta) is up.
DEBUG:openflow.of_01:Listening on 0.0.0.0:6633
INFO:openflow.of_01:[None 1] closed
INFO:openflow.of_01:[00-00-00-00-00-01 2] connected
DEBUG:misc.of_tutorial:Controlling [00-00-00-00-00-01 2]
[]
```

mininet@mininet-vm: ~

mininet@mininet-vm:~\$ cat h2.txt

```
08:39:34.828743 ARP, Request who-has 10.0.0.2 tell 10.0.0.1, length 28
  0x0000: ffff ffff ffff 0000 0000 0001 0806 0001 .....
  0x0010: 0800 0604 0001 0000 0000 0001 0a00 0001 .....
  0x0020: 0000 0000 0000 0a00 0002 .....
08:39:34.828766 ARP, Reply 10.0.0.2 is-at 00:00:00:00:00:02, length 28
  0x0000: 0000 0000 0001 0000 0000 0002 0806 0001 .....
  0x0010: 0800 0604 0002 0000 0000 0002 0a00 0002 .....
  0x0020: 0000 0000 0001 0a00 0001 .....
08:39:34.831160 IP 10.0.0.1 > 10.0.0.2: ICMP echo request, id 5099, seq 1, length 64
  0x0000: 0000 0000 0002 0000 0000 0001 0800 4500 .....E.
  0x0010: 0054 13fb 4000 4001 1fd7 0a00 0001 0a00 .T..@.....
  0x0020: 0002 0800 69f2 13eb 0001 b666 d75a 0000 ....i.....f.Z..
  0x0030: 0000 218d 0c00 0000 0000 1011 1213 1415 ..!.....
  0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#$$
  0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
  0x0060: 3637 67
08:39:34.831393 IP 10.0.0.2 > 10.0.0.1: ICMP echo reply, id 5099, seq 1, length 64
  0x0000: 0000 0000 0001 0000 0000 0002 0800 4500 .....E.
  0x0010: 0054 46d0 0000 4001 1fd7 0a00 0002 0a00 .TF...@.....
  0x0020: 0001 0000 71f2 13eb 0001 b666 d75a 0000 ....q.....f.Z..
  0x0030: 0000 218d 0c00 0000 0000 1011 1213 1415 ..!.....
  0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#$$
  0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
  0x0060: 3637 67
08:39:39.835283 ARP, Request who-has 10.0.0.1 tell 10.0.0.2, length 28
  0x0000: 0000 0000 0001 0000 0000 0002 0806 0001 .....
  0x0010: 0800 0604 0001 0000 0000 0002 0a00 0002 .....
  0x0020: 0000 0000 0000 0a00 0001 .....
08:39:39.883178 ARP, Reply 10.0.0.1 is-at 00:00:00:00:00:01, length 28
  0x0000: 0000 0000 0002 0000 0000 0001 0806 0001 .....
  0x0010: 0800 0604 0002 0000 0000 0001 0a00 0001 .....
  0x0020: 0000 0000 0002 0a00 0002 .....
```

mininet@mininet-vm:~\$ █

mininet@mininet-vm:~\$ cat h3.txt

```
08:39:34.828740 ARP, Request who-has 10.0.0.2 tell 10.0.0.1, length 28
  0x0000: ffff ffff ffff 0000 0000 0001 0806 0001 .....
  0x0010: 0800 0604 0001 0000 0000 0001 0a00 0001 .....
  0x0020: 0000 0000 0000 0a00 0002 .....
08:39:34.829769 ARP, Reply 10.0.0.2 is-at 00:00:00:00:00:02, length 28
  0x0000: 0000 0000 0001 0000 0000 0002 0806 0001 .....
  0x0010: 0800 0604 0002 0000 0000 0002 0a00 0002 .....
  0x0020: 0000 0000 0001 0a00 0001 .....
08:39:34.831159 IP 10.0.0.1 > 10.0.0.2: ICMP echo request, id 5099, seq 1, length 64
  0x0000: 0000 0000 0002 0000 0000 0001 0800 4500 .....E.
  0x0010: 0054 13fb 4000 4001 1fd7 0a00 0001 0a00 .T..@.....
  0x0020: 0002 0800 69f2 13eb 0001 b666 d75a 0000 ....i.....f.Z..
  0x0030: 0000 218d 0c00 0000 0000 1011 1213 1415 ..!.....
  0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#$$
  0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
  0x0060: 3637 67
08:39:34.832014 IP 10.0.0.2 > 10.0.0.1: ICMP echo reply, id 5099, seq 1, length 64
  0x0000: 0000 0000 0001 0000 0000 0002 0800 4500 .....E.
  0x0010: 0054 46d0 0000 4001 1fd7 0a00 0002 0a00 .TF...@.....
  0x0020: 0001 0000 71f2 13eb 0001 b666 d75a 0000 ....q.....f.Z..
  0x0030: 0000 218d 0c00 0000 0000 1011 1213 1415 ..!.....
  0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#$$
  0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
  0x0060: 3637 67
08:39:39.877184 ARP, Request who-has 10.0.0.1 tell 10.0.0.2, length 28
  0x0000: 0000 0000 0001 0000 0000 0002 0806 0001 .....
  0x0010: 0800 0604 0001 0000 0000 0002 0a00 0002 .....
  0x0020: 0000 0000 0000 0a00 0001 .....
08:39:39.883173 ARP, Reply 10.0.0.1 is-at 00:00:00:00:00:01, length 28
  0x0000: 0000 0000 0002 0000 0000 0001 0806 0001 .....
  0x0010: 0800 0604 0002 0000 0000 0001 0a00 0001 .....
  0x0020: 0000 0000 0002 0a00 0002 .....
```

mininet@mininet-vm:~\$ █

- Mengetikkan kembali perintah untuk merekam aktivitas node lagi pada node 2 dan 3. Untuk lebih jelasnya, kita bisa menyaksikan gambar dibawah ini

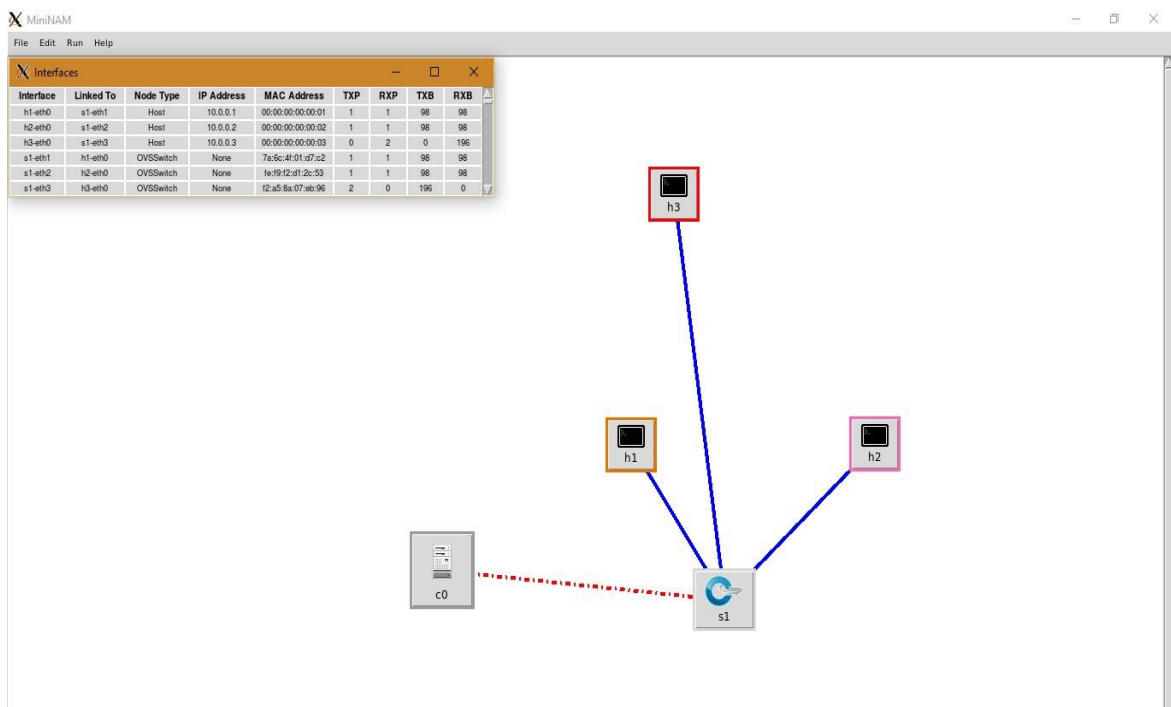
```

Node: h2
mininet@mininet-vm:~$ cat h2.txt
08:49:13.332689 ARP, Request who-has 10.0.0.5 tell 10.0.0.1, length 28
0x0000:  ffff ffff ffff 0000 0000 0001 0806 0001  .....
0x0010:  0800 0604 0001 0000 0000 0001 0a00 0001  .....
0x0020:  0000 0000 0000 0a00 0005  .....
08:49:14.331482 ARP, Request who-has 10.0.0.5 tell 10.0.0.1, length 28
0x0000:  ffff ffff ffff 0000 0000 0001 0806 0001  .....
0x0010:  0800 0604 0001 0000 0000 0001 0a00 0001  .....
0x0020:  0000 0000 0000 0a00 0005  .....
08:49:15.332863 ARP, Request who-has 10.0.0.5 tell 10.0.0.1, length 28
0x0000:  ffff ffff ffff 0000 0000 0001 0806 0001  .....
0x0010:  0800 0604 0001 0000 0000 0001 0a00 0001  .....
0x0020:  0000 0000 0000 0a00 0005  .....
mininet@mininet-vm:~$ █

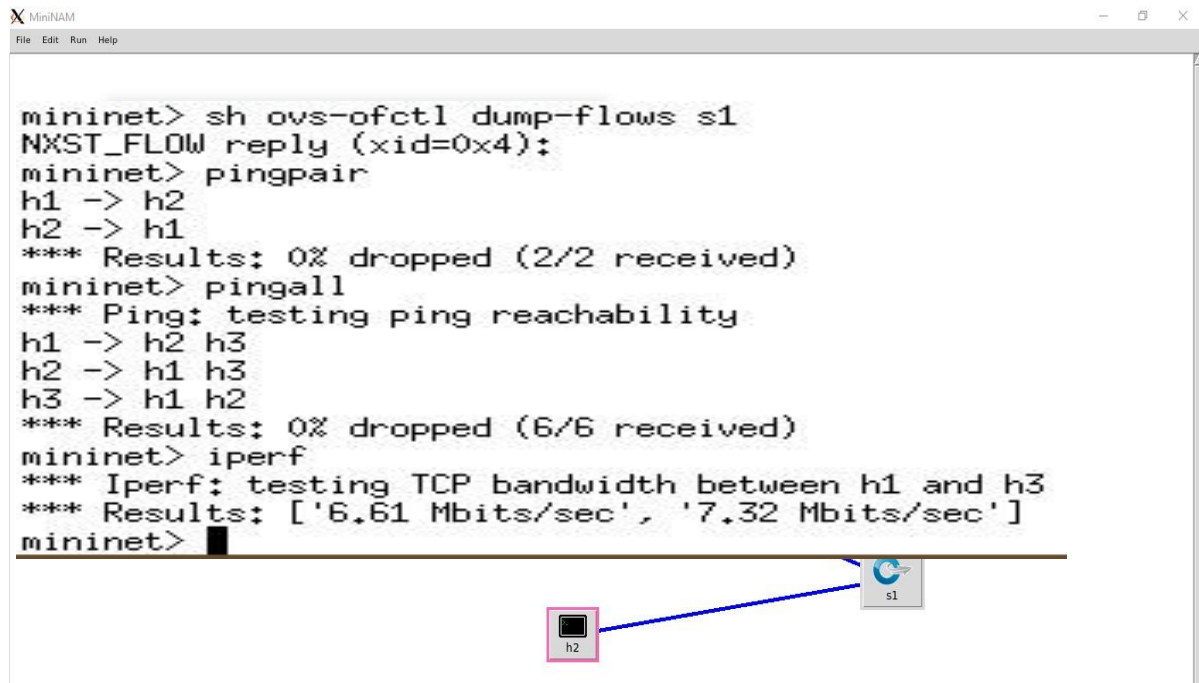
[listening on h3-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes] 1 packets transmitted, 1 received, 0% packet loss, time 0ms
all 0 packets captured
mininet@mininet-vm:~$ cat h3.txt
08:49:13.332686 ARP, Request who-has 10.0.0.5 tell 10.0.0.1, length 28
0x0000:  ffff ffff ffff 0000 0000 0001 0806 0001  .....
0x0010:  0800 0604 0001 0000 0000 0001 0a00 0001  .....
0x0020:  0000 0000 0000 0a00 0005  .....
08:49:14.331478 ARP, Request who-has 10.0.0.5 tell 10.0.0.1, length 28
0x0000:  ffff ffff ffff 0000 0000 0001 0806 0001  .....
0x0010:  0800 0604 0001 0000 0000 0001 0a00 0001  .....
0x0020:  0000 0000 0000 0a00 0005  .....
08:49:15.332860 ARP, Request who-has 10.0.0.5 tell 10.0.0.1, length 28
0x0000:  ffff ffff ffff 0000 0000 0001 0806 0001  .....
0x0010:  0800 0604 0001 0000 0000 0001 0a00 0001  .....
0x0020:  0000 0000 0000 0a00 0005  .....
mininet@mininet-vm:~$ █

```

- Tampilan jaringan yg kita buat tadi pada MiniNAM ketika ping dari 10.0.0.1 ke 10.0.0.2. terlihat ada komunikasi antara host 1 dan 2.



7. Tampilan jaringan yg kita buat tadi pada MiniNAM ketika ping dari 10.0.0.1 ke 10.0.0.5. terlihat tidak ada komunikasi antar host.



The screenshot shows the MiniNAM application window. The console displays the following commands and output:

```
mininet> sh ovs-ofctl dump-flows s1
NXST_FLOW reply (xid=0x4):
mininet> pingpair
h1 -> h2
h2 -> h1
*** Results: 0% dropped (2/2 received)
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3
h2 -> h1 h3
h3 -> h1 h2
*** Results: 0% dropped (6/6 received)
mininet> iperf
*** Iperf: testing TCP bandwidth between h1 and h3
*** Results: ['6.61 Mbits/sec', '7.32 Mbits/sec']
mininet>
```

Below the console, a network diagram is visible, showing a host icon labeled 'h2' connected to a switch icon labeled 's1' by a blue line.

disini kita akan melakukan perbandingan antara kecepatan menggunakan hub dan switch. berikan perintah iperf pada console.

8. sehingga dapat kita simpulkan bahwa switch dapat bekerja lebih cepat daripada hub

