MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

DEPARTMENT OF ICT

ASSIGNMENT NO : 01

Course Code : ICT-3108

Course Title : Network Planning and designing Lab.
AssignmentName :Mininet Walkthrough

Submitted by
Md. Md Sohanur Islam
ID : IT-18011
Session :2017-18
Year : 3rd Semester : 1st

Submitted to
Nazrul Islam
AssistantProfessor,
Department of ICT,MBSTU
Santosh,Tangail-1902

Date of Submission :26-10-2020

Objectives : This exeperiment is about Install and understand how to mininet emulator works . Beside , to leran how to mininet networks creates a network of virtual hosts, switches, controllers, and links and hosts run standard Linux network software, and its switches support OpenFlow for highly flexible custom routing and Software-Defined Networking

Explanation :

Mininet: Mininet is an emulator for deploying large networks on the limited resources of a simple single Computer or Virtual Machine. Mininet has been created for enabling research in Software Defined Networking (SDN) and OpenFlow. Mininet emulator allows running unmodified code interactively on virtual hardware on a simple PC. It provides convenience and realism at very low cost. The alternative to Mininet is hardware test beds which are fast, accurate but very expensive and shared. The other option is to use simulator which is very cheap but sometimes slow and require code modification. Mininet offers ease of use, performance accuracy and scalability.

Everyday Mininet Usage:

(1) **Display startup options: 'sudo mn -h'** command to display a help message describing Mininet's startup options

```
      --preserve-env=list       preserve specific environment variables
  -e, --edit                    edit files instead of running a command
  -g, --group=group             run command as the specified group name or ID
  -H, --set-home                set HOME variable to target user's home dir
  -h, --help                    display help message and exit
  -h, --host=host               run command on host (if supported by plugin)
  -i, --login                   run login shell as the target user; a command
                                may also be specified
  -K, --remove-timestamp        remove timestamp file completely
  -k, --reset-timestamp         invalidate timestamp file
  -l, --list                    list user's privileges or check a specific
                                command; use twice for longer format
  -n, --non-interactive         non-interactive mode, no prompts are used
  -P, --preserve-groups         preserve group vector instead of setting to
                                target's
  -p, --prompt=prompt           use the specified password prompt
  -r, --role=role               create SELinux security context with specified
                                role
  -S, --stdin                   read password from standard input
  -s, --shell                   run shell as the target user; a command may
                                also be specified
  -t, --type=type               create SELinux security context with specified
                                type
  -T, --command-timeout=timeout terminate command after the specified time
                                limit
  -U, --other-user=user         in list mode, display privileges for user
  -u, --user=user               run command (or edit file) as specified user
                                name or ID
  -V, --version                 display version information and exit
  -v, --validate                update user's timestamp without running a
                                command
  --                            stop processing command line arguments
sohan18011@sohan18011-VirtualBox:~$
```

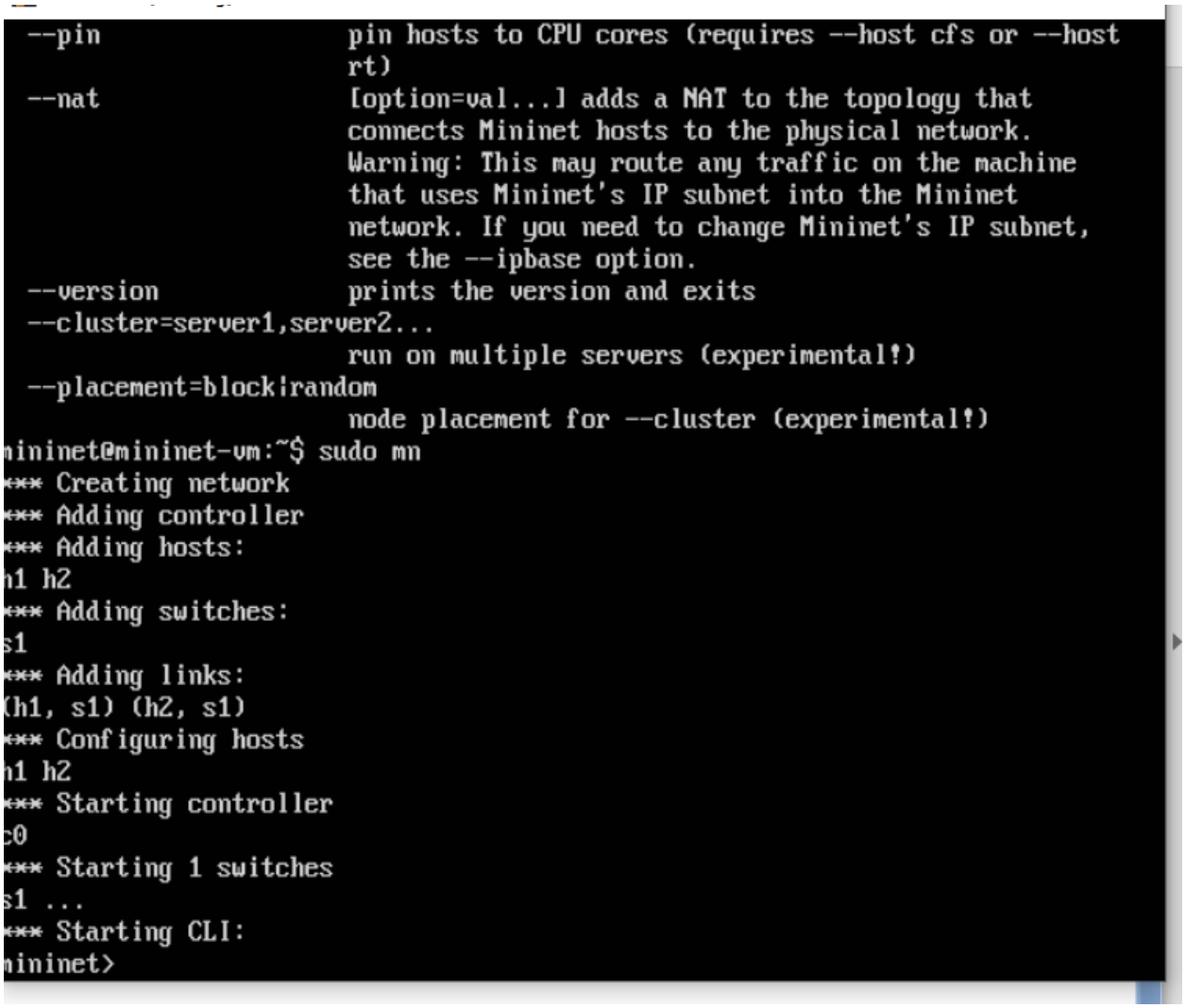
(2)Start Wireshark : To view control traffic using the OpenFlow Wireshark dissector, first open wireshark in the background

```
sohan18011@sohan18011-VirtualBox:~$ sudo wireshark &
[1] 4675
```

1f the system you are using does not have Wireshark and the OpenFlow plugin installed, you may be able to install both of them using Mininet's install.sh script as follows:

Interact with Hosts and Switches:

Start a minimal topology and enter the CLI



1.**Display Mininet CLI commands:**

```
*** Starting CLI:
mininet> help

Documented commands (type help <topic>):
========================================
EOF    gterm  iperfudp  nodes        pingpair      py      switch
dpctl  help   link      noecho       pingpairfull  quit    time
dump   intfs  links     pingall      ports         sh      x
exit   iperf  net       pingallfull  px            source  xterm

You may also send a command to a node using:
  <node> command {args}
For example:
  mininet> h1 ifconfig

The interpreter automatically substitutes IP addresses
for node names when a node is the first arg, so commands
like
  mininet> h2 ping h3
should work.

Some character-oriented interactive commands require
noecho:
  mininet> noecho h2 vi foo.py
However, starting up an xterm/gterm is generally better:
  mininet> xterm h2

mininet>
```

2.Display nodes:

```
mininet> nodes
available nodes are:
c0 h1 h2 s1
mininet>
```

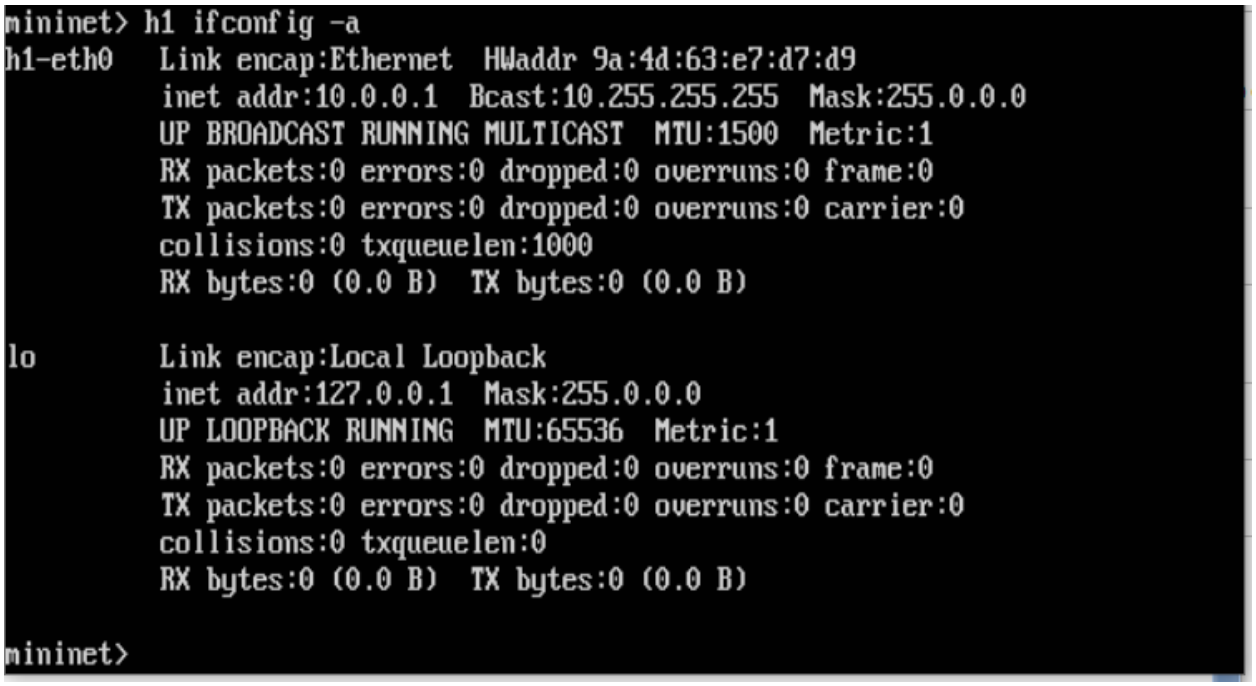
3.Display links:

```
mininet> nodes
available nodes are:
c0 h1 h2 s1
mininet> links
h1-eth0<->s1-eth1 (OK OK)
h2-eth0<->s1-eth2 (OK OK)
mininet> _
```

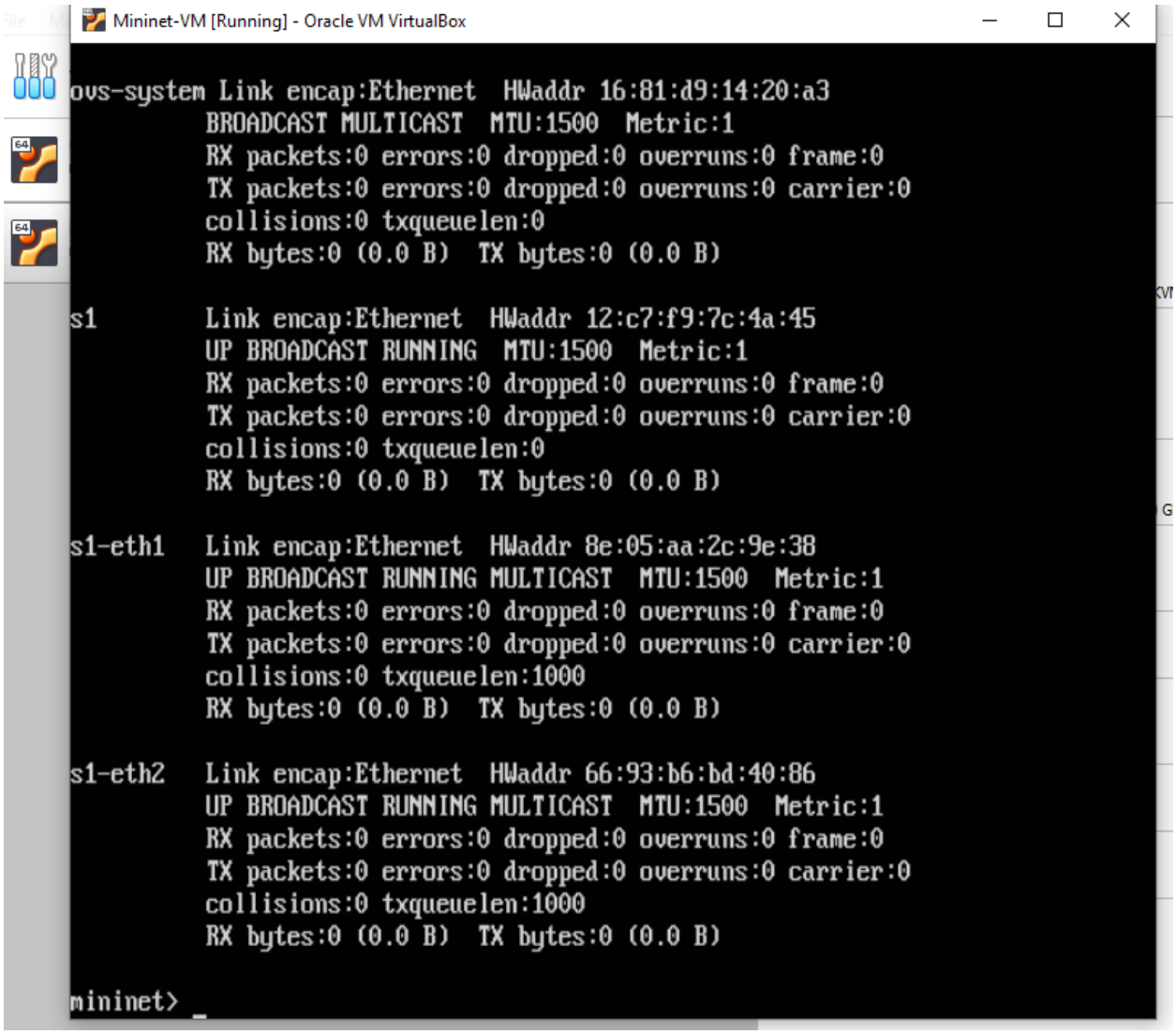
4.Dump information about all nodes:

```
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=1860>
<Host h2: h2-eth0:10.0.0.2 pid=1862>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None pid=1867>
<Controller c0: 127.0.0.1:6653 pid=1853>
mininet>
```

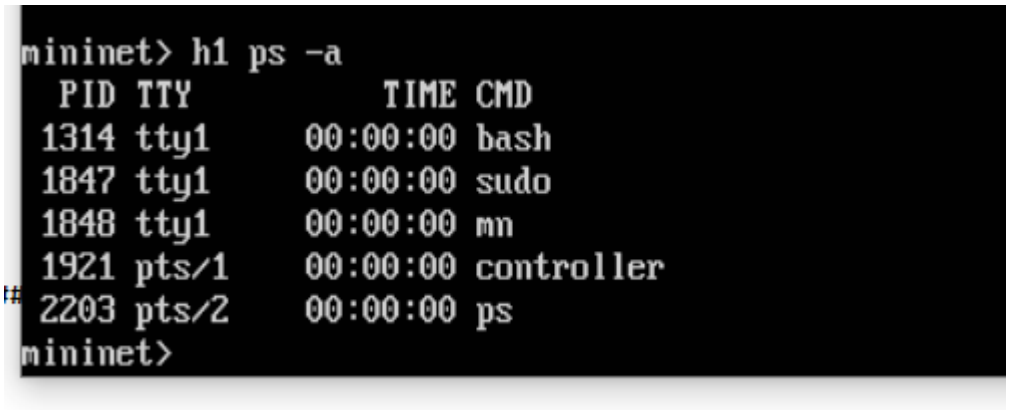
5.Run a command on a host process:



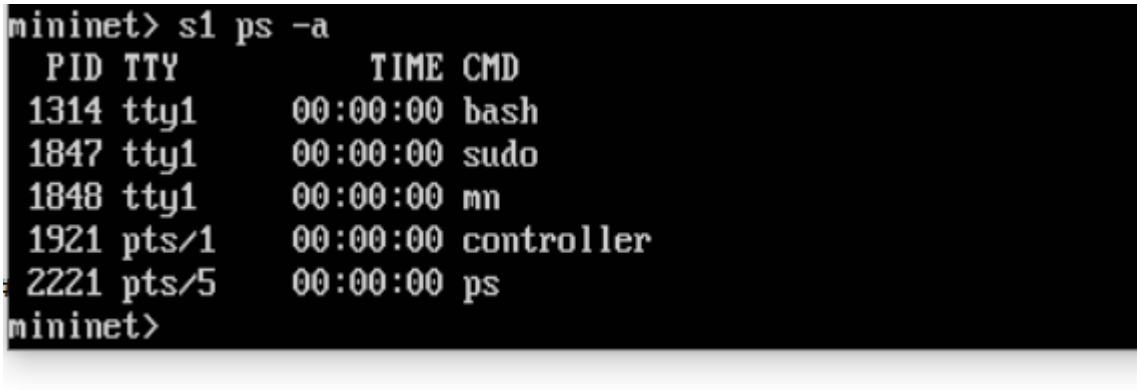
6.Running a command on the “switch” is the same as running it from a regular terminal:



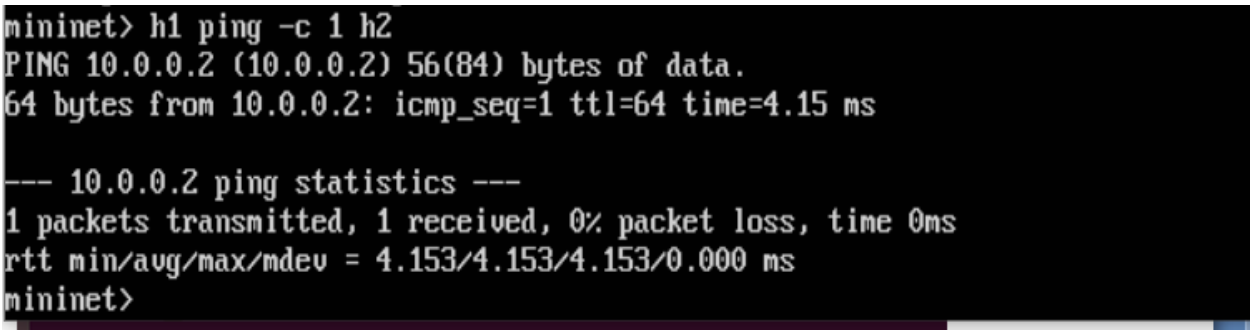
7. print the process list from a host process:



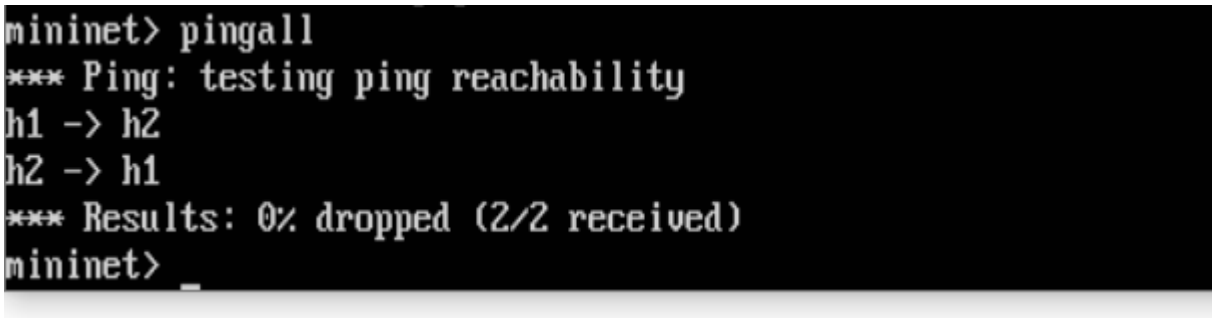
8. Host processseen by the root network namespace:



9.Now, verify that you can ping from host 0 to host 1



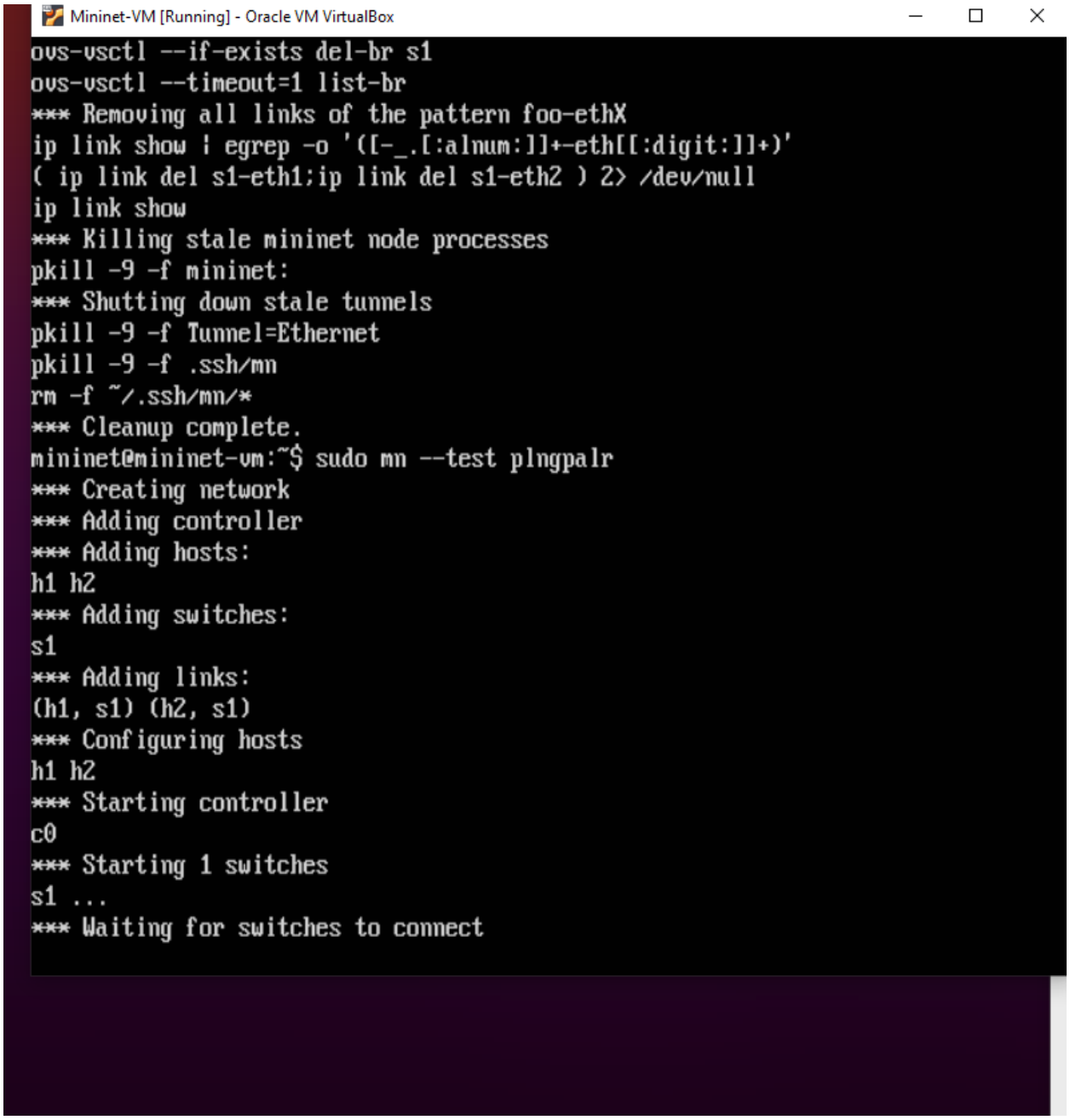
10.An easier way to run this test is to use the Mininet CLI built-in pingall command, which does an all-pairs ping m

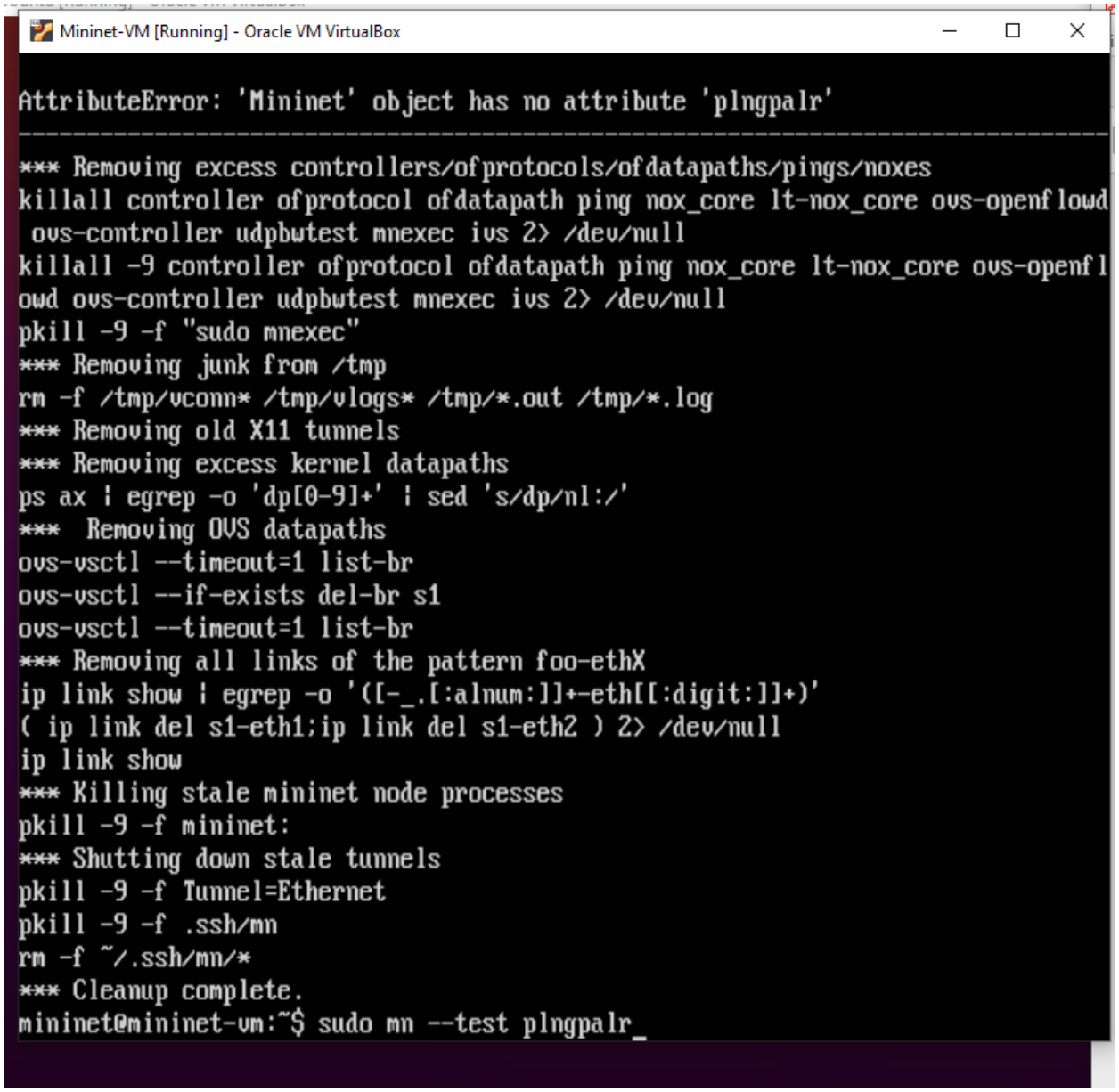


11.Try starting a simple HTTP server on h1, making a request from h2, then shutting down the web server:

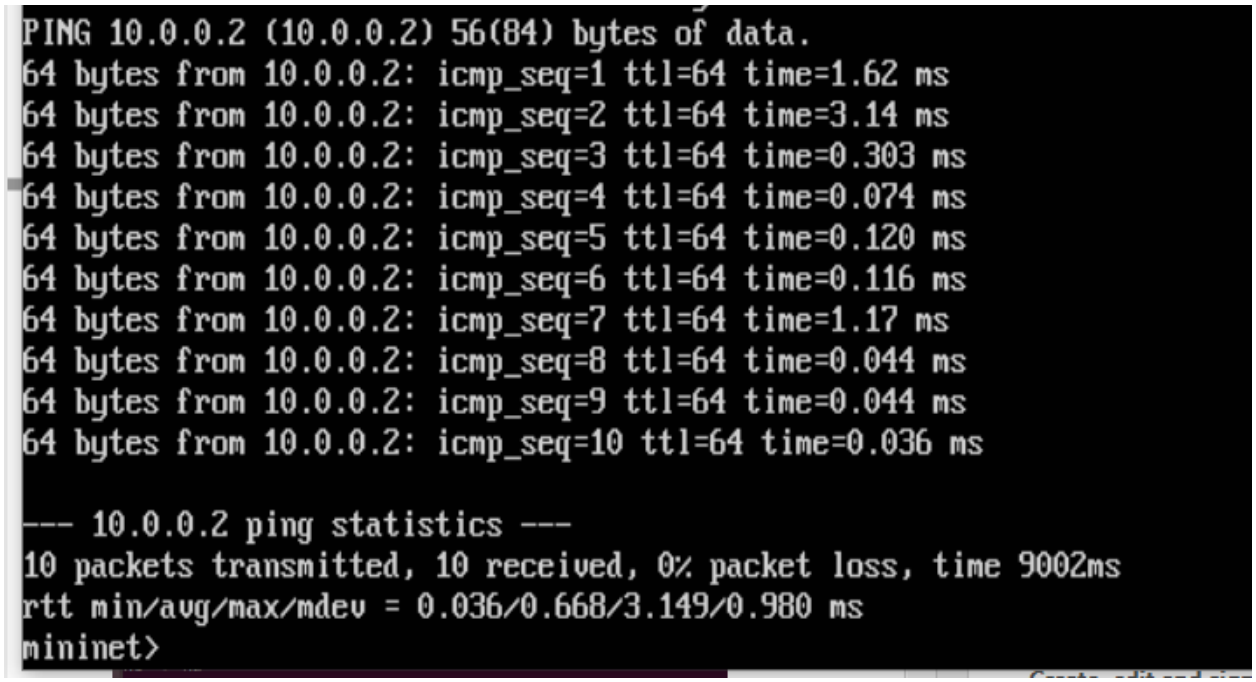
Advanced Startup Options:

A. **Run a regression test:**

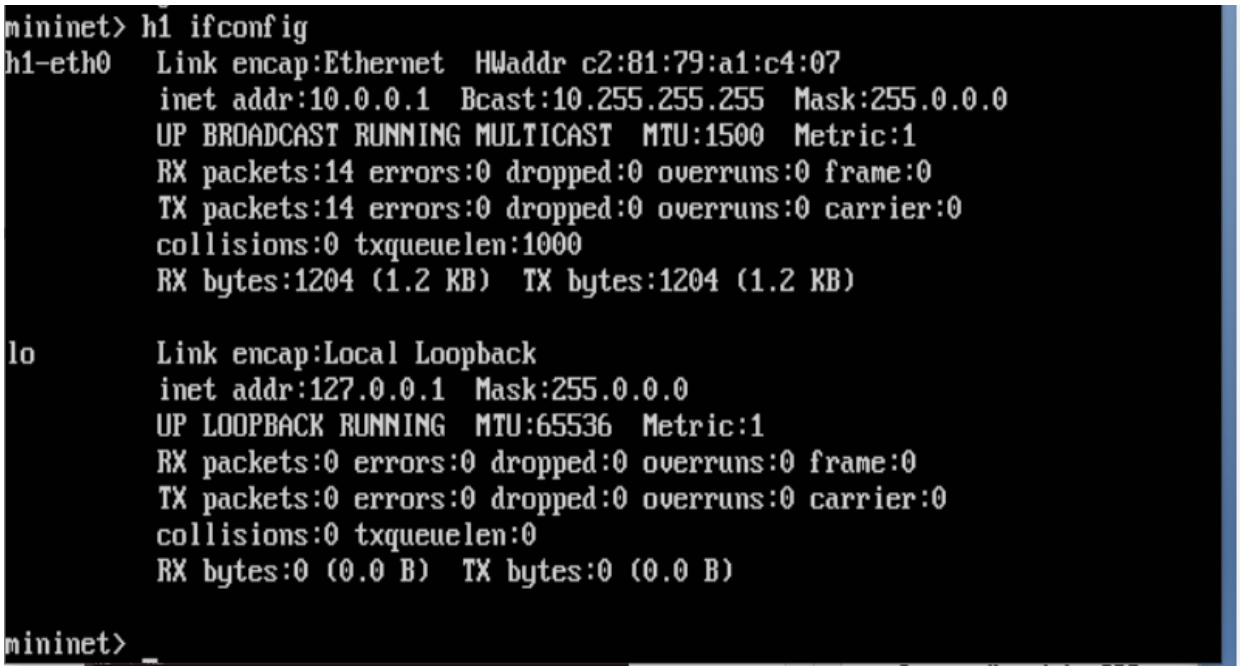

Mininet-VM [Running] - Oracle VM VirtualBox
ovs-vsctl --if-exists del-br s1
ovs-vsctl --timeout=1 list-br
*** Removing all links of the pattern foo-ethX
ip link show | egrep -o '([-_.[:alnum:]]+-eth[[:digit:]]+)'
(ip link del s1-eth1;ip link del s1-eth2) 2> /dev/null
ip link show
*** Killing stale mininet node processes
pkill -9 -f mininet:
*** Shutting down stale tunnels
pkill -9 -f Tunnel=Ethernet
pkill -9 -f .ssh/mn
rm -f ~/.ssh/mn/*
*** Cleanup complete.
mininet@mininet-vm:~$ sudo mn --test pingpair
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect



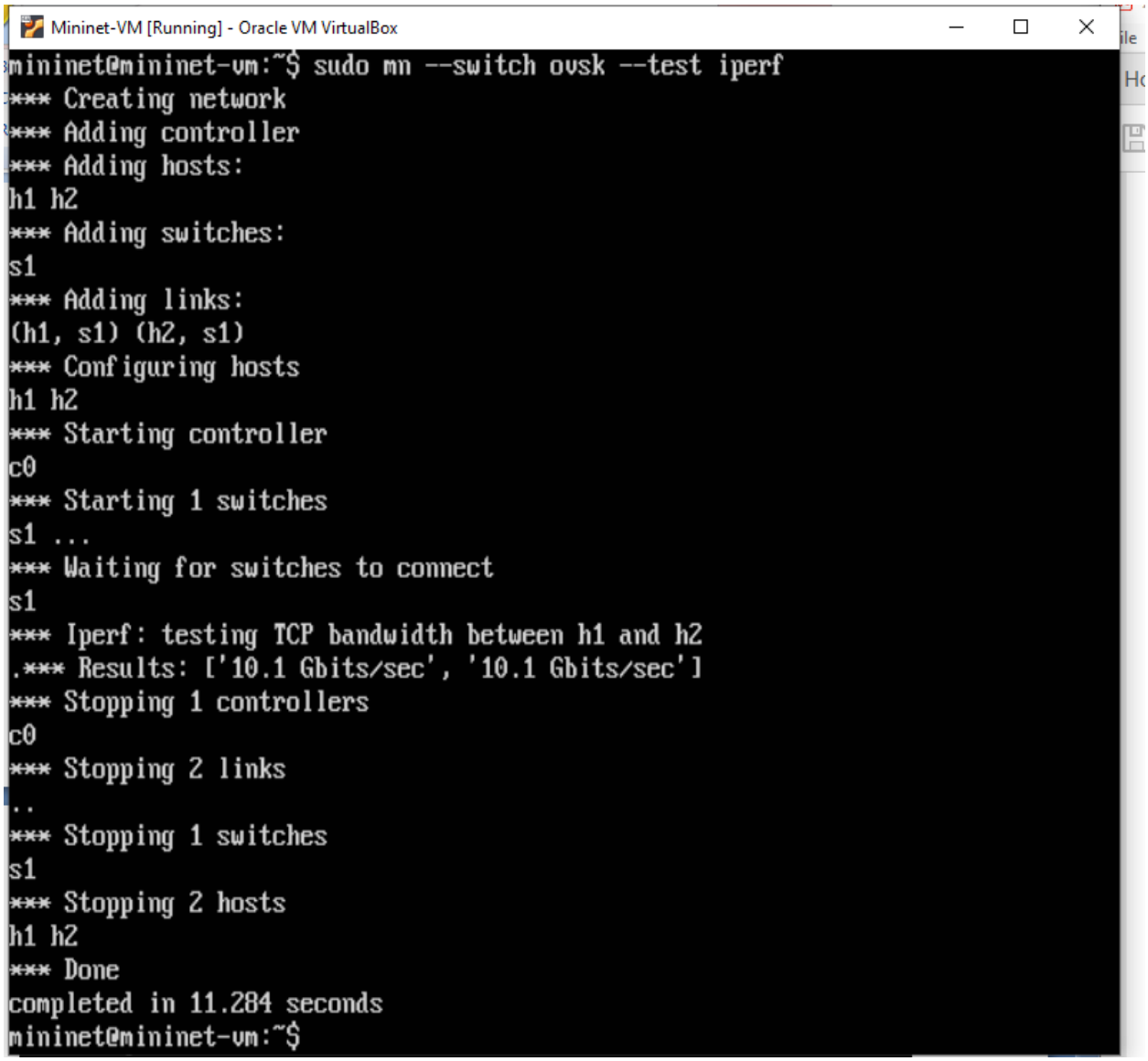
B. Mininet 2.0 allows you to set link parameters, and these can even be set automatially from the command line



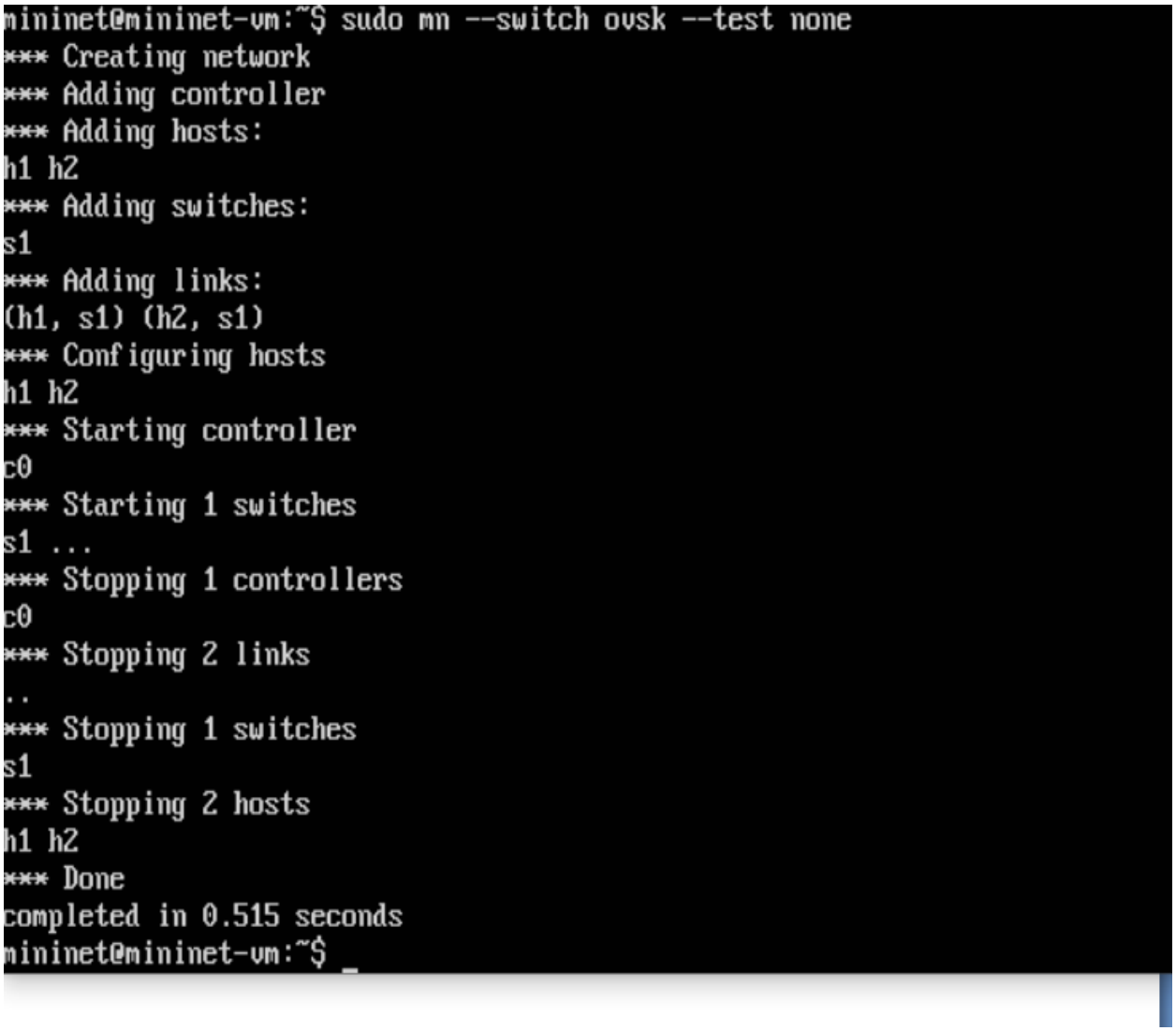
C.The --mac option is super-useful, and sets the host MAC and IP addrs to small, unique, easy-to-read IDs



D.The iperf-reported TCP bandwidth should be similar to the OpenFlow kernel module, and possibly faster



F.To record the time to set up and tear down a topology, use test ‘none’



G.To put switches in their own namespace, pass the innamespace option:

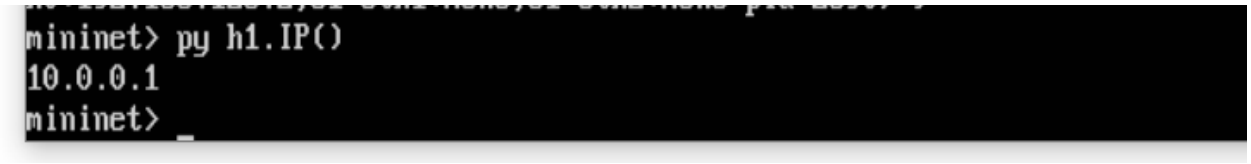
```
mininet@mininet-vm:~$ sudo mn --innamespace --switch user
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
c0 <-> s1
*** Testing control network
s1 -> c0
c0 -> s1
*** Results: 0% dropped (2/2 received)

*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1
*** Starting CLI:
mininet>
```

H. At the Mininet CLI, run:

```
*** Starting CLI:
mininet> py 'hello' + ' ' + 'sohan-it18011'
hello sohan-it18011
mininet>
```

I.. Print the accessible local variables

J. I can also evaluate methods of variables:



Conclusion:

This is the command of mininet walkthrough. have known that the walkthrough of mininet, from this labM. I install mininet in my virtual box then I run the command.This is very interesting for me for the first time. I take help from my senior brothers to complete my work.I also known that how to crate virtual hosts,switches,controls,and link.Mininet supports research ,development,learning,testing,and many other task.Alhamdulilla,I completed my work without any problem.