

## Connecting ONOS to Mininet

1. Run Mininet before onos
2. Start Mininet GUI  
`$sudo ~/mininet/examples/miniedit.py`
3. Move to onos directory in new terminal
4. Start onos server using Bazel  
`$ bazel run onos-local -- clean debug`
5. Start onos using localhost  
`$ onos localhost`
6. In Mininet Controller must be set as remote and port 6653 and saved
7. Run Mininet edit
8. Open Onos web (username : onos , pass: onos)-> The topology must be visible
9. Make sure FWD application is activated through onos  
`app activate org.onosproject.fwd`
10. Test reachability between h1 and h2

**H1 terminal:** ping 10.0.0.2

This should return something like

```
root@sdn-VirtualBox:/home/sdn# ping 10.0.0.2
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=15.3 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.318 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.107 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.043 ms
^C
--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3057ms
```

11. Collect results using iperf3 (start server in h2 and client in h1, save results in results.txt)

**H2 terminal:** iperf3 -s

**H1 terminal:** iperf3 -c 10.0.0.2 -t 15 -l 1 > results.txt

12. Output will look like this

```
"Host: h2"
-----
Server listening on 5201
-----
Accepted connection from 10.0.0.1, port 40608
[ 19] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 40610
[ ID] Interval      Transfer    Bitrate
[ 19] 0.00-1.00    sec   869 KBytes  7.12 Mbits/sec
[ 19] 1.00-2.00    sec   899 KBytes  7.36 Mbits/sec
[ 19] 2.00-3.00    sec   903 KBytes  7.40 Mbits/sec
[ 19] 3.00-4.00    sec   886 KBytes  7.26 Mbits/sec
[ 19] 4.00-5.00    sec   895 KBytes  7.33 Mbits/sec
[ 19] 5.00-6.00    sec   869 KBytes  7.12 Mbits/sec
[ 19] 6.00-7.00    sec   904 KBytes  7.41 Mbits/sec
[ 19] 7.00-8.00    sec   899 KBytes  7.36 Mbits/sec
[ 19] 8.00-9.00    sec   914 KBytes  7.48 Mbits/sec
[ 19] 9.00-10.00   sec   892 KBytes  7.31 Mbits/sec
[ 19] 10.00-11.00  sec   903 KBytes  7.40 Mbits/sec
[ 19] 11.00-12.00  sec   890 KBytes  7.29 Mbits/sec
[ 19] 12.00-13.00  sec   885 KBytes  7.25 Mbits/sec
[ 19] 13.00-14.00  sec   876 KBytes  7.17 Mbits/sec
[ 19] 14.00-15.00  sec   896 KBytes  7.34 Mbits/sec
[ 19] 15.00-15.00  sec    61.0 Bytes  5.08 Mbits/sec
```

13. To check results use “cat”

**H1 terminal :** #cat results.txt

14. Cut only the intervals and bitrate

**H1 terminal:** #cat results.txt | grep sec | head -15 | tr - “ ” | awk ‘{print \$4,\$8}’ > new\_result.txt

15. Check new results

**H1 terminal :** #cat new\_result.txt

16. Plot using Gunplot

**H1 terminal:**

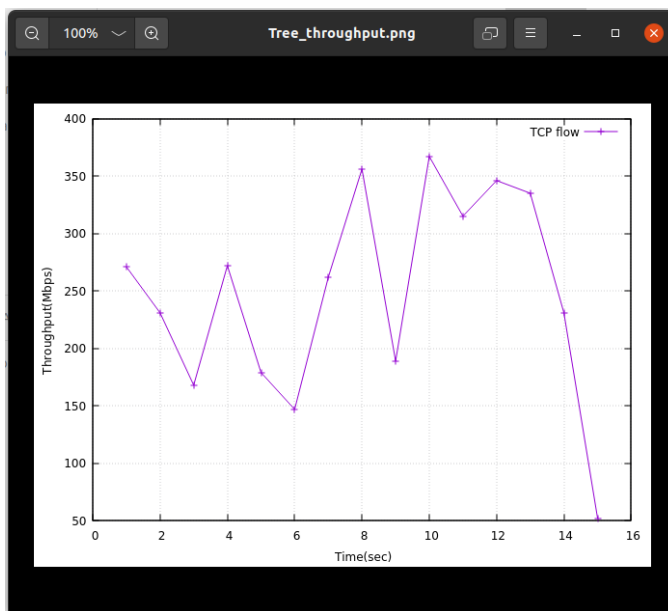
#gnuplot

#plot “ new\_result.txt” title “ccc” with linespoint

# set xlabel “Time(sec)”

#set ylabel “Throughput(Mbps)”

# replot



17. Export the image

18. Use iperf and UDP to get delay jitter and #of packets lost with a specific bit rate

**H2 terminal:** iperf -s -u

**H1 terminal:** iperf -c 10.0.0.2 -u -t 15 -b 3Mb -l 1 > results.txt

## Get Running apps in ONOS

```
onos> apps -a -s
```

## Getting devices in ONOS

```
onos> devices -s
```

## Checking port info

```
onos> ports -s device:deviceID
```

## Checking interfaces configurations

```
onos> interfaces
```