Lab of Network Software

Yaman Parasher 04/06/2020

Introduction:

- TenPings app is a kind of hybrid bidriectional app that when receives an IP packet (i.e., a ping request) configures required flow rules for bidirectional communication between the source (src) and the destination (dst) hosts
- Moreover, TenPings app memorizes the history of received ping requests with time details in a separate datebase
- TenPings app also provides two CLI commands: show the history of performed pings using "tenping list" command & reset the history of performed pings using the “tenpings reset” command

Preliminary Steps:

1. Starting ONOS
2. Starting MIninet & displaying network topology on ONOS GUI interface
3. Starting “Reactive Forwarding” app in ONOS which will help in installation of flow rules one by one at each of the openflow switch
4. Running "pingall" command in Mininet
5. Showing List of disovered host on ONOS GUI interface by enabling the show host button located inside the bottom left bar of the interface.
6. Stopping the “Reactive Forwarding” app from the application section of ONOS GUI interface
7. Starting the “ARP proxy” app from the application section of ONOS GUI interface

According to step 4 of preliminary section, Figure 1 represents the network topology with OpenFlow switches and host connected together.

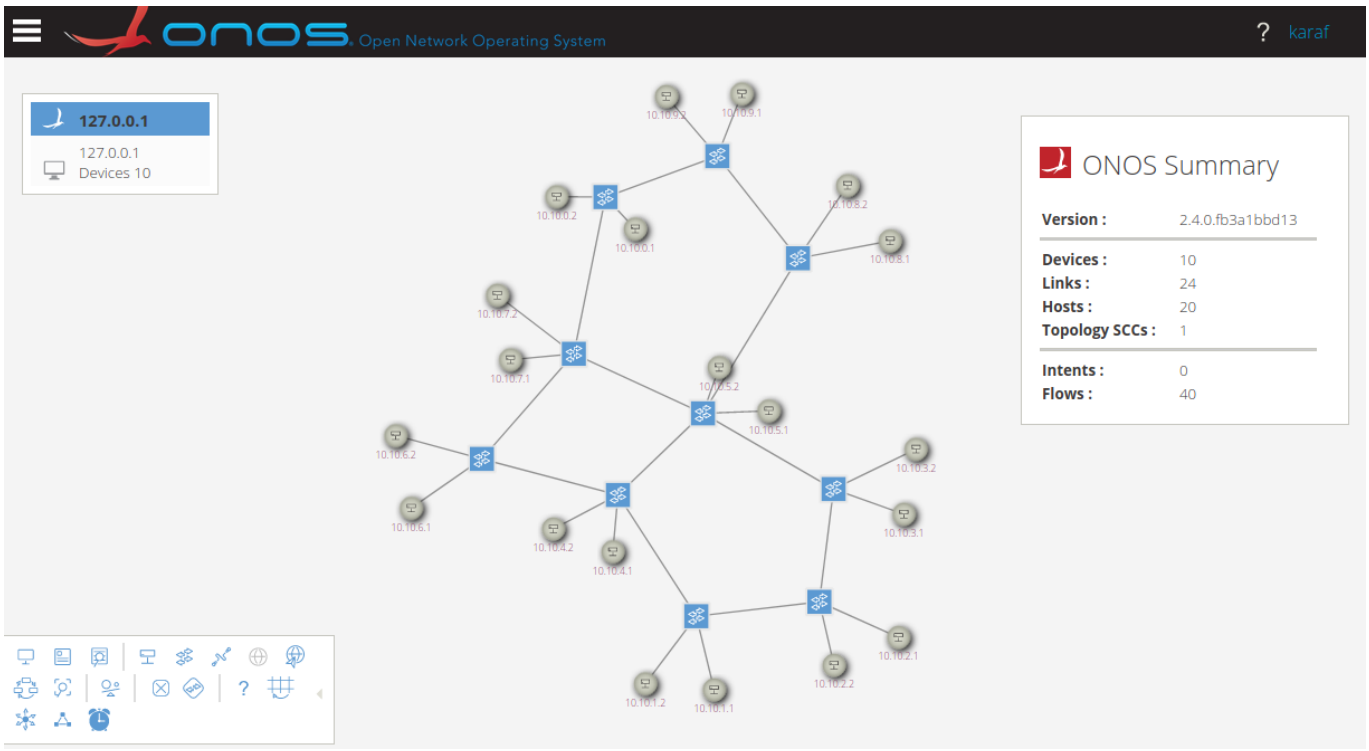

Figure 1: Schematic of Network Topology displaying Openflow Switches & Connected Hosts

The next Figure 2 represents the ping test performed for all the host connected in the topology. 0% dropped defines that all the host are able to communicate successfully with all other host present in the topology. Meaning they all are logially & physically connected & are reachable to each other.

```
mininet> pingall
*** Ping: testing ping reachability
h01 -> h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h02 -> h01 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h11 -> h01 h02 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h12 -> h01 h02 h11 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h21 -> h01 h02 h11 h12 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h22 -> h01 h02 h11 h12 h21 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h31 -> h01 h02 h11 h12 h21 h22 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h32 -> h01 h02 h11 h12 h21 h22 h31 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h41 -> h01 h02 h11 h12 h21 h22 h31 h32 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h42 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h51 h52 h61 h62 h71 h72 h81 h82 h91 h92
h51 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h52 h61 h62 h71 h72 h81 h82 h91 h92
h52 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h61 h62 h71 h72 h81 h82 h91 h92
h61 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h62 h71 h72 h81 h82 h91 h92
h62 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h71 h72 h81 h82 h91 h92
h71 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h72 h81 h82 h91 h92
h72 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h81 h82 h91 h92
h81 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h82 h91 h92
h82 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h91 h92
h91 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h92
h92 -> h01 h02 h11 h12 h21 h22 h31 h32 h41 h42 h51 h52 h61 h62 h71 h72 h81 h82 h91
*** Results: 0% dropped (380/380 received)
```

Figure 2: Pingall Command of Mininet

TenPings app installation:

In this section, at the very first stage I build the downloaded tenpings app using "mvn clean install" command that triggers the jar packaging of the app. This includes compiling the sources, executing the tests and packaging the compiled files in a JAR file. Last step the install phase installs the resulting artifact into the local repository, so it can be used as dependencies by other Maven builds. After that I have also upload the app on the ONOS GUI interface to activate it directly.

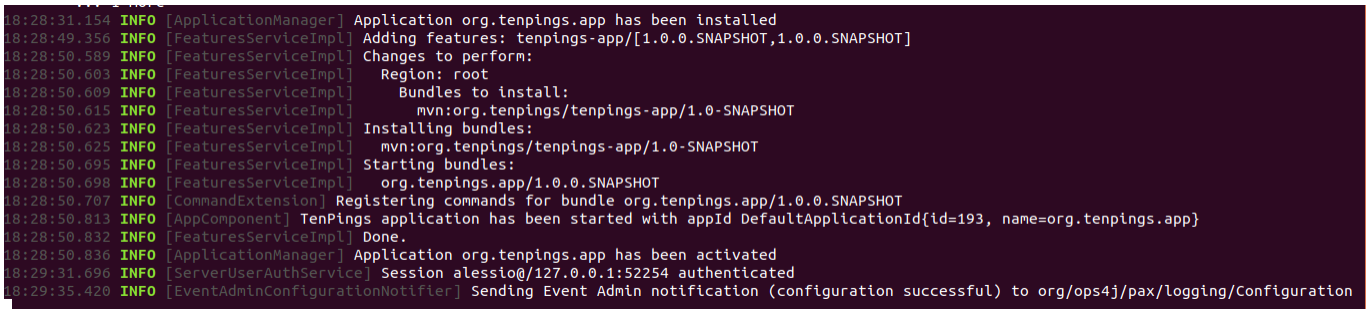

Figure 3: Log showing Installation of Tenpings App

Figure 3 represent the log generated after the installation & activation of the tenpings-app.

TenPings app execution:

In this section, I performed a ping between two hosts. First between ***h01*** & ***h41*** and then between ***h21*** & ***h61***, till the transfer of 7 packets for each case. From Figure 4 & 5, it could be clearly seen that the first ping took a bit long time due because when the switch connected to source host (***h01***) sends the packet received from the source host to the controller, the controller took some time to discover destination of the packet and meanwhile do path computation so that it can install the flow reach on each device or switch traversed by the packet all the way to the destination, before sending out the packet out message to the first switch. The ARP packet here are managed by the ARP proxy application which simplies the environment by making the controller already know about the host that are connected.

After this, I generate a log of the performed actions using log:tail command in the ONOS CLI which can be seen through Figure 5. Here it shows the installation of flow rules along the path computed when the controller discovered the packet from the switch which was directly connected to the source host ***h01***. Similary after running "tenpings list" command in the ONOS CLI, I got the details about list of ping performed. This could be seen in Figure **??**. As instructed in the end, all the flow-rules installed by the app were removed using REST API interface to start from scratch.

```
mininet> h01 ping h41
PING 10.10.4.1 (10.10.4.1) 56(84) bytes of data.
64 bytes from 10.10.4.1: icmp_seq=1 ttl=64 time=63.9 ms
64 bytes from 10.10.4.1: icmp_seq=2 ttl=64 time=2.03 ms
64 bytes from 10.10.4.1: icmp_seq=3 ttl=64 time=1.73 ms
64 bytes from 10.10.4.1: icmp_seq=4 ttl=64 time=1.29 ms
64 bytes from 10.10.4.1: icmp_seq=5 ttl=64 time=1.41 ms
64 bytes from 10.10.4.1: icmp_seq=6 ttl=64 time=1.37 ms
64 bytes from 10.10.4.1: icmp_seq=7 ttl=64 time=1.62 ms
^C
--- 10.10.4.1 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6010ms
rtt min/avg/max/mdev = 1.297/10.482/63.911/21.813 ms
```

Figure 4: Pinging request between ***h01*** & ***h41***

```
17:06:08.413 INFO [AppComponent] [---TENPINGS---] from of:1000000000000000/1 ETH_TYPE: IPv4
17:06:08.414 INFO [AppComponent] [---TENPINGS---] this PING has been added to database
17:06:08.415 INFO [AppComponent] [---TENPINGS---] received packet from device of:1000000000000000 host 56:39:02:3E:3B:9B/None to host 16:53:96:EC:58:45/None
17:06:08.415 INFO [AppComponent] [---TENPINGS---] path:
17:06:08.435 INFO [AppComponent] --- link of:1000000000000000/3->of:1000000000000007/5
17:06:08.435 INFO [AppComponent] --- link of:1000000000000007/3->of:1000000000000005/5
17:06:08.441 INFO [AppComponent] --- link of:1000000000000005/3->of:1000000000000004/4
17:06:08.446 INFO [AppComponent] [---TENPINGS---] i=0 installing rules on device of:1000000000000000 in 1 out 3
17:06:08.453 INFO [AppComponent] [---TENPINGS---] i=1 installing rules on device of:1000000000000007 in 5 out 3
17:06:08.457 INFO [AppComponent] [---TENPINGS---] i=2 installing rules on device of:1000000000000005 in 5 out 3
17:06:08.464 INFO [AppComponent] [---TENPINGS---] i=2 installing rules on device of:1000000000000004 in 4 out 1
17:06:08.473 INFO [AppComponent] [---TENPINGS---] sending packet out to device of:1000000000000000
17:06:13.543 INFO [AppComponent] [---TENPINGS---] from of:1000000000000000/1 ETH_TYPE: ARP
17:06:13.542 INFO [AppComponent] [---TENPINGS---] from of:1000000000000004/1 ETH_TYPE: ARP
17:06:23.965 INFO [EventAdminConfigurationNotifier] Sending Event Admin notification (configuration successful) to org/ops4j/pax/logging/Configuration
```

Figure 5: Log generated by the First Ping between ***h01*** & ***h41***