reH2020 i**P** **O**ver **I**c**N**– the bet**T**er IP (POINT)

HowTo-SDN

Installation and Configuration of the OpenVSwitch and the OpenDayLight Controller in POINT

List of Authors:

George Petropoulos, Mays AL-Naday, Marievi Xezonaki

1. Introduction

2. OpenVSwitch (OVS) Installation for POINT

- 2.1. Prerequisites
- 2.2. Configuration Instructions

3. Opendaylight POINT Application

- 3.1. Prerequisites
- 3.2. Build Instructions
- 3.3. Execution Instructions

4. Appendix

1. Introduction

This document describes the steps required to install and configure an SDN environment. It comprises two main parts:

- SDN Switching with OpenVSwitch: covering the configuration of OVS-based switches in the POINT network, both for soft switches in linux boxes, and the PICA hardware.
- SDN Control: covering the download, installation and configuration of the OpenDayLight SDN Controller to realise the ICN over SDN functions in the POINT network.

There are two different options to deploy the ODL controller:

(a) using the packaged Opendaylight Boron release artifacts (*but then there will be no control or monitoring of the deployed ICN network i.e. the POINT functionalities will not exist*), and

(b) using the POINT Opendaylight application, which also packages a complete ICN-aware Opendaylight container.

Only the second option will be presented in this document.

Appropriate commands to install prerequisite software and configure the development environment are provided, as well as compilation and execution instructions. Finally, the required Blackadder configuration file changes to deploy an ICN topology over SDN networks are specified.

2. OpenVSwitch (OVS) Installation for POINT

Here we define the steps required to install one major version of OVS, that is 2.3.3. This version has been verified to function as expected in different switch formats including: *Pica* hardware switches, linux bare-metal, linux VM (Vbox, KVM and VMWare) and in mininet 2.2.1. For a more comprehensive description of different tested versions and their performance, please check the Appendix in this document. In case OVS is used with Mininet, the steps of Section 2.1 must be followed inside the Mininet VM and the steps of Section 2.2 do not need to be executed.

Note: OS versions with a kernel newer than 3.14.x do not allow to install OVS 2.3.3.

2.1. Prerequisites

To install Openvswitch v2.3.3, follow the defined steps:

```
~ sudo -s
# cd /root
# wget http://openvswitch.org/releases/openvswitch-2.3.3.tar.gz
# tar zxvf openvswitch-2.3.3.tar.gz

# cd openvswitch-2.3.3/
# ./configure --prefix=/usr --with-linux=/lib/modules/`uname
-r`/build --disable-ssl
# make
# make install
# make modules_install
# rmmod openvswitch
# depmod -a
```

To disable openvswitch controller from starting on boot:

```
# /etc/init.d/openvswitch-controller stop
# update-rc.d openvswitch-controller disable
```

To start OVS server process:

```
# /etc/init.d/openvswitch-switch start
```

Verify that Openvswitch is properly installed:

```
# ovs-vsctl show
```

You can follow the exact same steps for any Openvswitch version. Just replace version 2.3.3 in the respective commands above with the desired version, e.g. 2.3.0.

2.2. Configuration Instructions

- The first thing one would need to set up in an OVS switch is a bridge that supports OpenFlow 1.3 protocol. To do so, use the run the following in with admin/root privileges (i.e. sudo):

  ```
  $ ovs-vsctl add-br <BRIDGE_NAME>
  ```

  ```
  $ ovs-vsctl set bridge <BRIDGE_NAME> protocols=OpenFlow13
  ```

  If the switch is a PICA8 hardware, the last line need to be extended to include datapath type:

  ```
  $ ovs-vsctl set bridge -- <BRIDGE_NAME> datapath_type=pica8
  protocols=OpenFlow13
  ```

- Next would be to add switch interfaces to the bridge and configure them to support flows with explicit port numbers, to do so run (again with an admin/root account):

  ```
  $ ovs-vsctl add-port <BRIDGE_NAME> <IF_NAME> -- set interface
  <IF_NAME> options:key=flow ofport_request=<PORT_NUMBER>
  ```

  If the switch is a PICA8 hardware, then the last line need to be extended to configure the port type as ‘pica8’:

  ```
  $ ovs-vsctl add-port <BRIDGE_NAME> <IF_NAME> -- set interface
  <IF_NAME> type=pica8 options:key=flow
  ofport_request=<PORT_NUMBER>
  ```

- With Pica8 hardware, one would need to configure the port mod to be ‘up’, in order for the interface to be activated, as follows:

  ```
  $ ovs-ofctl -O OpenFlow13 mod-port <BRIDGE_NAME> <IF_NAME> up
  ```

- If the switch should also be connected to a SDN controller (e.g. ODL), then the controller must be set for the bridge - provided that the controller is reachable from the switch - in addition to the fail mode. The latter defines the behaviour of the switch in case the controller becomes unreachable:

```
$ovs-vsctl set-controller <BRIDGE_NAME>
tcp:CONTROLLER_IP_ADDRESS:CONTROLLER_TCP_PORT
```

The controller tcp port is normally set to `6633`.

```
$ovs-vsctl set-fail-mode <BRIDGE_NAME> <secure|standalone>
```

Upon completion of these configurations, the switch is ready to be deployed in a POINT network.

3. Opendaylight POINT Application

This section presents the requirements and instructions to install and deploy the implemented Opendaylight POINT application. Eventually, the POINT software modules are packaged with additional required Opendaylight modules and artifacts, and a full deployment of the Opendaylight SDN controller is available.

3.1. Prerequisites

1. Install OpenJDK 8.

   For any Ubuntu version (14-16.04), execute the following commands:

   ```
   ~ sudo add-apt-repository ppa:webupd8team/java -y
   ~ sudo apt-get update
   ~ sudo apt-get install oracle-java8-installer
   ~ sudo update-alternatives --config java
   ~ sudo update-alternatives --config javac
   ```

2. Install Maven 3.3.9.

   ```
   ~ wget
   http://apache.mirrors.lucidnetworks.net/maven/maven-3/3.3.9/binaries/apache-maven-3.3.9-bin.tar.gz
   ~ tar -zxf apache-maven-3.3.9-bin.tar.gz
   ~ sudo cp -R apache-maven-3.3.9 /usr/local
   ~ sudo ln -s /usr/local/apache-maven-3.3.9/bin/mvn /usr/bin/mvn
   ```

   `~ mvn -version` (to verify Maven installation)

3. As the POINT-enabled SDN application is an Opendaylight Boron application, it is essential that the development environment for implementing such applications must be configured properly. Hence, follow the provided instructions by the Opendaylight community, available at: https://wiki.opendaylight.org/view/GettingStarted:Development_Environment_Setup Specifically, update the Maven settings to include the Opendaylight repositories.

   ```
   ~ mkdir ~/.m2
   ~ wget -q -O -
   https://raw.githubusercontent.com/opendaylight/odlparent/master
   /settings.xml > ~/.m2/settings.xml
   ```

3.2. Build Instructions

From the root POINT folder, execute the following commands:

```
~ cd sdn/
~ mvn clean install -DskipTests
```

To generate the javadoc documentation execute the following command. It will be available at the target/site/apidocs/index.html.

```
~ mvn javadoc:javadoc
```

To generate the doxygen documentation execute the following command. It will generate the html and latex documentation at target/doxygen/html/index.html and target/doxygen/latex/index.html.

```
~ doxygen doxygen.config
```

3.3. Execution Instructions

To run the POINT-enabled application execute the following commands:

```
~ ./distribution/opendaylight-karaf/target/assembly/bin/karaf
```

Install the POINT feature with the following commands in the Opendaylight console:

```
> feature:install tm-sdn
> feature:install point-ui
```

To check the execution log, execute the following command. You will see a few messages declaring that ICN modules have been successfully initialized and configured.

```
> log:tail
```

To also check the Opendaylight user interface, use your browser to navigate to http://localhost:8181/index.html. In the left menu you will also see the ICN-SDN tab.

To shutdown the Opendaylight controller, execute:

```
> shutdown -f
```

4. Appendix

OVS Version	**Pica8**	**Linux bare-metal**	**Linux VM (Vbox, KVM, VMWare)**	**Mininet 2.2.1, OVS inside mininet**	**Mininet 2.2.1, OVS outside Mininet in the hosting machine**
2.3.x	OK	OK	OK	OK	flooding
2.4.x	?	?	?	Parsing issues, rules accepted but probabilistic matching.	?
2.5.x	?	?	Parsing issue, rules accepted but not matching	Parsing issue, rules accepted but not matching	?
2.6.5	OK	?	?	?	?
2.7.x	?	?	Rules format is not recognised	Rules format is not recognised	?

			and hence rules are rejected at insertion time	and hence rules are rejected at insertion time	