

SNMP4SDN: Hydrogen: Release Plan

Contents

- [Introduction](#)
- [Release Deliverables](#)
- [Release Milestones](#)
- [Expected Dependencies on Other Projects](#)
- [Compatibility with Previous Releases](#)
- [Themes and Priorities](#)
- [Release Note](#)
 - [Release Note on M3](#)
 - [Release Note on M4](#)
 - [Release Note on M5](#)
 - [Release Note for the Formal Release](#)

Introduction

A southbound plugin that can configure off-the-shelf commodity Ethernet switches via SNMP and CLI.

Release Deliverables

Name	Description
SNMP SouthBound Plugin	A southbound plugin that configures Ethernet switches via SNMP.
CLI support	For SNMP SouthBound Plugin to be also capable of configuring Ethernet switches via CLI.
Extension To SAL	Extensions to the SAL configuration APIs.

Release Milestones

Milestone	Offset 1 Date	Deliverables	
M1	7/24/2013	Name	Description
		Release Plan	Candidate Release Plan
M2	8/21/2013	Name	Description
		Release Plan	Final Release Plan
M3	9/18/2013	Name	Description
		SNMP SouthBound Plugin	A southbound plugin that configures Ethernet switches via SNMP.
M4	10/16/2013	Name	Description
		CLI support	For SNMP SouthBound Plugin to be also capable of configuring Ethernet switches via CLI.
		Extension To SAL	Extensions to the SAL configuration APIs.
		API Freeze	
M5	11/13/2013	Name	Description
		Code Freeze	
		Testing	1. Continuous Integration Testing 2. Continuous System Testing

RC0	11/20/2013	Name	Description
		Bug fixes	
RC1	11/27/2013	Name	Description
		Bug fixes	
RC2	12/4/2013	Name	Description
		Bug fixes	
Formal Release	12/11/2013	Name	Description

Expected Dependencies on Other Projects

Depends On	Dependency Description	Needed By	Is in Other Project Release Plan
Controller	Southbound plugin interface	M2	False

Compatibility with Previous Releases

Themes and Priorities

1. Existing SAL API's semantics on Ethernet switch as well as the implementation with the aid of SNMP and CLI.
2. The required extensions to SAL API to take advantage of some things that can be done via SNMP and CLI.

Release Note

Release Note on M3

For the target of being an OpenDayLight SAL plugin to enable Ethernet switch on the SDN paradigm, we support SNMP & CLI as the south bound protocol, also we propose new mechanisms for switch discovery and topology discovery for Ethernet switch in SDN.

So, there are three things to work out as just mentioned, (1) SNMP CLI, (2) switch discovery, and (3) topology discovery. The architecture and design has been posted on wiki. For the first one, SNMP and CLI are used for flow configuration on Ethernet switch. We have completed the implementation of using SNMP to configured flows on the forwarding table. As to using CLI to configured flows on ACL, it will be done by milestone M4.

For the second task, switch discovery, it will be achieved via SNMP trap sent from the switch. Simply speaking, every switch is set to automatically send SNMP trap to the controller after it boot up. We had introduced this approach in the mailing list, and the feedbacks include: agreeing with this approach since it is very similar to how OpenFlow switch is set to join the controller in initial, or other ideas such as using LLDP or using DNS-SD. And then their limitations or concern of prerequisite had been discussed. So we will still use the approach of SNMP trap, which will be done by M4.

For the third task, topology discovery, the plugin queries each switch by SNMP, and each switch responds with its LLDP data, then the switches' topology could be resolved. Now the implementation is done, and passed the unit test. Then, together with when the implementation of switch discovery is done, after milestone M4, the Ethernet switch should be able to appear in the GUI dashboard.

Finally, in terms of SAL API implementation, including the APIs under ContainerListener, the flowprogrammer, flowreader, topology, inventory, and datapacket, most of them will be done by M4, and then by milestone M5 those works related to system integration to make the plugin runnable will be done, and the extended APIs for specific purposes such as disabling STP and broadcast will also be done by milestone M5.

Release Note on M4

1. Functionalities implementation: the functionalities of this plugin, including API implementations using SNMP and CLI, and the solutions for Ethernet switch discovery in SDN, including switch discovery and topology discovery. API implementations using SNMP had been done by M3, and so does the topology discovery. Switch discovery is also done by M4. Their unit tests with a real subnet of Ethernet switches also passed. Then, the rest of the aforementioned functionalities will be done by M5.
2. Integration test: we had written and pushed the code for integration test for flow programming, but we have difficulties to make it work. We plan to meet our original release plan for M4, which is about CLI and API extension, and then after M4 we can continue on solving the integration test problem.

3. API extension implementation: we had proposed some API extension for the purposes of such as disabling STP and flooding. Then as suggested that extensions to be done via MD-SAL, we will expose these APIs through MD-SAL. API freeze is now moved to M5, because that in addition to the API extensions we had proposed, we will add some new API, but we are still working on writing models to create new APIs in MD-SAL. We will propose our design through MD-SAL after M4, and then complete the implementation by M5.

Release Note on M5

1. API implementation using SNMP had been done by M3. As to API implementation using CLI, including IConfigService and some of the existing OpenDaylight APIs, IConfigService is done by M5, and the others will be done in the end of this year (including flow configuration on ACL as well as reading node's or nodeConnector's properties and statistics).

2. The proposed API extension is exposed in the IConfigService, whose implementation depends on switch's own CLI. The implementation for D-link DES-3200 switch is done as an example of IConfigService implementation. The API extension with MD-SAL will be done in January of next year.

Release Note for the Formal Release

Features

- An OpenDayLight SAL plugin to make commodity Ethernet switches become SDN compatible device: this plugin can configure the forwarding table (assisted by SNMP) and configure specific functions such as disabling STP, flooding, etc (assisted by CLI)
- Analogous to the behavior of OF switch, mechanisms for switch discovery and topology discovery on Ethernet switch are proposed and realized (take advantage of SNMP trap) so that the controller will be notified of the existence of a new switch and its ports, after the switch reboots.
- The aforementioned functions could successfully work on a real Ethernet switch product
- Building ServiceProvider with SNMP4SDN is passed
- More efforts on the regression test of this plugin and with the controller will be made

Non-Code Aspects (user docs, examples, tutorials, articles)

- Posted on OpenDaylight wiki ([Project Proposal](#))

Security Issues

- Authentication information is stored in a file (SNMP community, CLI username, CLI password) ==> the data in the file could be encrypted for security.

Quality Assurance (test coverage, etc)

- 18 APIs, 20 unit tests in total:
 - FlowProgrammer: 5 APIs, 5 unit tests (all passed)
 - ConfigService (extended APIs): 13 APIs, 13 unit tests (all passed)
 - Switch discovery: 1 unit test (passed)
 - Topology discovery: 1 unit test (passed)

Standards (summary of standard compliance)

- SNMP v2c

Schedule (initial schedule and changes over the release cycle)

- Flow configuration on ACL is not included in this release