Network Intent Composition

Welcome to Network Intent Composition

Introduction

This Network Intent Composition project will enable the controller to manage and direct network services and network resources based on describing the “Intent” for network behaviors and network policies. Intents are described to the controller through a new NorthBound Interface, which provides generalized and abstracted policy semantics instead of specific configuration commands. The Intent based NBI allows for a descriptive way to get what is desired from the infrastructure, unlike the current SDN interfaces which are based on describing how to provide different services. This NBI will accommodate orchestration services and network and business oriented SDN applications. The Network Intent Composition function will use existing OpenDaylight Network Service Functions and Southbound Plugins to control both virtual and physical network devices. The Network Intent Composer will be designed to be protocol agnostic such that any control protocol can be used such as Openflow, OVSDB, I2RS, Netconf, BGP etc.

This project provides a structure capable to support different Intent definitions and work with multiple southbound plugins at the same time, also contains its own implementation for Intent Lifecycle Management.

NIC basics

Config dev environment

1. http://docs.opendaylight.org/en/stable-boron/developer-guide/network-intent-composition-(nic)-developer-guide.html
2. https://wiki.opendaylight.org/view/GettingStarted:Development_Environment_Setup
3. https://wiki.opendaylight.org/view/OpenDaylight_Controller:Gerrit_Setup
4. https://wiki.opendaylight.org/view/GettingStarted:Pulling,_Hacking,_and_Pushing_All_the_Code_from_the_CLI

Tips

- Debugging: ./karaf debug

It will listen to 5005 port

Basic guide using Intents for QoS

QoS attribute mapping

Project Facts

Project Creation Date: January 22nd, 2015

Lifecycle State: Incubation

Type: Application

Primary Contact: Yrineu Rodrigues - yfrfelipe@gmail.com

Project Lead: Yrineu Rodrigues - yfrfelipe@gmail.com

Committers:

Active committers:

- Yrineu Rodrigues - yfrfelipe@gmail.com username: yrineu_rodrigues

Inactive committers:

- Anu Mercian - freebirdsanu@gmail.com: amercian
- Dave Lenrow – david.lenrow@huawei.com
- Saket Mahajani - saket.mahajani@gmail.com username: saket
- Raphael Amorim - raphael.amorim@gmail.com username: raphaelamorim
- Icaro Camelo - icarorvc@gmail.com username: icarocamelo

Active reviewers:

- Raphael Amorim - raphael.amorim@gmail.com username: raphaelamorim
- Icaro Camelo - icarorvc@gmail.com username: icarocamelo
- Yrineu Rodrigues - yfrfelipe@gmail.com username: yrineu_rodrigues

Mailing List: nic-dev@lists.opendaylight.org
Archives: mailing list archives

Meetings: See Community Meetings

Repository: git clone https://git.opendaylight.org/gerrit/nic

Jenkins: jenkins silo

1. Create a basic topology using mininet

```
        sudo mn --topo single --mac --controller=remote,ip=127.0.0.1,
port=6633 --switch ovsk,protocols=OpenFlow13
```

2. Run karaf distribution and install odl-nic-core-mdsal, odl-nic-console, odl-nic-listeners

```
       karaf>feature:install odl-nic-core-mdsal odl-nic-console odl-nic-
listeners
```

3. Create an Intent qos:

```
        intent:qosConfig -p High_Quality -d 46
```

Note: The DSCP value ranges from 0-63

4. Create an Intent to allow traffic between two devices applying the QoS service

```
        karaf>intent:add -a ALLOW -t 00:00:00:00:00:01 -f 00:00:00:00:00:02 -
q QOS -p High_Quality
```

5. Verify if a new rule was created on OF switch with a field 'mod_nw_tos:184'

Basic guide to create Intents to ALLOW or BLOCK

1. Create a basic topology using mininet

```
        sudo mn --topo single --mac --controller=remote,ip=127.0.0.1,
port=6633 --switch ovsk,protocols=OpenFlow13
```

2. Run the karaf distribution and install odl-nic-core-mdsal, odl-nic-console, odl-nic-listeners

```
       karaf>feature:install odl-nic-core-mdsal odl-nic-console odl-nic-
listeners
```

3. Create an Intent to allow all traffic between two devices

```
        karaf>intent:add -f 00:00:00:00:00:01 -t 00:00:00:00:00:02 -a ALLOW
        karaf>intent:add -f 00:00:00:00:00:02 -t 00:00:00:00:00:01 -a ALLOW
```

4. Test the connectivity between those two devices, must be allowed

5. To test the BLOCK action, please, execute steps 1, 2 and 3 changing the action ALLOW to BLOCK

6. Verify the connectivity between those two devices, must be blocked

List of available commands

Name	Description
intent:add	Adds an intent to the configuration tree
intent: remove	Removes an intent from the configuration tree
intent:show	Shows the details of an intent
intent:list	List all the intents in both operational (default) and configuration trees
intent: compile	Runs the conflict detection and resolution algorithm for the current intents in the configuration tree

How to install

In order to install NIC your have to open the karaf CLI. Once your opendaylight karaf console shows up, type:

Gerrit Patches: code patches /reviews

Bugs:

- open bugs

```
    ________                       ________                .__  .__       .
__   __
   \_____  \ ______   ____   ____ \______ \ _____  ___.__.|  | |__| ____ |
|___/  |_
     /   |   \\____ \_/ __ \ /    \ |    |  \\__  \<   |  ||  | |  |/ ___\|
|  \   __\
    /    |    \  |_> >  ___/|   |  \|    `   \/ __ \\___  ||  |_|  / /_/  >
Y  \  |
    \_______  /   __/ \___  >___|  /_______  (____  / ____||____/__\___
/|___|  /__|
            \/|__|        \/     \/        \/     \/\/
/_____/      \/


Hit '<tab>' for a list of available commands
and '[cmd] --help' for help on a specific command.
Hit '<ctrl-d>' or type 'system:shutdown' or 'logout' to shutdown OpenDaylight.

opendaylight-user@root>feature:install odl-nic-console
```

That command will install odl-nic-core as well, if not installed. After feature installation is done you can check the status of all bundles by executing:

```
opendaylight-user@root>bundle:list | grep intent
252 | Active   |  80 | 1.0.0.SNAPSHOT                          |
intentengine-api
253 | Active   |  80 | 1.0.0.SNAPSHOT                          |
intentengine-impl
254 | Active   |  80 | 1.0.0.SNAPSHOT                          |
intentengine-compiler
255 | Active   |  80 | 1.0.0.SNAPSHOT                          |
intentengine-cli
```

After all the bundles are active, if you type intent<tab> you should be able to see the list of commands:

```
opendaylight-user@root>intent
intent            intent:add        intent:compile    intent:list
intent:remove     intent:show
```

List of commands

intent:add

```
DESCRIPTION
        intent:add

        Adds an intent to the controller.

Examples: --actions [ALLOW] --from <subject> --to <subject>
          --actions [BLOCK] --from <subject>

SYNTAX
        intent:add [options]

OPTIONS
        -a, --actions
                Action to be performed.
                -a / --actions BLOCK/ALLOW
                (defaults to [BLOCK])
        --help
                Display this help message
        -t, --to
                Second Subject.
                -t / --to <subject>
                (defaults to any)
        -f, --from
                First subject.
                -f / --from <subject>
                (defaults to any)
```

intent:remove

```
DESCRIPTION
        intent:remove

        Removes an intent from the controller.

SYNTAX
        intent:remove id

ARGUMENTS
        id
                Intent Id
```

intent:show

```
DESCRIPTION
        intent:show

        Shows detailed information about an intent.

SYNTAX
        intent:show id

ARGUMENTS
        id
                Intent Id
```

intent:list

```
DESCRIPTION
        intent:list

        Lists all intents in the controller.

SYNTAX
        intent:list [options]

OPTIONS
        -c, --config
                List Configuration Data (optional).
                -c / --config <ENTER>
        --help
                Display this help message
```

intent:compile (EXPERIMENTAL)

```
DESCRIPTION
        intent:compile

        [EXPERIMENTAL] Compile all intents and provide the results after
conflict resolution

SYNTAX
        intent:compile
```

Compilation process demo

This is an initial and simplified implementation of the conflict resolution algorithm. In this version there is a temporary endpoint- selector which expects a CSV list of IP addresses. There is also a new experimental CLI command (intent:compile) to compile intents and show the results of the conflict resolution. See the steps to simulate an example of conflict resolution below:

```
    opendaylight-user@root>intent:add -f 10.0.0.5 -t 10.0.0.2,10.0.0.3 -a
ALLOW
    Intent created (id: beb65aed-c88c-4979-9640-9b48fd30e39b)
    opendaylight-user@root>intent:add -f 10.0.0.5 -t 10.0.0.2,10.0.0.10 -a
BLOCK
    Intent created (id: c93c2c02-14d8-4ebe-a121-8dfceee80c18)
    opendaylight-user@root>intent:add -f 10.0.0.1,10.0.0.4 -t 10.0.0.2 -a
ALLOW
    Intent created (id: 88a752fc-28b4-458c-b07b-22d52e0e6177)
    opendaylight-user@root>intent:compile
```

Output:

```
Original policies:

from [10.0.0.5] to [10.0.0.2, 10.0.0.3] apply ALLOW
from [10.0.0.1, 10.0.0.4] to [10.0.0.2] apply ALLOW
from [10.0.0.5] to [10.0.0.2, 10.0.0.10] apply BLOCK

>>> Compiled policies:
from [10.0.0.1, 10.0.0.4] to [10.0.0.2] apply ALLOW
from [10.0.0.5] to [10.0.0.2] apply BLOCK
from [10.0.0.5] to [10.0.0.3] apply ALLOW
from [10.0.0.5] to [10.0.0.10] apply BLOCK

opendaylight-user@root>
```

NOTE: The compilation is not happening in the normal application flow. Currently, the only way to test this feature is using the CLI and the results will not be applied to the network [1].

[1] On Lithium only, but in Beryllium flows will be pushed.

Project Information

Project Proposal

Documentation

Proposed Community Goals for NIC project

Draft for discussion 9/4/2015

1. Solve the problem of resource contention between disparate ODL apps/services (e.g. conflicting changes to forwarding tables, this is the “multiple writer” problem in today’s ODL). Prove that NIC can enforce/enable cooperation between disparate services, including existing services that are not “cooperation aware”.

2. Provide a controller based, performant, easy to deploy. scale-out neutron/netvirt implementation as one of the services that cooperate within NIC. The goal here is to make ODL controller plugin the obvious openstack network implementation for advanced cloud networking, etc.

See more (recommended):

Clarifying the relation between the community goals for Be and the release plan

1. ODL Summit 2016 Presentations
    a. NIC demo
2. ODL Summit 2015 Presentations
    a. NIC Be Planning discussion (+ Intent Graph Presentation)
    b. What is the Intent Anyway
    c. Boulder Project
3. OpenStack Summit 2015
    a. Panel with Dave Lenrow
    b. Intents Panel with Dave Lenrow

Documents

Affinity_Service_Chaining_Proposal_ODP_7-23-2013.pdf

Clarifying the mapping service and other parts of NIC

Weekly Meetings

Schedule and Logistics

We have weekly meetings every **Friday at 8:00 AM - 9:00 (PST)**.

Meetings are held using **WebEx**.

- Weekly on Friday at 08:00 PST/PDT | 11:00 EST/EDT | 15:00 UTC during Daylight Time/16:00 UTC during Standard Time
    - Webex meeting link
    - Meeting Number: 199 413 454
    - Meeting password: This meeting does not require a password.
    - Call-in toll-free number (US/Canada): +1-855-797-9485
    - Call-in toll number (US/Canada): +1-415-655-0002

- Host Key: 466804
- IRC: Freenode/#opendaylight-nic

Meeting Topics

New entries **ONLY IF** there are special topics to be discussed during the meeting.

Meeting topics

Past Meetings

- Meeting minutes from prior meetings
- Face-to-face meeting at Ciena on 2/18/15

Requirements

Release Planning

Release	Release Plan	Agile Board	Release Notes	Release Review	Installation Guide	User Guide	Developer Guide	Operations Guide	How-To's /Tutorials
Boron	Release Plan	NIC Trello Board							
Beryllium	Release Plan	NIC Trello Board							
Carbon	Release Plan	NIC Trello Board							
Nitrogen	Release Plan	NIC Trello Board							
Oxygen	Release Plan	NIC Trello Board	-	-	-	-			

Release Notes