Cisco Open SDN Controller 1.1 Installation Guide

First Published: March 06, 2015

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

CHAPTER 1 **Overview** 1

System Requirements 1

CHAPTER 2 **Installing Cisco Open SDN Controller** 3

Installing Open SDN Controller—VMware ESXi 3

Installing Open SDN Controller—Oracle VM VirtualBox 7

CHAPTER 3 **Post-Installation Setup Tasks** 11

Setting Up BGP-LS and PCEP 11

Single Node Setup 12

Manual Setup 12

RESTCONF Setup 13

Setting Up NETCONF 13

Configuring NETCONF Support on Cisco ASR 9000 Series and IOS XRv Routers 14

Enabling NETCONF Support on the Cisco Open SDN Controller 14

Setting Up OpenFlow 15

Configuring OpenFlow Support on a Cisco ASR 9000 Series Router 15

Configuring OpenFlow Support on a Cisco Nexus 3000 Series Switch 17

Enabling OpenFlow Support on the Cisco Open SDN Controller 18

Configuring OpenFlow Clusters on Cisco ASR 9000 Series Routers 18

Configuring OpenFlow Clusters on Cisco Nexus 3000 Series Switches 19

Example: Configuring OpenFlow Clusters on an OVS Switch 21

REST API Integration with an Open SDN Controller Cluster 22

Installing Wireshark 24

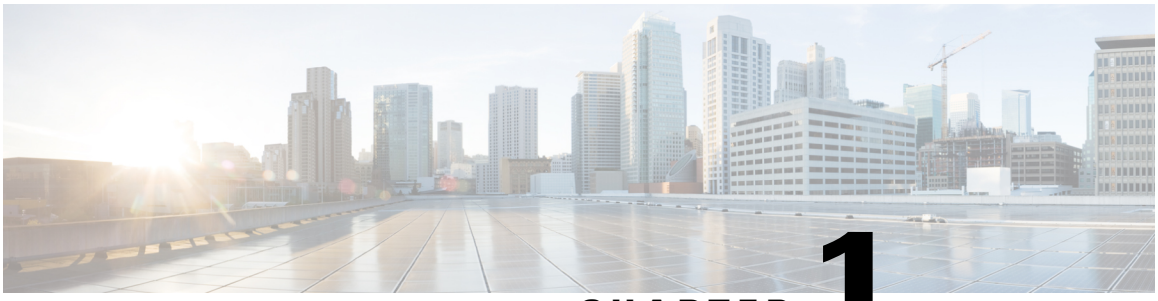
APPENDIX A **Sample BGP-LS and PCEP Configurations** 27

BGP-LS Peer 27

ISIS (Redistributes ISIS link states to BGP-LS) 27

Interfaces: (Loopback for MPLS-TE and BGP-LS) 28

MPLS for PCEP 28



CHAPTER 1

Overview

The following guide covers the prerequisites that need to be met before you install Cisco Open SDN Controller (Open SDN Controller hereafter) and details the installation process. It also covers how to configure support for BGP-LS/PCEP, NETCONF, and OpenFlow.

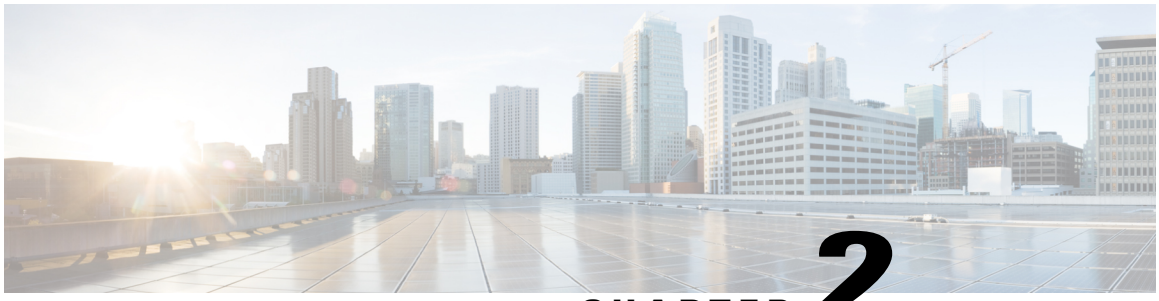
Begin by reviewing the following section:

- System Requirements, page 1

System Requirements

The following table lists what is required to install and run Cisco Open SDN Controller:

Hardware, Software, and OS Requirements	
Disk Space	64 GB
Memory	16 GB RAM
Processor	Intel four-core processor
Virtualization Software	• VMware vSphere 5.1 (which includes EXSi 5.1, vCenter Server 5.1, and vCenter Client 5.1) or later • Oracle VM VirtualBox 4.3 or later



CHAPTER 2

Installing Cisco Open SDN Controller

This chapter contains the following sections:

- Installing Open SDN Controller—VMware ESXi, page 3
- Installing Open SDN Controller—Oracle VM VirtualBox, page 7

Installing Open SDN Controller—VMware ESXi

This procedure describes how to install Open SDN Controller on VMware ESXi hypervisors.

Before You Begin

Note the following points before completing this procedure:

- Hostname resolution is recommended, but not required, for the installation. If you decide to make use of it, specify a fully qualified domain name either in the Deploy OVF Template wizard's Properties page or the /etc/hosts file.
- The config field in the Deploy OVF Template wizard's Properties page can safely be left blank.
- If you are setting up a 3-node cluster, you also need to complete Step 3 for the other 2 nodes in the cluster.
- When configuring Open SDN controller from the browser, use only its IP address in the address bar and configuration fields. Do not use its FQDN name.
- If you are using a multiple Network Interface Card (NIC) setup, have the IP addresses for the second and third NICs ready.
- Have values for the following controller settings ready:
  - IP address
  - Hostname
  - Netmask
  - Gateway address
  - DNS server addresses (recommended, but not mandatory)

- NTP server address (recommended, but not mandatory)

You will need these for Step 3i.

Step 1 Download a copy of the Open SDN Controller distribution file:

a) Access the main Download Software page by opening the URL provided in the confirmation email you received after purchasing Open SDN Controller.

b) If prompted, log into Cisco.com.
If you don't already have a Cisco.com account, sign up for one here.

c) From the navigation pane, select **Products** > **Cloud and Systems Management** > **Network Controllers and Applications** > **Cisco Open SDN Controller** > **Cisco Open SDN Controller 1.1**.
Note You may need to click an item more than once in order to view its sub-elements.

d) From Open SDN Controller's Download Software page, locate osc-fcs-vmware.zip and click **Download**.

e) Unzip the zip file to a local directory.
Two files should now be available: the distribution file (osc-fcs-vmware.ova) and the documentation overview (which links to the documentation provided with this release).

Step 2 Log in to your VMware vSphere client and import the OVA distribution.

Step 3 Complete the Deploy OVF Template wizard:

a) From the client's main menu, select **File > Deploy OVF Template**. The Deploy OVF Template wizard opens.

b) Deploy OVF Template wizard
Click **Browse** to navigate to the OVA distribution.

c) Select the OVA distribution and then click **Open**.

d) Click **Next**.
From the OVF Template Details page, you can view information such as the corresponding product, vendor, and download size.

e) Click **Next**.
From the Name and Location page, you can specify the name and location of the template you are deploying.

f) Click **Next**.
From the Host/Cluster page, you can specify the host or cluster on which to run the template.

g) Click **Next**.
From the Disk Format page, you can select the format in which virtual disks are stored. We recommend that you choose the Thick Provision Lazy Zeroed option.

h) Click **Next**.
From the Network Mapping page, you can specify the networks that the deployed template will use and map every source network with the correct destination network. Although it is not mandatory, we recommend that you pair each source network with a unique destination network.

i) Click **Next**.
From the Properties page, you can customize the configuration settings for this deployment. At this point, you have two options:

- You can leave this wizard page blank and move on to the last wizard page. See Step 3j.
- You can enter the appropriate settings. Values must be entered for the following four fields:

- ip_0 (IP address): Note that you must specify an IPv4 address. IPv6 addresses are not supported.
- host_fqdn (hostname)
- netmask_0 (netmask)
- gateway_0 (gateway address)

Caution If you fail to enter a value for any of these four fields, the installation will not complete successfully.

While they are not required, we recommend that you also enter values for the following fields:

- dns_0 and dns_1 (DNS server addresses)
- ntp (NTP server address)

j) Click **Next**.
From the Ready to Complete page, you can view a summary of all the settings you have specified in the wizard.

k) Click **Finish** to deploy the template. After a few minutes, a dialog box opens and indicates that the deployment has successfully completed.

l) Click **Close**.

Step 4 Determine the controller's IP address, which you will need for Step 5a:

a) From your vSphere client's device list, right-click the controller's entry and then select **Open Console**.

b) Log in to the controller as the sysadmin user with the following credentials:

1. login: sysadmin
2. password: sysadmin

At this point, you are prompted to configure a new password for the sysadmin user.

c) Enter a new password for the admin user and then press the Enter key.
The OS Configuration Console opens.

d) Open the Network Settings page and note the value listed in the IP Address field.

Step 5 Complete the setup wizard:

a) Open the following URL: https://<*controller's-IP-address*>
The setup wizard's Cluster Configuration page opens.

b) Configure the cluster that will run on the controller:

1. Select whether you want to create a one or three-node cluster.
2. Specify the nodes that belong to the cluster and then click **Next**.

Note the following:

- The first field is automatically populated with the IP address of the node you just accessed.
- If you select the three-node option, you will need to enter the IP addresses for the other two nodes you want to add to the cluster before proceeding.
- Open SDN Controller will validate every IP address you enter and indicate whether the address is reachable.

The setup wizard's Network Configuration page opens.

3 Do one of the following:

- Click **Back** to correct any errors you made when specifying the IP addresses of the cluster nodes.
- Click **Skip Multi-nic Config** to skip this wizard page and submit the API (Northbound) interface information you specified in the Cluster Configuration page.
- Enter the relevant device (Southbound) and controller–cluster (East–West) interface information and then click **Submit Network Config**.

  Note the following:

  - Open SDN Controller will use tooltips to indicate any invalid IP addresses you have entered, as well as any fields that require an IP address in order to complete the validation process.
  - This button is available only after you have entered valid network configuration information.

After you click either the Skip Multi-nic Config or Submit Network Config button, the setup wizard's Configuration Status page opens:

- If configuration completes successfully (this typically takes about 5 minutes) and the authentication/login service starts, the setup wizard's Admin Account page opens. Proceed to Step 6.
- If configuration does not complete:

  1 Check ps.log (located in the /opt/cisco/platform/platform-services/data directory) to determine the errors that took place and make the necessary fixes.

  2 Establish an SSH connection with the controller's VM.

  3 Run the following command:

  sudo vi /opt/cisco/platform/platform-services/data/node-config-state

  4 In the resulting file, set the state as Not Configured and then save the file.

  5 Refresh your browser to restart the setup wizard.

  6 Repeat Step 5b.

Step 6 Enter a new passphrase for the admin user twice (the second time to confirm it) and then click **Change Passphrase**.
Note the following:

- We recommend that you do not edit the value already set for the Current Passphrase field.
- Passwords must contain a minimum of 6 characters.
- After the password is changed, you are automatically directed to the Login page.

Step 7 Log in to the controller, using the username **admin** and the password you just set.
Note that the necessary Open SDN Controller plug-ins are installed automatically when you install the controller. To view all of the plug-ins that are available, select **Features** from the main toolbar's Management menu.

Installing Open SDN Controller—Oracle VM VirtualBox

Complete the following procedure to install Open SDN Controller on a virtual machine using Oracle VM VirtualBox. When configuring Open SDN Controller from the browser, use only its IP address in the address bar and configuration fields. Do not use its FQDN name.

Step 1 Download a copy of the Open SDN Controller distribution file:

a) Access the main Download Software page by opening the URL provided in the confirmation email you received after purchasing Open SDN Controller.

b) If prompted, log into Cisco.com.
If you don't already have a Cisco.com account, sign up for one here.

c) From the navigation pane, select **Products** > **Cloud and Systems Management** > **Network Controllers and Applications** > **Cisco Open SDN Controller** > **Cisco Open SDN Controller 1.1**.
Note You may need to click an item more than once in order to view its sub-elements.

d) From Open SDN Controller's Download Software page, locate osc-fcs-virtualbox.zip and click **Download**.

e) Unzip the zip file to a local directory.
Two files should now be available: the distribution file (osc-fcs-virtualbox.ova) and the documentation overview (which links to the documentation provided with this release).

Step 2 Create the virtual machine on which you will install the controller:

a) Launch VirtualBox 4.3.x

b) Select **File** > **Import Appliance**.

c) Navigate to the distribution file you downloaded in Step 1, select it, and then click **Open**.
The required OS, memory, and hard drive settings are automatically set.

Note If you plan to use Open SDN Controller for development purposes only, you can lower the amount of memory allocated to the controller from 16 GB of ram to 8 GB.

d) Click **Import**.

Step 3 Specify the network settings necessary in order for the controller to operate properly on the virtual machine:

a) Select the new virtual machine and then click **Settings** from the VirtualBox main menu.

b) From the Settings menu, click **Network**.

c) If necessary, select the Enable Network Adapter checkbox.

d) From the Attached to: drop-down list, select **Bridged Adapter**.

e) From the Name: drop-down list, select a wired network adapter.
VirtualBox does not provide wireless support for bridged networking.

Note the following:

- If multiple network adapters are listed, select each adapter's tab and repeat Steps 3c through 3e.
- If you plan to create a 3 node cluster in Step 8 of this procedure, complete Steps 3f through 3h. Otherwise, skip ahead to Step 4.

f) Click **Advanced** to display additional network adapter options.

g) From the Adapter Type: drop-down list, select **Paravirtualized Network (virtio-net)**.

h) Click **OK**.

Step 4 Start the virtual machine by doing one of the following:

- Select the virtual machine and then click **Start** from the VirtualBox main menu.
- Right-click the virtual machine and then select Start.

  Once you start the virtual machine, it will automatically grab a DHCP address.

Step 5 Log in to the controller as the sysadmin user with the following credentials:

a) login:sysadmin
b) password: sysadmin
   At this point, you are prompted to configure a new password for the sysadmin user.

Step 6 Enter a new password for the sysadmin user and then press the Enter key.
The OS Configuration Console opens.

Step 7 Determine the controller's IP address, which you will need for Step 8a:

a) Select the **Drop to shell** option.
b) Run the following command: **ifconfig eth0 | grep inet**
c) Note the value listed for **inet addr**.

Step 8 Complete the setup wizard:

a) Open the following URL: https://<*controller's-IP-address*>
   The setup wizard's Cluster Configuration page opens.

b) Configure the cluster that will run on the controller:

   1 Select whether you want to create a one or three-node cluster.

   2 Specify the nodes that belong to the cluster and then click **Next**.

   Note the following:

   - The first field is automatically populated with the IP address of the node you just accessed.
   - If you select the three-node option, you will need to enter the IP addresses for the other two nodes you want to add to the cluster before proceeding.
   - Open SDN Controller will validate every IP address you enter and indicate whether the address is reachable.

   The setup wizard's Network Configuration page opens.

   3 Do one of the following:

   - Click **Back** to correct any errors you made when specifying the IP addresses of the cluster nodes.
   - Click **Skip Multi-nic Config** to skip this wizard page and submit the API (Northbound) interface information you specified in the Cluster Configuration page.
   - Enter the relevant device (Southbound) and controller–cluster (East–West) interface information and then click **Submit Network Config**.

     Note the following:

     ◦ Open SDN Controller will use tooltips to indicate any invalid IP addresses you have entered, as well as any fields that require an IP address in order to complete the validation process.

◦ This button is available only after you have entered valid network configuration information.

After you click either the Skip Multi-nic Config or Submit Network Config button, the setup wizard's Configuration Status page opens:

- If configuration completes successfully (this typically takes about 5 minutes) and the authentication/login service starts, the setup wizard's Admin Account page opens. Proceed to Step 6.
- If configuration does not complete:

  1. Check ps.log (located in the /opt/cisco/platform/platform-services/data directory) to determine the errors that took place and make the necessary fixes.
  2. Establish an SSH connection with the controller's VM.
  3. Run the following command:

     sudo vi /opt/cisco/platform/platform-services/data/node-config-state
  4. In the resulting file, set the state as Not Configured and then save the file.
  5. Refresh your browser to restart the setup wizard.
  6. Repeat Step 5b.

Step 9 Enter a new passphrase for the admin user twice (the second time to confirm it) and then click **Change Passphrase**. Note the following:

- We recommend that you do not edit the value already set for the Current Passphrase field.
- Passwords must contain a minimum of 6 characters.
- After the password is changed, you are automatically directed to the Login page.

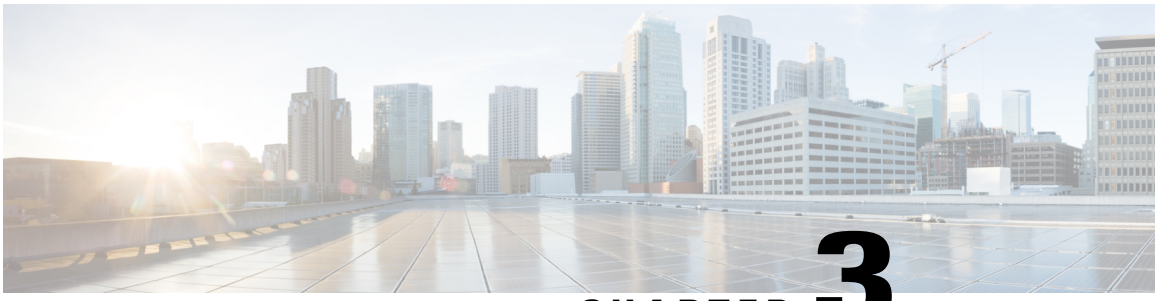
Step 10 Click the arrow at the top of the page to open the Admin Configuration page.

Step 11 (Optional) Create a secondary admin user for the controller by entering the necessary information and then clicking **Create User**. You can skip this step by clicking **Skip User Creation**.

Step 12 Click the arrow at the top of the page to complete cluster configuration and open the Cisco Open SDN controller login page.

Step 13 Log in to the controller.
If you set up a secondary admin user, enter that user's credentials. Otherwise, enter the default admin user's credentials.

Note that the necessary Open SDN Controller plug-ins are installed automatically when you install the controller. To view all of the plug-ins that are available, select **Features** from the main toolbar's Management menu.



CHAPTER 3

Post-Installation Setup Tasks

Now that you have installed Open SDN Controller, complete the tasks described in the following sections to:

- Configure support for BGP-LS/PCEP, NETCONF, and OpenFlow.
- Integrate REST APIs with an Open SDN Controller cluster.
- Install the Wireshark network protocol analyzer.

- Setting Up BGP-LS and PCEP, page 11
- Setting Up NETCONF, page 13
- Setting Up OpenFlow, page 15
- REST API Integration with an Open SDN Controller Cluster, page 22
- Installing Wireshark, page 24

Setting Up BGP-LS and PCEP

In this section, we will cover how to set up BGP Link-State (BGP-LS) and Path Computation Element Protocol (PCEP) support in a single node configuration.

A BGP-LS session is required between Open SDN Controller and one or more BGP-LS speakers. BGP-LS is used to communicate the contents of the Interior Gateway Protocol (IGP) link-state database from the network up to Open SDN Controller.

A PCEP session is required between Open SDN Controller and any router running a Path Computation Client (PCC) and functioning as the head-end of an MPLS TE tunnel. PCEP is used to convey parameters that the router will use to setup an MPLS TE tunnel.

Note that if your network contains a BGP peer that sends 100,000 or more route updates, you need to update the datastore's default configuration in order for the datastore to function properly. To do so:

1. Navigate to the following directory: /opt/cisco/controller/etc
2. Open the datastore configuration file (org.opendaylight.controller.cluster.datastore.cfg) in a text editor.
3. Locate these settings and make the following changes:

- **operation-timeout-in-seconds**—Change from 5 to 30 seconds.
- **shard-transaction-commit-timeout-in-seconds**—Change from 30 to 600 seconds.

4 Save the changes you have made.

The changes will take effect immediately and do not require you to restart the controller.

Single Node Setup

The Cisco Open SDN controller comes preconfigured with both a baseline BGP-LS and PCEP configuration, as specified in the following files:

- 31-bgp.xml
- 41-bgp-example.xml
- 32-pcep.xml
- 39-pcep-provider.xml

See Sample BGP-LS and PCEP Configurations to view examples of what BGP-LS and PCEP configurations look like.

There are two ways for you to set up BGP and PCEP for use with the controller: either manually or via RESTCONF. Select a method and complete its corresponding procedure.

Note Segment routing is not supported in this release of Open SDN Controller.

Manual Setup

Step 1 Install the Cisco Open SDN controller. See the instructions provided earlier in this document.

Step 2 Configure BGP-LS and BGP-LS peers by completing the procedure described at the following URL—https://wiki.opendaylight.org/view/BGP_LS_PCEP:User_Guide
Note the following:

- You will also need to open 41-bgp-example.xml in a text editor and make the following changes to these settings:
  - **local-peer-id**: Enter the controller's IP address.
  - **remote-peer-id**: Enter the IP address for every route reflector or peer in your network, creating a separate entry for each entity.
  - **iana-linkstate-attribute-type**: Set to **true** to ensure that devices work properly with Cisco IOS XR 5.3.0.
- By default, Open SDN Controller uses draft-ietf-pce-stateful-pce-07, which works well with Cisco IOS XR 5.3.0. No additional configuration is required for PCEP.

Step 3 (Optional) Enable the TCP-MD5 modules and services that both BGP-LS and PCEP will use by completing the procedure described at the following URL—https://wiki.opendaylight.org/view/BGP_LS_PCEP:TCP_MD5_Guide#RESTCONF_Configuration

Step 4 To view BGP-LS and PCEP data, open either the BGPLS Manager or PCEP Manager application in Open SDN Controller.

RESTCONF Setup

Step 1 Install the Cisco Open SDN controller. See the instructions provided earlier in this document.

Step 2 Configure BGP-LS and BGP-LS peers using RESTCONF by completing the procedure described at the following URL—https://wiki.opendaylight.org/view/BGP_LS_PCEP:User_Guide#Configuration_through_RESTCONF

Step 3 (Optional) Enable the TCP-MD5 modules and services that both BGP-LS and PCEP will use by completing the procedure described at the following URL—https://wiki.opendaylight.org/view/BGP_LS_PCEP:TCP_MD5_Guide#RESTCONF_Configuration

Step 4 To view BGP-LS and PCEP data, do one of the following:

- From the UI, open either the BGPLS Manager or PCEP Manager application.
- Use the RESTCONF APIs provided here—https://wiki.opendaylight.org/view/BGP_LS_PCEP:Restconf

Setting Up NETCONF

In this section, we will cover the following topics:

- How to configure NETCONF support on Cisco ASR 9000 Series and IOS XRv routers
- How to enable NETCONF support on the Cisco Open SDN controller
- How to mount NETCONF-enabled routers to the controller

Note that if your network has 5,000 or more NETCONF-enabled devices connected to the controller, you need to update the datastore's default configuration in order for the datastore to function properly. To do so:

1. Navigate to the following directory: /opt/cisco/controller/etc
2. Open the datastore configuration file (org.opendaylight.controller.cluster.datastore.cfg) in a text editor.
3. Locate these settings and make the following changes:
   - **operation-timeout-in-seconds—**Change from 5 to 30 seconds.
   - **shard-transaction-commit-timeout-in-seconds—**Change from 30 to 600 seconds.
4. Save the changes you have made.

5 Restart the controller.

Configuring NETCONF Support on Cisco ASR 9000 Series and IOS XRv Routers

Step 1 Verify that the following are installed on your device:

- Cisco IOS XR software that supports both NETCONF and YANG
- Base image—asr9k-mini-px.vm
- asr9k-mgbl Package Installation Envelope (PIE)—asr9k-mgbl-px.pie
- asr9k-k9sec PIE—asr9k-k9sec-px.pie

Step 2 Activate crypto keys by opening a shell and entering the following command: **crypto key generate dsa**

Step 3 Configure NETCONF over SSH:

- **ssh server v2**
- **ssh server netconf port 830**
- **xml agent tty**
- **!**
- **netconf agent tty**
- **!**
- **netconf-yang agent ssh**
- **!**
- **crypto key generate rsa**
- **!**
- **end**

Step 4 Configure SNMP for IOS XRv routers:
snmp-server community <*community-string*> **RO/RW**

Enabling NETCONF Support on the Cisco Open SDN Controller

Step 1 Install the Cisco Open SDN controller. See the instructions provided earlier in this document.

Step 2 Select **Features** from the main toolbar's Management menu and verify that NETCONF has been installed.

Setting Up OpenFlow

In this section, we will cover the following topics:

- How to configure OpenFlow support on Cisco ASR 9000 Series routers and Nexus 3000 Series switches
- How to enable OpenFlow support on the Cisco Open SDN controller

Configuring OpenFlow Support on a Cisco ASR 9000 Series Router

Step 1 Verify that the following are installed on your ASR 9000 Series router:

- Cisco IOS-XR software—version 5.1.2 or later
- Base image—asr9k-mini-px.vm
- asr9k-k9sec Package Installation Envelope (PIE)—asr9k-k9sec-px.pie
- asr9k-mpls PIE—asr9k-mpls-px.pie

Step 2 Implement the OpenFlow Agent, as described at the following URL—http://www.cisco.com/c/en/us/td/docs/routers/asr9000/software/asr9k_r5-1/sysman/configuration/guide/b-sysman-cg51xasr9k/b-sysman-cg51xasr9k_chapter_01110.html Focus on the following sections in this document:

- Configuring a Layer 2 Logical Switch for the OpenFlow Agent
- Configuring a Layer 2_Layer 3 Logical Switch for the OpenFlow Agent
- Configuring a Layer 3_VRF Logical Switch for the OpenFlow Agent
- Configuring a Layer 3_Dual-stack Logical Switch for the OpenFlow Agent

Note The tasks you need to complete will vary, depending on the device and model type you are setting up.

Step 3 Make the following global configuration changes:

- `router# configure terminal` (to enter global configuration mode)
- `router(config)# onep` (to enter OneP configuration mode)

Step 4 Verify that a connection has been established between the ASR 9000 device and the controller by pinging one device from the other.

Step 5 Verify that the ASR 9000 device is listed by the Inventory Manager:

a) Log into the controller.
b) From the Applications pane, select **Inventory Manager**. All of the devices associated with the controller are listed here.

c) Locate the entry for the ASR 9000 device and confirm that its Node ID begins with openflow

Step 6 Verify that the following flows have been installed on the controller by running the following command: `router#` **sh openflow switch 3 flows**
The following is an example of what you will see:

```
Thu Jan 15 04:06:27.684 UTC
Logical Switch Id: 3
 Total flows: 2
 Flow: 1
   Match:
   Actions:           drop
   Priority:          0
   Table:             0
   Cookie:            0x2b00000000000005
   Duration:          8202.287s
   Number of packets: 0
   Number of bytes:   0
  Flow: 2
   Match:             dl_type=0x88cc
   Actions:           CONTROLLER:65535
   Priority:          100
   Table:             0
   Cookie:            0x2b00000000000009
   Duration:          8202.288s
   Number of packets: 0
   Number of bytes:   0
 router#
 router#
```

Step 7 In case the ASR 9000 device needs to be operated in Hybrid OpenFlow mode, you must remove the default flows listed in Step 6 by making the following POST requests:

Note Before every RESTCONF request you make, you must first generate a security token. See Making RESTCONF Requests for more information.

(POST request #1)

- URL—https://token:$token@<*controller-IP-address*>/controller/restconf/operations/sal-flow:remove-flow
- Payload—

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?> <input
xmlns="urn:opendaylight:flow:service">
   <flow-table>0</flow-table>
   <node xmlns:inv="urn:opendaylight:inventory">/inv:nodes/inv:node[inv:id="openflow:device-id
 "]</node>
   <priority>0</priority>
</input>
```

(POST request #2)

- URL—https://token:$token@<*controller-IP-address*>/controller/restconf/operations/sal-flow:remove-flow
- Payload—

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?> <input
xmlns="urn:opendaylight:flow:service">
    <flow-table>0</flow-table>
    <node
```

```
xmlns:inv="urn:opendaylight:inventory">/inv:nodes/inv:node[inv:id="openflow:<device-id>"]</node>

    <priority>100</priority>
 </input>
```

Configuring OpenFlow Support on a Cisco Nexus 3000 Series Switch

The following Nexus 3000 Series switches support OpenFlow:

- Nexus 3016
- Nexus 3048
- Nexus 3064
- Nexus 3132Q
- Nexus 3164Q
- Nexus 3172

Complete the following procedure to configure OpenFlow support on any of these switches.

Step 1 Verify that the following are installed on your Nexus 3000 Series switch:

- Cisco NX-OS software—version 6.0(2)U3(1)
- Base package—ofa-1.1.5-r3-n3000-SPA-k9.ova

Step 2 Configure the Cisco Plug-in for OpenFlow, as described at the following URL: http://www.cisco.com/c/en/us/td/docs/switches/datacenter/sdn/configuration/b_openflow_agent_nxos.html

Note The tasks you need to complete will vary, depending on the device and model type you are setting up.

Step 3 Make the following global configuration change:

```
!
hardware profile openflow
!
```

Step 4 Make the following configuration change for each interface on the OpenFlow logical switch:

```
!
switchport mode trunk
spanning-tree port type edge trunk
!
```

Step 5 Verify that a connection has been established between the Nexus 3000 Series switch and the controller by pinging one device from the other.

Step 6 Verify that the Nexus 3000 Series switch is listed by the Inventory Manager:

a) Log into the controller.

b) From the Applications pane, select **Inventory Manager**. All of the devices associated with the controller are listed here.
c) Locate the entry for the Nexus 3000 Series switch and confirm that its Node ID begins with openflow:

Step 7 Open the following RESTCONF URL to check the flows and statistics for the OpenFlow devices you manage:
https://<*controller's-IP-address*>/controller/restconf/operational/opendaylight-inventory:nodes/node/openflow:openflowid

Enabling OpenFlow Support on the Cisco Open SDN Controller

Step 1 Install the Cisco Open SDN controller. See the instructions provided earlier in this document.

Step 2 Select **Features** from the main toolbar's Management menu and verify that OpenFlow has been installed.

Configuring OpenFlow Clusters on Cisco ASR 9000 Series Routers

Step 1 In your router's configuration, specify the IP address for each controller node that will belong to the cluster:

```
openflow
 switch 4 pipeline 132
  interface GigabitEthernet0/0/0/1
  interface GigabitEthernet0/0/0/2
  controller ipv4 <controller1-IP-address> port 6653 security none
  controller ipv4 <controller2-IP-address> port 6653 security none
  controller ipv4 <controller3-IP-address> port 6653 security none
 !
```

Step 2 On the switch, open a console and run the following command:
show openflow switch 4 controllers

The following information is returned:

```
Thu Apr 23 21:22:20.930 UTC

Logical Switch Id: 4
        Total Controllers: Not available
        Controller: 1
           Address             :  <controller1-IP-address>:6653
           Protocol            :  tcp
           VRF                 :  default
           Local Trustpoint:   :  Not available
           Remote Trustpoint:  :  Not available
           Connected           :  Yes
           Role                :  Slave
           last_error          :  Invalid argument
```

```
   state               :  ACTIVE
   sec_since_connect   :  32638
Total Controllers: Not available
Controller: 1
   Address             :  <controller2-IP-address>:6653
   Protocol            :  tcp
   VRF                 :  default
   Local Trustpoint:   :  Not available
   Remote Trustpoint:  :  Not available
   Connected           :  Yes
   Role                :  Unknown
   last_error          :  Invalid argument
   state               :  ACTIVE
   sec_since_connect   :  32655
Total Controllers: Not available
Controller: 1
   Address             :  <controller3-IP-address>:6653
   Protocol            :  tcp
   VRF                 :  default
   Local Trustpoint:   :  Not available
   Remote Trustpoint:  :  Not available
   Connected           :  Yes
   Role                :  Unknown
   last_error          :  Invalid argument
   state               :  ACTIVE
   sec_since_connect   :  32655
```

Step 3 For each controller, view the values configured for the following fields:

- **Connected—**Its value should be `Yes`.
- **Role—**Its value should be either `Master` or `Slave`.
- **State—**Its value should be **ACTIVE**.

Note Due to an OpenFlow plugin bug (CSCup65404), the Role field does not indicate the correct role for a controller node. Until this bug is fixed, the value `Slave` indicates that the node is the master node and the value `Unknown` indicates the node is a slave node.

Configuring OpenFlow Clusters on Cisco Nexus 3000 Series Switches

Step 1 In your switch's configuration, specify the IP address for each controller node that will belong to the cluster:

```
openflow
  switch 1
    protocol-version negotiate
    logging flow-mod
    pipeline 201
    controller ipv4 <controller1-IP-address> port 6653 vrf management security none
    controller ipv4 <controller2-IP-address> port 6653 vrf management security none
```

```
    controller ipv4 <controller3-IP-address> port 6653 vrf management security none
    of-port interface ethernet1/1
    of-port interface ethernet1/2
hardware profile openflow
virtual-service ofan3k4
  activate
```

Step 2 On the switch, open a console and run the following command:
show openflow switch 1 controllers

The following information is returned:

```
Logical Switch Id: 1
Total Controllers: 4
  Controller: 1
    <controller1-IP-address>:6653
    Protocol: tcp
    VRF: management
    Connected: Yes
    Role: Slave
    Negotiated Protocol Version: OpenFlow 1.3
    state:ACTIVE
    sec_since_connect:8

  Controller: 2
    <controller2-IP-address>:6653
    Protocol: tcp
    VRF: management
    Connected: Yes
    Role: Unknown
    Negotiated Protocol Version: OpenFlow 1.3
    state:ACTIVE
    sec_since_connect:19

  Controller: 3
    <controller3-IP-address>:6653
    Protocol: tcp
    VRF: management
    Connected: Yes
    Role: Unknown
    Negotiated Protocol Version: OpenFlow 1.3
    state:ACTIVE
    sec_since_connect:14
```

Step 3 For each controller, view the values configured for the following fields:

- **Connected—**Its value should be `Yes`.
- **Role—**Its value should be either `Master` or `Slave`.
- **State—**Its value should be **ACTIVE**.

Note Due to an OpenFlow plugin bug (CSCup65404), the Role field does not indicate the correct role for a controller node. Until this bug is fixed, the value `Slave` indicates that the node is the master node and the value `Unknown` indicates the node is a slave node.

Example: Configuring OpenFlow Clusters on an OVS Switch

The following example describes how to configure an OpenFlow cluster on an OVS switch using Mininet.

Step 1 Open Mininet and run the following commands:

- **sudo mn --switch ovsk,protocols=OpenFlow13**
- **sudo ovs-vsctl set-controller s1 tcp:**<*controller1-IP-address*>**:6653 tcp:**<*controller2-IP-address*>**:6653 tcp:**<*controller3-IP-address*>**:6653**
- **sudo ovs-vsctl list CONTROLLER**

The following information is returned:

```
mininet@mininet-vm:~$ sudo ovs-vsctl list CONTROLLER
_uuid               : db748dc0-56fb-465a-9cee-66b8e497f64f
connection_mode     : []
controller_burst_limit: []
controller_rate_limit: []
enable_async_messages: []
external_ids        : {}
inactivity_probe    : []
is_connected        : true
local_gateway       : []
local_ip            : []
local_netmask       : []
max_backoff         : []
other_config        : {}
role                : slave
status              : {sec_since_connect="3", state=ACTIVE}
target              : "tcp: <controller3-IP-address>:6653"

_uuid               : 68e7d74e-f0a9-4aa5-998b-209e00b21ff3
connection_mode     : []
controller_burst_limit: []
controller_rate_limit: []
enable_async_messages: []
external_ids        : {}
inactivity_probe    : []
is_connected        : true
local_gateway       : []
local_ip            : []
local_netmask       : []
max_backoff         : []
other_config        : {}
role                : slave
status              : {sec_since_connect="3", state=ACTIVE}
target              : "tcp: <controller2-IP-address>:6653"
```

```
_uuid               : bcd393bf-e8e1-4969-b979-a3e056f17776
connection_mode     : []
controller_burst_limit: []
controller_rate_limit: []
enable_async_messages: []
external_ids        : {}
inactivity_probe    : []
is_connected        : true
local_gateway       : []
local_ip            : []
local_netmask       : []
max_backoff         : []
other_config        : {}
role                : master
status              : {sec_since_connect="3", state=ACTIVE}
target              : "tcp: <controller1-IP-address>:6653"
mininet@mininet-vm:~$
```

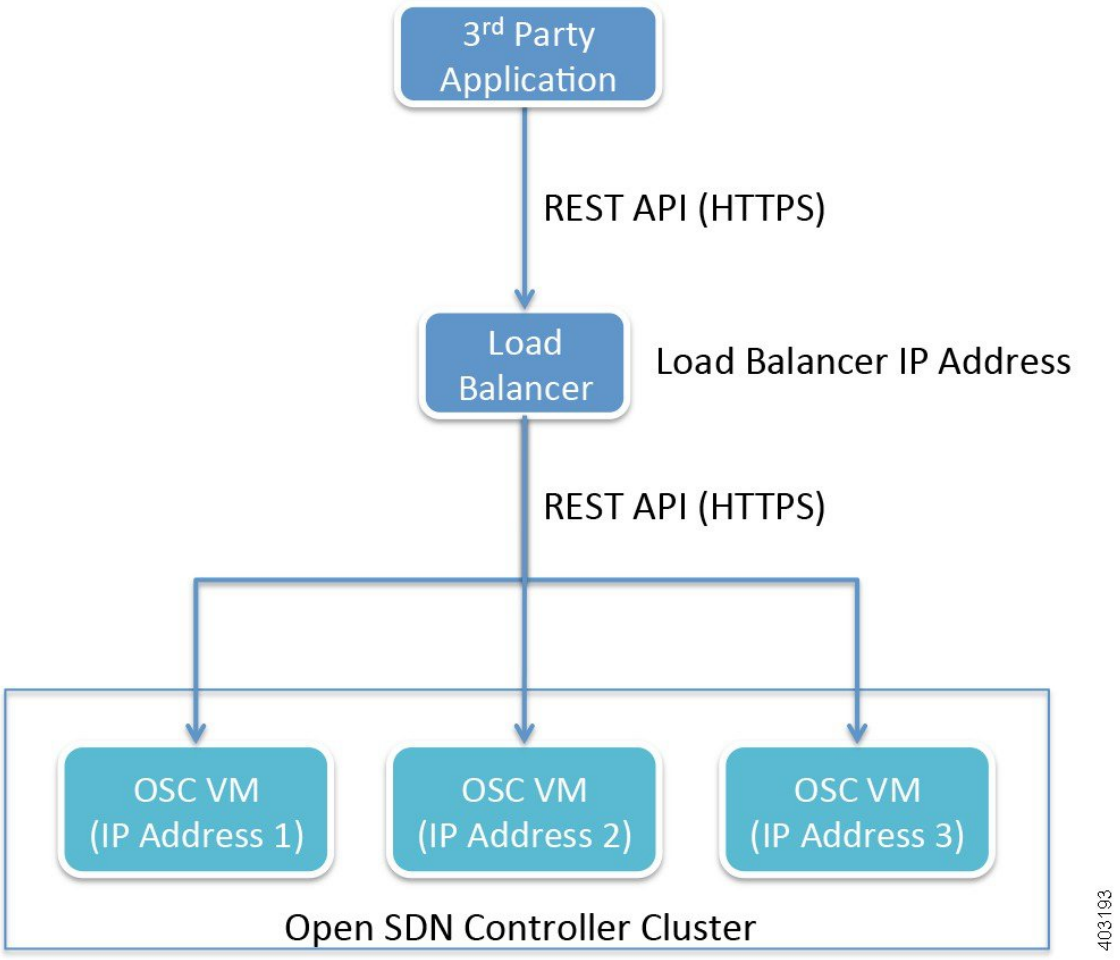
Step 2 Verify that either the master or slave role is assigned to each controller.

REST API Integration with an Open SDN Controller Cluster

Open SDN Controller supports both single and three node cluster deployments. A cluster provides high availability (HA) for the controller in the event of a node failure. When integrating external applications that utilize the REST API with the controller, you need to deploy an external 3rd party load balancer (either hardware or software) to distribute API requests among the controller nodes equally.

To eliminate the possibility of the load balancer being a single point of failure, you can deploy multiple load balancers in either a clustered or active/standby manner, depending on how the load balancers implement HA.

Figure 1: REST API Integration Flow



The following considerations need to be made when selecting the load balancer you want to use:

- HTTPS support
- Node failure and retry handling
- HA support

Once you have chosen a load balancer, refer to its documentation for information on how to configure it for use with the Open SDN Controller cluster. After you have done so, you will then need to register the IP address for each cluster node with the load balancer. Again, refer to the load balancer's documentation for details.

Here are a few points to keep in mind when using a load balancer with a cluster:

- When sending REST API requests, client applications will access the load balancer's IP address (or Virtual IP address in HA deployments).

- Open SDN Controller uses token-based authentication, where the initial HTTPS request uses basic authentication to obtain a token and all subsequent requests are sent with this token. This mechanism should be transparent to the load balancer.
- Since Open SDN Controller utilizes a common cluster-wide token cache, a REST API request does not have any session affinity and can potentially be sent to a different cluster node.
- The load balancer needs to be configured to pass through all HTTP methods (including the OPTIONS method).
- The use of a load balancer will help prevent or minimize the impact of a denial-of-service (DoS) attack.

Installing Wireshark

Before You Begin

- Verify that your controller has an internet connection.
- Download a local copy of the Wireshark source code:
  1. Open the Wireshark download page - https://www.wireshark.org/download.html
  2. From the Stable Release (1.12.4) section, click the Source Code link.

     Ensure that you are downloading the tar file named wireshark-1.12.4.tar.bz2.
  3. Enable read/write permissions to the tar file:
     a. Open a shell and navigate to the directory in which you downloaded the tar file.
     b. Enter the following command:

        chmod 777 wireshark-1.12.4.tar.bz2

- In the following procedure, note that <*directory-1*> is the local directory in which you downloaded the Wireshark source code and <*directory-2*> is the directory on the controller in which you placed a copy of the Wireshark source code.

Step 1 From an SSH client, log into the controller, entering **sysadmin** as the username and **cisco** as the password.
The OS Configuration Console opens.

Step 2 Select the Drop to shell option and then press **Enter**.

Step 3 Stop the iptables service:
sudo service iptables stop

Step 4 Enter the following commands to install the necessary libraries:

- **sudo yum install libpcap libpcap-devel gcc cc**
- **sudo yum install bison**
- **sudo yum install flex**
- **sudo yum install glib2**

- **sudo yum install glib2-devel**

Note Type **y** and then press **Enter** whenever you are asked whether you want to proceed.

Step 5 Use either cURL or SFTP to transfer the Wireshark source code to the controller.
(Via cURL)

Open the cURL tool and enter the following command:

- If you are behind a firewall, **curl -O -x** <*proxy-server-address*>**:**<*proxy-port-number*> **https://1.na.dl.wireshark.org/src/wireshark-1.12.4.tar.bz2 –k**
- If you are not behind a firewall, **curl -O https://1.na.dl.wireshark.org/src/wireshark-1.12.4.tar.bz2 –k**

(Via SFTP)

a) Open an SFTP client and enter the following command:
sftp sysadmin@<*controller-IP-address*>

If prompted, enter **yes** to continue.

b) Enter the controller's password.
c) Enter the following command:
put <*directory-1*>**/wireshark-1.12.4.tar.bz2** /<*directory-2*>/

Step 6 Configure Wireshark on your controller:

- **bunzip2 ./wireshark-1.12.4.tar.bz2**
- **tar -xf wireshark-1.12.4.tar**
- **cd** /<*directory-2*>**/wireshark-1.12.4**
- **./configure --disable-wireshark**
- **make**

Note The make command takes a few minutes to complete.

Step 7 Verify that Wireshark installed successfully:
./tshark -G protocols | grep -i openflow

The resulting output should look like this:

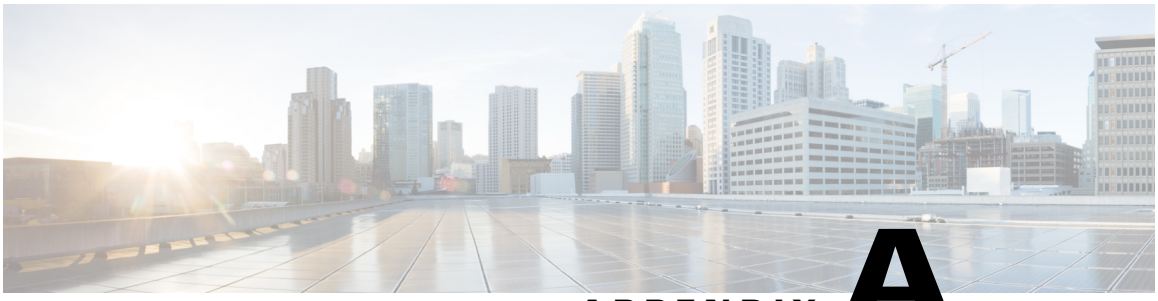
```
OpenFlow       openflow      openflow
OpenFlow 1.0   openflow_v1   openflow_v1
OpenFlow 1.3   openflow_v4   openflow_v4
OpenFlow 1.4   openflow_v5   openflow_v5
```

Step 8 Use Tshark (terminal-based Wireshark) to capture packets on your controller.
For example, say you want to capture packets for interface eth0 and UDP port 1812. To do so, enter the following commands:

- **sudo ./tshark -i eth0 -w** /<*directory-2*>/<*capture-file-name*>**.cap**
- **sudo ./tshark -f "udp port 1812" -i eth0 -w** /<*directory-2*>/<*capture-file-name*>**.cap**

Note that the time it takes to complete this step varies, depending on the number of packets that need to be captured.

Step 9 Start the iptables service:
sudo service iptables start



APPENDIX A

Sample BGP-LS and PCEP Configurations

This appendix provides examples of what BGP-LS and PCEP configurations look like. To access the relevant configuration files, navigate to the /opt/cisco/controller/etc/opendaylight/karaf directory and open them in a text editor to make any necessary changes. The values you need to specify for your setup will differ from the ones provided here.

- BGP-LS Peer, page 27
- ISIS (Redistributes ISIS link states to BGP-LS), page 27
- Interfaces: (Loopback for MPLS-TE and BGP-LS), page 28
- MPLS for PCEP, page 28

BGP-LS Peer

```
router bgp 65504
 bgp router-id 30.30.30.30
 bgp cluster-id 30.30.30.30
 address-family ipv4 unicast
  network 46.0.0.30/32
  network 49.0.0.30/32
  network 55.0.0.30/32
  network 56.0.0.30/32
 !
 address-family link-state link-state
 !
 neighbor 198.18.1.25
  remote-as 65504
  update-source MgmtEth0/0/CPU0/0
  address-family ipv4 unicast
   route-reflector-client
  !
  address-family link-state link-state
   route-reflector-client
  !
 !
```

ISIS (Redistributes ISIS link states to BGP-LS)

```
router isis pce-poc
 is-type level-2-only
 net 72.0000.0000.0030.00
```

```
 distribute bgp-ls level 2
 address-family ipv4 unicast
  metric-style wide
  mpls traffic-eng level-2-only
  mpls traffic-eng router-id Loopback0
  redistribute static
 !
 interface Loopback0
  address-family ipv4 unicast
  !
 !
 interface GigabitEthernet0/0/0/0
  point-to-point
  address-family ipv4 unicast
```

Interfaces: (Loopback for MPLS-TE and BGP-LS)

```
ipv4 unnumbered mpls traffic-eng Loopback0
interface Loopback0
 ipv4 address 30.30.30.30 255.255.255.255
!
interface MgmtEth0/0/CPU0/0
 ipv4 address 198.18.1.57 255.255.255.0
!
interface GigabitEthernet0/0/0/0
 ipv4 address 57.0.0.30 255.255.255.0
!
```

MPLS for PCEP

```
mpls traffic-eng
 interface GigabitEthernet0/0/0/0
 !
 interface GigabitEthernet0/0/0/1
 !
 interface GigabitEthernet0/0/0/2
 !
 interface GigabitEthernet0/0/0/3
 !
 interface GigabitEthernet0/0/0/4
 !
 pce
  peer ipv4 198.18.1.25
  !
  stateful-client
   instantiation
  !
 !
 auto-tunnel pcc
  tunnel-id min 1 max 99
 !
 reoptimize timers delay installation 0
```